Abstract

This book is for all programmers, whether you are a novice or an experienced pro. The beginner will find its carefully paced exercises especially helpful. Of course those who have already familiar with programming are likely to derive more benefits from this book. After reading this book you will find yourself at a moderate level of expertise in C, C++, Java and Python from where you can take yourself to next levels. The command-line interface is one of the nearly all well built trademarks of Linux. There exists an ocean of Linux commands, permitting you to do nearly everything you can be under the impression of doing on your Linux operating system. Although, this to the end of time creates a problem: by all of so copious commands accessible to manage, you don't comprehend where and at which point to fly learning them, especially when you are learner. If you are facing this problem, and are peering for a painless method to begin your command line journey in Linux, you've come to the right place, as in this book, we will launch you to a hold of well liked and helpful Linux commands.
• Educational institutions are teaching it
• Corporate societies are employing it
• Pupils need it
• (Pedagogues desire it... ;)
• (Coders perceive it... ;)


## Contents

<table>
<thead>
<tr>
<th>Linux Commands</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>C Exercises</td>
<td>60</td>
</tr>
<tr>
<td>C++ Exercises</td>
<td>118</td>
</tr>
<tr>
<td>Java Exercises</td>
<td>171</td>
</tr>
<tr>
<td>Python Exercises</td>
<td>246</td>
</tr>
<tr>
<td>PHP Exercises</td>
<td>378</td>
</tr>
<tr>
<td>JavaScript Exercises</td>
<td>403</td>
</tr>
</tbody>
</table>
The command-line interface is one of the nearly all well built trademarks of Linux. There exists an ocean of Linux commands, permitting you to do nearly everything you can be under the impression of doing on your Linux operating system. Although, this to the end of time creates a problem: by all of so copious commands accessible to manage, you don't comprehend where and at which point to fly learning them, especially when you are learner. If you are facing this problem, and are peering for a painless method to begin your command line journey in Linux, you've come to the right place, as in this, we will launch you to a hold of well liked and helpful Linux commands.

---

**Description:**

Display system date and time.

---

**Command:**

date

---

**Description:**

Display calendar.
Command:
cal

Description:
Display date, time and calendar.

Command:
date & cal

Description:
Display August month 2016 year calendar.

Command:
cal 8 2016

Description:
Used to clear the terminal window.

Command:

```
clear
```

Description:

Exit from the terminal window.

Command:

```
extit
```

Description:

Display free and used system memory.

Command:

```
free
```
Description:

Display free and used system memory in bytes.

Command:

free  -b

Description:

Display free and used system memory in kilobytes.

Command:

free  -k

Description:

Display free and used system memory in megabytes.

Command:

free  -m
Description:

Change user password.

Command:

passwd

Description:

Power-off the machine.

Command:

shutdown

Description:

Power-off the machine immediately.

Command:
Description:

Power-off the machine after 10 minutes.

Command:

```
shutdown  -h +10
```

Description:

Print current working directory.

Command:

```
echo $PWD
```

Description:

Print previous working directory.
Command:

```
echo $OLDPWD
```

Description:

Executes the 11th command in command history.

Command:

```
!11
```

Description:

Reveals your command history.

Command:

```
history
```

Description:

Power off or reboot the Operating system.
Command:
sudo reboot

Description:
Display the IP address of the host.

Command:
ip address

Description:
List the size of files and directories.

Command:
ls -s

Description:
View mounted file systems.

Command:

```
mount
```

Description:

Display the information of disk usage of files and directories.

Command:

```
du
```

Description:

Tells you how long the system has been running.

Command:

```
uptime
```
Description:
Set current date as 02 Nov 1988.

Command:
```bash
date -- set 1998-11-02
```

Description:
Set current time as 12:11:02 IST.

Command:
```bash
date -- set 12:11:02
```

Description:
View and change the configuration of the network interfaces on the system.

Command:
```bash
ifconfig
```
Description:

Lists all files and directories in the present working directory.

Command:

ls

Description:

Report the process information.

Command:

ps

Description:

Display disk usage.

Command:
df

Description:
Display disk usage in gigabytes, megabytes, or kilobytes.

Command:
df -H

Description:
Delete every file and every directory.

Command:
rm -r *

Description:
Provides a quick overview of the currently running processes.
Command:

`top`

---

**Description:**

*The system performs an immediate reboot.*

---

Command:

`reboot`

---

**Description:**

*Terminate processes without having to log out or reboot.*

---

Command:

`kill`

---

**Description:**

*Change the current working directory.*
Command:
cd

Description:
Create a new session on the system.

Command:
login

Description:
List open files.

Command:
lsof

Description:
List USB devices.

Command:

lsusb

Description:

Check the status of the network services.

Command:

service network status

Description:

Start the network service.

Command:

service network start
Description:

Stop the network service.

Command:

```
service network stop
```

Description:

Restart the network service.

Command:

```
service network restart
```

Description:

Report information about the users currently on the machine and their processes.

Command:

```
w
```
Description:

Display the current directory.

Command:

pwd

Description:

Displays CPU architecture information (such as number of CPUs, threads, cores, sockets, and more).

Command:

lscpu

Description:

Displays the number of processing units available to the current process.
Command:
nproc

Description:
The system performs an immediate reboot.

Command:
init 6

Description:
Power-off the machine.

Command:
init 0

Description:
List files by date.
Command:
ls -lrt

Description:
Report information about storage devices such as hard disks, flash drives etc.

Command:
lsblk

Description:
Show exit status of previous command.

Command:
echo $?
Lists a few useful info commands.

**Command:**

```
info
```

**Description:**

*Prints current year's calendar.*

---

**Command:**

```
cal -y
```

**Description:**

*Check the status of all the services.*

---

**Command:**

```
service --status-all
```
Description:

Display time in hh:mm:ss.

Command:

date +%T

Description:

Tells when the user last logged on and off and from where.

Command:

last -1 username

Description:

Sort files and directories by extension name.

Command:

ls -X
Description:

Display the manual for the `pwd` command.

Command:

```
man pwd
```

Description:

Displays information about running processes in the form of a tree.

Command:

```
pstree
```

Description:

Resets your terminal.

Command:
Description:

Displays What date is it this Friday.

Command:

date -d fri

Description:

Displays the size of each individual file.

Command:

du -a

Description:

Display information about the Advanced configuration and power Interface.
Command:
acpi

Description:
Takes you two folders back.

Command:
cd ../..

Description:
Takes you to the previous directory.

Command:
cd -

Description:
Displays a list of shell built-in commands.
Command:

help

Description:

Lists your last logins.

Command:

last yourusername

Description:

Create a new directory called myfiles.

Command:

mkdir myfiles

Description:
Remove the directory myfiles.

Command:

`rmdir myfiles`

Description:

Disable password for a specific user "root1".

Command:

`passwd -d root1`

Description:

Switch to user "root1".

Command:

`sudo su root1`
Description:

Exit from the terminal window.

Command:

logout

Description:

Creates a user "root1".

Command:

useradd "root1"

Description:

Assign password to user "root1".

Command:

passwd "root1"
Description:
Repeats the last command.

Command:
!!

Description:
Display Who you are logged in as.

Command:
whoami

Description:
Display the login name of the current user.

Command:
Description:

Report the name of the kernel.

Command:

uname

Description:

Print the kernel version.

Command:

uname  -v

Description:

Print the operating system.
Command:

uname -o

Description:

Report the machine hardware name.

Command:

uname -m

Description:

Print version information and exit.

Command:

uname --version

Description:

Print the kernel release.
Command:

```
uname  -r
```

**Description:**

Report the network node hostname.

---

Command:

```
uname  -n
```

**Description:**

Display all port connections (both TCP and UDP).

---

Command:

```
netstat  -a
```

**Description:**
Display only TCP (Transmission Control Protocol) port connections.

Command:

netstat -at

Description:

Display only UDP (User Datagram Protocol) port connections.

Command:

netstat -au

Description:

Display all active listening ports.

Command:

netstat -I
Description:
Display all active listening TCP ports.

Command:
netstat -It

Description:
Display all active listening UDP ports.

Command:
netstat -lu

Description:
Reveal all the information about the current user (user id, username, group id, group name etc.).

Command:
Description:

Reveal all the information about the user "root1" (user id, username, group id, group name etc.).

Command:

id root1

Description:

Print the machine's architecture.

Command:

arch

Description:

Display the list of available fonts.
Command:

fc-list

Description:

Create two directories (myfiles, files).

Command:

mkdir myfiles files

Description:

install apache (CentOS).

Command:

yum install httpd

Description:
install apache (Ubuntu).

Command:

```
apt install httpd
```

Description:

upgrade apache (CentOS).

Command:

```
yum update httpd
```

Description:

upgrade apache (Ubuntu).

Command:

```
apt update httpd
```
Description:

uninstall apache (CentOS).

Command:

yum remove httpd

Description:

uninstall apache (Ubuntu).

Command:

apt remove httpd

Description:

Display usage summary for the command (date).

Command:

date --help
Description:

List active connections to/from system.

Command:

ss -tup

Description:

List internet services on a system.

Command:

ss -tupl

Description:

Display all active UNIX listening ports.

Command:
netstat -lx

Description:
Display all the active interfaces details.

Command:
ifconfig

Description:
Display information of all network interfaces.

Command:
ifconfig -a

Description:
Compare the contents of two files (1.txt, 2.txt).
Command:

diff 1.txt 2.txt

Description:

Tells you how many lines, words, and characters there are in a file (1.txt).

Command:

wc 1.txt

Description:

Compresses file (1.txt), so that it take up much less space.

Command:

gzip 1.txt

Description:

Uncompresses file (1.txt) compressed by gzip.
Command:

gunzip 1.txt

Description:

Examine the contents of the file (1.txt).

Command:

cat 1.txt

Description:

Display calendar.

Command:

ncal
Removes the file (1.txt).

Command:

```bash
rm 1.txt
```

Description:

Rename a file named 1.txt to 0.txt.

Command:

```bash
mv 1.txt 0.txt
```

Description:

Replace the contents of 0.txt with that of 1.txt.

Command:

```bash
cp 1.txt 0.txt
```
Description:
Create a empty file (test.txt).

Command:
touch test.txt

Description:
Print the last 10 lines of a file (1.txt).

Command:
tail 1.txt

Description:
Print N number of lines from the file (1.txt).

Command:
tail -n N 1.txt
Description:

Prints the number of words in a file (1.txt).

Command:

wc -w 1.txt

Description:

Prints the number of characters from a file (1.txt).

Command:

wc -m 1.txt

Description:

Prints the length of the longest line in a file (1.txt).

Command:
wc -L 1.txt

Description:

Print information about usb ports, graphics cards, network adapters etc.

Command:

lspci

Description:

View contents of a file (1.txt).

Command:

less 1.txt

Description:

Display calendar (last month, current month, and next month).
Command:
cal -3

Description:
Compare the contents of three files (1.txt, 2.txt, 3.txt) line by line.

Command:
diff3 1.txt 2.txt 3.txt

Description:
Compare two files (1.txt, 2.txt) line-by-line.

Command:
comm 1.txt 2.txt

Description:
Perform byte-by-byte comparison of two files (1.txt, 2.txt).
Command:

cmp 1.txt 2.txt

Description:

Prints the CRC checksum and byte count for the file "myfiles.txt".

Command:

cat

cat 1.txt 2.txt > 0.txt

Description:

Append contents of files (1.txt, 2.txt) into one file (0.txt).
Append contents of files (1.txt, 2.txt, 3.txt) into one file (0.txt).

Command:

```bash
sed  r 1.txt 2.txt 3.txt > 0.txt
```

Description:

Append contents of files (1.txt, 2.txt, 3.txt) into one file (0.txt).

Command:

```bash
sed  h 1.txt 2.txt 3.txt > 0.txt
```

Description:

Append contents of files (1.txt, 2.txt, 3.txt) into one file (0.txt).

Command:

```bash
sed  -n  p 1.txt 2.txt 3.txt > 0.txt
```

Description:

Append contents of files (1.txt, 2.txt, 3.txt) into one file (0.txt).
**Shortcuts:**

<table>
<thead>
<tr>
<th>ctrl+c</th>
<th>Halts the current command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ctrl+z</td>
<td>Stops the current command</td>
</tr>
<tr>
<td>ctrl+d</td>
<td>Logout the current session</td>
</tr>
<tr>
<td>ctrl+w</td>
<td>Erases one word in the current line</td>
</tr>
<tr>
<td>ctrl+u</td>
<td>Erases the whole line</td>
</tr>
<tr>
<td>ctrl+r</td>
<td>Type to bring up a recent command</td>
</tr>
</tbody>
</table>

**Description:**

**Writes contents of a file (0.txt) to output, and prepends each line with line number.**

**Command:**

```
nl 0.txt
```

**Description:**

**Create a empty file (test1.txt) inside a directory (test).**
Command:

```
mkdir test
cd test
pwd
touch test1.txt
```

Description:

**Gather information about hardware components such as CPU, disks, memory, USB controllers etc.**

Command:

```
sudo lshw
```

Description:

**Gather information about file system partitions.**

Command:

```
sudo fdisk -l
```
Description:

Displays the line (good morning) in which the string (good) is found in the file (1.txt).

Command:

grep good 1.txt

Description:

Append contents of files (1.txt, 2.txt, 3.txt) into one file (0.txt) using for loop.

Command:

for i in {1..3}; do cat "$i.txt" >> 0.txt; done

Description:

Search for files (test.txt, test1.txt, test2.txt, test.php, test.html) in a directory as well as its sub-directories.
Command:

find test*

Description:

Displays status related to a file (1.txt).

Command:

stat 1.txt

###

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vi</td>
<td>Open vi editor</td>
</tr>
<tr>
<td>i</td>
<td>Go to Insert mode</td>
</tr>
<tr>
<td>a =20;</td>
<td></td>
</tr>
<tr>
<td>b =64;</td>
<td></td>
</tr>
<tr>
<td>print (a + b);</td>
<td></td>
</tr>
<tr>
<td>Hit Escape to return to Normal mode.</td>
<td></td>
</tr>
<tr>
<td>:w hello.py</td>
<td>Save text</td>
</tr>
<tr>
<td>:q</td>
<td>Quit</td>
</tr>
<tr>
<td>python hello.py</td>
<td>Print the output:84</td>
</tr>
</tbody>
</table>
**Description:**

Download the file (file.txt) from url "http://website.com/files/file.txt".

**Command:**

```
wget http://website.com/files/file.txt
```

**Description:**

Display host's numeric ID in hexadecimal format.

**Command:**

```
hostid
```

**Description:**

Display file type of the file (myfiles.txt).

**Command:**

```
file myfiles.txt
```
Description:

Create a file (myfiles.txt) containing a text (Hello World).

Command:

```
echo 'Hello World' > myfiles.txt
```

Description:

Create a file (myfiles.txt) containing a text (Hello World).

Command:

```
printf 'Hello World' > myfiles.txt
```

Description:

Display IP address of the hostname.

Command:

Command:
Description:

Add a new line of text to an existing file (1.txt).

Command:

```bash
echo "Hello world!" >> 1.txt
echo "this is 2nd line text" >> 1.txt
echo "last line!" >> 1.txt
```

Description:

Displays a single line description about a command (cal).

Command:

```bash
whatis cal
```
| vi               | Open vi editor         |
| i                | Go to Insert mode      |
| Type some text. |                       |
| Hit Escape to return to Normal mode. |
| :w test.txt     | Save text              |
| :q               | Quit                   |
| :q!              | Quit without saving    |

###
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vi</td>
<td>Open vi editor</td>
</tr>
<tr>
<td>i</td>
<td>Go to Insert mode</td>
</tr>
<tr>
<td>$name = &quot;Paul&quot;;</td>
<td></td>
</tr>
<tr>
<td>print &quot;$name&quot;;</td>
<td></td>
</tr>
<tr>
<td>Hit Escape to return to Normal mode.</td>
<td></td>
</tr>
<tr>
<td>:w hello.pl</td>
<td>Save text</td>
</tr>
<tr>
<td>:q</td>
<td>Quit</td>
</tr>
<tr>
<td>perl hello.pl</td>
<td>Print the output: Paul</td>
</tr>
</tbody>
</table>

###
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vi</td>
<td>Open vi editor</td>
</tr>
<tr>
<td>i</td>
<td>Go to Insert mode</td>
</tr>
<tr>
<td>echo &quot;What is your name?&quot;</td>
<td></td>
</tr>
<tr>
<td>read PERSON</td>
<td></td>
</tr>
<tr>
<td>echo &quot;Hello, $PERSON&quot;</td>
<td></td>
</tr>
<tr>
<td>Hit Escape to return to Normal mode.</td>
<td></td>
</tr>
<tr>
<td>:w hello.sh</td>
<td>Save text</td>
</tr>
<tr>
<td>:q</td>
<td>Quit</td>
</tr>
<tr>
<td>sh hello.sh</td>
<td>Output:</td>
</tr>
<tr>
<td></td>
<td>What is your name?</td>
</tr>
<tr>
<td></td>
<td>If you enter: Zara Ali</td>
</tr>
<tr>
<td></td>
<td>Hello, Zara Ali</td>
</tr>
</tbody>
</table>
Description:

Check the network connectivity between host (your connection) and server (Google server).

Command:

ping google.com

Description:

Find the location of source/binary file of a command (cal).

Command:

whereis cal

Description:

List the files in the bin directory.
Command:

`ls /bin`

Description:

List the files in the bin directory and the etc directory.

Command:

`ls /bin /etc`

Description:

Moves the file test.txt to the folder newrepo.

Command:

`mv test.txt ./newrepo`

Description:

Deletes all the lines in the test.txt containing the word.
Command:

```bash
sed -i "/tue/d" test.txt
```

```python
import subprocess
subprocess.call(\'linux command\')
```

```python
import os
os.system(\'linux command\')
```

```python
import os
os.system(\'ls\')
```

**List all the files and directories in the current directory**
**Command:**

`last reboot`

**Description:**

Show system reboot history

**Command:**

`dmesg`

**Description:**

Displays the messages from the kernel ring buffer (a data structure that records messages related to the operation of the kernel)

**Command:**

`cat /proc/cpuinfo`

**Description:**

Display CPU information
Command:

cat /proc/meminfo

Description:
Display memory information

Command:

lspci -tv

Description:
Display PCI (Peripheral Component Interconnect) devices

Command:

lsusb -tv

Description:
Display USB devices
Command:

free -h

Description:

Display free and used memory ( -h for human readable, -m for MB, -g for GB)

Command:

mpstat 1

Description:

Display processor related statistics

Command:

vmstat 1

Description:
Display virtual memory statistics

**Command:**

```bash
iostat 1
```

**Description:**

Display Input / Output statistics

**Command:**

```bash
watch df -h
```

**Description:**

Execute "df -h" command, showing periodic updates

**Command:**

```bash
ps -ef
```

**Description:**
Display all the currently running processes on the system

**Command:**

`ip a`

**Description:**

Display all network interfaces and IP address

**Command:**

`dig wikipedia.org`

**Description:**

Display DNS information for domain (wikipedia.org)

**Command:**

`host wikipedia.org`

**Description:**
Display the IP address details of the specified domain (wikipedia.org)

**Command:**

```bash
netstat -nutlp
```

**Description:**
Display listening Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP) ports and corresponding programs

**Command:**

```bash
rpm -qa
```

**Description:**
List all installed packages

**Command:**

```bash
yum list installed
```

**Description:**
List all installed packages (CentOS)
yum info httpd

**Description:**

Display description and summary information about package "httpd" (CentOS)

**Command:**

du -ah

**Description:**

Display disk usage for all files and directories in human readable format

**Command:**

du -sh

**Description:**

Display total disk usage off the current directory

**Command:**
Description:

Change to the /etc directory

Command:

cd /etc

Description:

List the status of all the processes along with process id and PID

Command:

ps -A

#include <stdio.h>
int main()
{
    printf("Hello world\n");
    return 0;
}

gcc Hello.c
Description:

Compile the C program saved in Hello.c file

Command:

```c
#include <iostream>
int main()
{
    std::cout << "Hello world!";
    return 0;
}

```
g++ Hello.cpp

Description:

Compile the C++ program saved in Hello.cpp file

Command:

tty

Description:

Displays the file name of the terminal connected to standard input
**Command:**

```java
public class MyClass {
    public static void main(String [] args) {
        System.out.println("Hello, World!");
    }
}
```

`javac MyClass.java`

**Description:**

Compile the Java program saved in `MyClass.java` file using `javac` compiler

**Command:**

```bash
od -b myfiles.txt
```

**Description:**

Displays the contents of `myfiles.txt` file in octal format

**Command:**

```bash
od -c myfiles.txt
```
**Description:**

Displays the contents of *myfiles.txt* file in character format

**Command:**

```
od -An -c myfiles.txt
```

**Description:**

Displays the contents of *myfiles.txt* file in character format but with no offset information

**Command:**

```
csplit myfiles.txt 13 62 101
```

**Description:**

If the file *myfiles.txt* has 123 lines, the `csplit` command would create four files: the xx00 file would contain lines 1–12, the xx01 file would contain lines 13–61, the xx02 file would contain lines 62–100, the xx03 file would contain lines 101–123

**Command:**
md5sum myfiles.txt

**Description:**

Prints a 32-character (128-bit) checksum of `myfiles.txt` file using the MD5 algorithm

**Command:**

more myfiles.txt

**Description:**

Displays the content of `myfiles.txt` file

**Command:**

sha1sum myfiles.txt

**Description:**

Prints SHA1 (160-bit) checksum of `myfiles.txt` file

SHA 1 → Secure Hash Algorithm 1

**Command:**
shred myfiles.txt

**Description:**

Overwrites the `myfiles.txt` file repeatedly – in order to make it harder for even very expensive hardware probing to recover the data

**Command:**

```
cat myfile.txt
```

```
01. Einstein
02. Newton
03. Maxwell
04. Tesla
05. Edison
```

```
tac myfile.txt
```

```
05. Edison
04. Tesla
03. Maxwell
02. Newton
01. Einstein
```

**Description:**

Print the lines of `myfile.txt` in reverse (from last line to first)
Command:

chkconfig --list

Description:
Displays a list of system services and whether they are started (on) or stopped (off) in run levels 0–6

Command:

chkconfig --list

Description:
Displays a list of system services and whether they are started (on) or stopped (off) in run levels 0–6

Command:

halt -p

Description:
Power-off the system
Command:

lastlog

Description:

Prints the details of the last login (login-name, port and last login time)

Command:

lastlog -t 1

Description:

Displays the login information (1 day ago)

Command:

lastlog -u manju

Description:

Display lastlog information for a particular user (manju)
**Command:**
cat /etc/passwd

more /etc/passwd

less /etc/passwd

getent passwd

**Description:**
List all users on Linux

**Command:**
tail -5 /etc/passwd

head -5 /etc/passwd
Description:

List last 5 users on Linux

List first 5 users on Linux

Command:

```
wall "The system will be shutdown in 10 minutes."
```

Description:

The message (The system will be shutdown in 10 minutes.) will be broadcasted to

all users that are currently logged in

Command:

```
chage -l manju
```

Description:

List the password and its related details for a user (manju)
Command:

```
chage -M 10 manju
```

Description:
Set Password Expiry Date for an user (manju)

Command:

```
chage -E "2020-07-30" manju
```

Description:
Set the Account Expiry Date for an User (manju)

Command:

```
chage -I 10 manju
```

Description:
Force the user (manju) account to be locked after 10 inactivity days
Command:

ftp 192.168.42.77

Description:

Connect to an FTP server at remote server IP address "192.168.42.77"

Command:

arp -a

Description:

Lists all the peers connected at various interfaces along with their MAC Addresses and IP addresses

Command:

dnsdomainname

Description:
Display the system's DNS domain name

**Command:**

```
domainname
```

**Description:**

Display the name of the domain your machine belongs to

**Command:**

```
echo 'Hello World!' | base64
```

**Output:** SGVsbG8gV29ybGQhCg==

**Description:**

Encode text (Hello World!) to base64

**Command:**

```
echo 'SGVsbG8gV29ybGQhCg==' | base64 -d
```

**Output:** Hello World!

**Description:**
Decoded text (SGVsbG8gV29ybGQhCg==) is: Hello World!

**Command:**

```
fc-cache -f -v
```

**Description:**

Build font information cache files

**Command:**

```
cat 1.txt

Einstein
Newton
Albert
```

```
fmt 1.txt

Einstein Newton Albert
```

**Description:**

Formats text in a single line
Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics.
Albert Einstein was a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics.
a German-born theoretical physicist, widely acknowledged to be one of the greatest physicists of all time. Einstein is known for developing the theory of relativity, but he also made important contributions to the development of the theory of quantum mechanics.

Command:

traceroute google.com

Description:

Prints the route that a packet takes to reach the Google (172.217.26.206) host from the local machine

Command:

cat 1.txt
Description:

View the contents of zipped file

Command:

```
zdiff 1.txt.gz 2.txt.gz
```
Command:

ss | less

Description:
List all connections

Command:

ss -aA tcp

Description:
Filter out TCP (Transmission Control Protocol) connections

Command:

ss -aA udp

Description:
Filter out UDP (User Datagram Protocol) connections
Command:

ss -lnt

Description:

Display only listening sockets

Command:

ss -ltp

Description:

Print process name and PID

Command:

ss -s

Description:

Print summary statistics
**Command:**

`ss -tl6`

**Description:**

Display only IPv6 connections

---

**Command:**

`ss -tl -f inet`

**Description:**

Display only IPv4 socket connections

---

**Command:**

`ss -t4 state established`

**Description:**

Display all IPv4 TCP sockets that are in connected state
Command:

```
pmap 3244
```

Description:

View the memory map of a process with Process ID (3244)

Command:

```
apropos -r 'remove file'
```

Description:

Find command that removes file

Command:

```
apropos editor
```

Description:

Display information about the editing programs that are available on a system
Command:

apropos pstree

Description:

Provide information about the pstree command (which displays the names of the processes currently on the system in the form of a tree diagram)

The `apropos` command is useful when you know what you want to do, but you have no idea what command you should be using to do it. If you were wondering how to locate files, for example, the commands

apropos find

and

apropos locate

would have a lot of suggestions to offer.

basename /etc/passwd

Output: passwd

basename /usr/local/apache2/conf/httpd.conf
Output: httpd.conf

echo a b c d e f | xargs

Output: a b c d e f

echo a b c d e f | xargs -n 3

Output: 

\{ display only 3 items per line \}

a b c
d e f

Command:

env

Description:

Print out a list of all environment variables
printenv HOME

**Description:**

Print HOME variable value

cat score.txt

Albert-30
John-50
William-80
Stephen-20
Justin-40

cut -d - -f2 score.txt

30
50
80
20
40

cut -d - -f1 score.txt

Albert
John
William
Stephen
Justin
cat 1.txt
Hello World

cat 2.txt
Computer Program

paste 1.txt  2.txt
Hello World  Computer Program

join 1.txt  2.txt
Hello World  Computer Program
**Command:**

```
rev 1.txt
```

**Description:**

Reverse lines of a file (1.txt)

```
cat 3.txt

22
33
11
77
55

sort 3.txt

11
22
33
55
77
```

sorts numeric values in 3.txt file and displays sorted output
cat 1.txt

Hello World

cat 1.txt | tr "[a-z]" "[A-Z]"

HELLO WORLD

cat 5.txt

zz
zz
yy
yy
yy
xx

uniq 5.txt

zz  
yy  
xx

removes duplicate lines and displays unique lines
**Command:**

```
ls -l *.txt
```

**Description:**

Lists the files with .txt extension

---

The thing with Linux is that the developers themselves are actually customers too: that has always been an important part of Linux.

*Linus Torvalds*
<table>
<thead>
<tr>
<th>Linux</th>
<th>Unix</th>
</tr>
</thead>
<tbody>
<tr>
<td>Free to use <em>(open source)</em></td>
<td>Licensed Operating System <em>(closed source)</em></td>
</tr>
<tr>
<td>Linux is just the kernel</td>
<td>Unix is a complete package of Operating System</td>
</tr>
<tr>
<td>Bash <em>(Bourne Again SHell)</em> is default shell for Linux</td>
<td>Bourne Shell is default shell for Unix</td>
</tr>
<tr>
<td>Portable and is booted from a USB Stick</td>
<td>Unportable</td>
</tr>
<tr>
<td>Source code is accessible to the general public</td>
<td>Source code is not accessible to anyone</td>
</tr>
<tr>
<td>Uses Graphical User Interface with an optional Command Line Interface</td>
<td>Uses Command Line Interface</td>
</tr>
</tbody>
</table>

**Command:**

```bash
echo $SHELL
```

**Description:**

Print the Default shell of user

**Command:**

```bash
echo $0
```

**Description:**
Display the name of the currently running process ($0 is the name of the running process). If you use it inside of a shell then it will return the name of the shell. If you use it inside of a script, it will return the name of the script.

**Command:**

```
echo *
```

**Description:**

Print all files and folders – similar to ls command.

**Command:**

```
ps -p $$
```

**Output:**

```
PIDS TTY TIME CMD
3352 pts/0 00:00:00 bash
```

**Description:**

Print the process ID of the current shell ($$ is the process ID of the current shell).

**Command:**
```
cat /etc/shells
```

**Description:**

List shells

**Command:**

`last`

**Description:**

List last logins of users and what happened such as "shutdown" or "crash" etc.

**Command:**

`last`

**Description:**

List last logins of users and what happened such as "shutdown" or "crash" etc.

**Command:**
bzip2 -k phy.txt

**Description:**

Compresses but does not deletes the original file

phy.txt → phy.txt.bz2

Command:

bzip2 -d phy.txt.bz2

**Description:**

Decompresses the compressed file (phy.txt.bz2)

phy.txt.bz2 → phy.txt

Command:

bzcat phy.txt.bz2
Display the contents of compressed file (phy.txt.bz2)

Command:

```
bunzip2 phy.txt.bz2
```

Decompresses the compressed file (phy.txt.bz2)

Command:

```
crontab -l
```

Display current logged-in user's crontab entries

Command:

```
cat /dev/null > phy.txt
```
```bash
cp /dev/null phy.txt

echo "" > phy.txt

Description:
Empty the content of a file (phy.txt)

Command:

nohup ping google.com &

Description:
Ping google.com and send the process to the background

Command:
nohup ping google.com > log.txt &

**Description:**

Save the ping logs to log.txt

pgrep -a ping

Output:

3858 ping google.com
4200 ping google.com
4236 ping google.com

kill 3858

pgrep -a ping

Output:

4200 ping google.com
4236 ping google.com

**Command:**

ls -la /home
Description:

Display the contents of /home

Command:

```
sudo shutdown 2
```

Description:

Power-off the machine after 2 minutes

Command:

```
shutdown -c
```

Description:

Cancel the shutdown process

Command:

```
pr 36.txt
```
**Description:**

Display the contents of the file (36.txt) one page after the other

**Command:**

```
stty -a
```

**Description:**

Display all current terminal settings

**Command:**

```
ls -1
```

**Description:**

List files one per line

**Command:**

```
yes John

**Description:**

Outputs a string (John) repeatedly until killed

**Command:**

`vdir`

**Description:**

List files and directories in the current directory (one per line) with details

**Command:**

`who -b`

**Description:**

Print when the system was booted

# Open phy.txt with nano
nano phy.txt

# Open phy.txt with vim

vim phy.txt

User Request

Shell  Application

Linux Kernel

Computer Hardware
# Displays information regarding the users currently on the machine, login time, IDLE time, TTY and CPU time

Output:

11:12:10 up 1:29, 2 users, load average: 0.02, 0.04, 0.10

<table>
<thead>
<tr>
<th>USER</th>
<th>TTY</th>
<th>FROM</th>
<th>LOGIN@</th>
<th>IDLE</th>
<th>JCPU</th>
<th>PCPU</th>
<th>WHAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>manju</td>
<td>:0</td>
<td>:0</td>
<td>02:43</td>
<td>?xdm?</td>
<td>3:30</td>
<td>0.65s</td>
<td>gdm-session-worker [pa</td>
</tr>
<tr>
<td>manju</td>
<td>pts/0</td>
<td>:0</td>
<td>11:01</td>
<td>2.00s</td>
<td>0.10s</td>
<td>0.01s</td>
<td>w --ip-addr</td>
</tr>
</tbody>
</table>

w -short

# Omits CPU time and login information

Output:

11:11:46 up 1:28, 2 users, load average: 0.02, 0.04, 0.11

<table>
<thead>
<tr>
<th>USER</th>
<th>TTY</th>
<th>FROM</th>
<th>IDLE</th>
<th>WHAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>manju</td>
<td>:0</td>
<td>:0</td>
<td>?xdm?</td>
<td>gdm-session-worker [pam/gdm-password]</td>
</tr>
<tr>
<td>manju</td>
<td>pts/0</td>
<td>:0</td>
<td>2.00s</td>
<td>w --short</td>
</tr>
</tbody>
</table>

**Command:**

`findmnt`

**Description:**

Display a list of currently mounted file systems
**Description:**
List IP addresses and network interfaces

**Command:**
```
ip addr show
```

---

**Description:**
List active (listening) ports

**Command:**
```
netstat -pnltu
```

---

**Description:**
Display systemd, kernel and journal logs

**Command:**
```
journalctl
```
**Description:**

Display the status of network service

**Command:**

```bash
sudo systemctl status network
```

**Description:**

Start the network service

**Command:**

```bash
sudo systemctl start network
```

**Description:**

Stop the network service

**Command:**

```bash
sudo systemctl stop network
```
**Description:**

Display the current state of Booleans

**Command:**

```bash
getenforce
```

**Description:**

Reports whether SELinux is enforcing, permissive or disabled

---

**Security-Enhanced Linux (SELinux)** is a security architecture for Linux systems that allows administrators to have more control over who can access the system.

---

```bash
setenforce 0
```
getenforce

Output:
Permissive

setenforce 1

getenforce

Output:
Enforcing

- **Enforcing** - SELinux security policy is enforced.
- **Permissive** - SELinux prints warnings instead of enforcing.
- **Disabled** - No SELinux policy is loaded.
**Command:**

`sestatus`

**Description:**

Display the current status of the SELinux that is running on your system

**Command:**

`ps -aef`

**Description:**

Display full listing of processes on your system

**Command:**

`sar`
Display System Activity Report

Command:

```
ulimit
```

Description:

Report the resource limit of the current user

Output:

```
Unlimited
```

The current user can consume all the resources the current system supports

2 types of resource limitation:

- **Hard resource limit**: The physical limit that the user can reach.
- **Soft resource limit**: The limit that is manageable by the user (its value can go up to the hard limit)

Command:
ulimit -a

**Description:**

Report all the resource limits for the current user

**Command:**

ulimit -s

**Description:**

Check the maximum stack size of the current user

**Command:**

ulimit -e

**Description:**

Check out the max scheduling priority of the current user
Command:

ulimit -u

Description:
Display the maximum number of user processes

Command:

ulimit -v

Description:
Check out the size of virtual memory

Command:

ulimit -n

Description:
Check out how many file descriptors a process can have
Command:

man limits.conf

Description:
Display the in-depth information on the limits.conf configuration file

Command:

sar -V

Description:
Display the sar version

Command:

sar -u 2 5

Description:
Report CPU details total 5 times with the interval of 2 seconds
Command:

`sar -n DEV 1 3 | egrep -v lo`

Description:

Report about network interface, network speed, IPV4, TCPV4, ICMPV4 network traffic and errors

Command:

`sar -v 1 3`

Description:

Report details about the process, kernel thread, i-node, and the file tables

Command:

`sar -S 1 3`

Description:

Report statistics about swapping
**Command:**

```
sar -b 1 3
```

**Description:**

Report details about I/O operations like transaction per second, read per second, write per second

**Command:**

```
sudo systemctl status firewalld
```

**Description:**

Display the status of the `firewalld` service

**Command:**

```
sudo systemctl start firewalld
```

**Description:**

Start the `firewalld` service
**Command:**

```bash
classic-firewall-config
```

**Description:**

Start the graphical firewall configuration tool

---

**Command:**

```bash
classic-firewall-cmd --list-all-zones
```

**Description:**

List all zones

---

**firewalld** is a firewall management tool for Linux operating systems
**Description:**

Check the currently set default zone

**Command:**

```
firewall-cmd --get-default-zone
```

**Description:**

Display currently allowed service on your system

**Command:**

```
firewall-cmd --list-services
```

**Description:**

List the ports that are open on your system

**Command:**

```
firewall-cmd --list-ports
```
firewall-cmd --zone=work --list-services

**Description:**

List services that are allowed for the public zone

**Command:**

mtr --report google.com

**Description:**

Provides information about the route that Internet traffic takes between the local system and a remote host (google.com)

**Command:**

sudo yum install samba

**Description:**

install Samba (CentOS)
**Samba** is client/server technology that implements network resource sharing across operating systems. With Samba, files and printers can be shared across Windows, Mac and Linux/UNIX clients.

**Command:**

```
sudo firewall-cmd --add-service samba --permanent
```

**Description:**

Add Samba service to firewall

**Command:**

```
zip q.zip q.txt
```

**Description:**

Create a zip file (q.zip)
unzip q.zip

Description:

Unzip a zip file (q.zip)

zipcloak q.zip

# zipcloak prompts you for a password, and then ask you to confirm it:

Enter password:
Verify password:

...if the passwords match, it encrypts q.zip file

unzip q.zip

# When you try to unzip the q.zip file, it prompts you for the password before allowing you to extract the file (q.txt) it contains
Command:

grep -l "Einstein" *

Description:

Display the names of the files with the word (Einstein) present in it

Command:

zipsplit -n 1048576 q.zip

Description:

Split q.zip file to create a sequence of zipfiles (q1.zip, q2.zip.....) – each no larger than 1048576 bytes (one megabyte)

You could concatenate (q1.zip, q2.zip.....) into a new file, w.zip, with the command:

    cat q*.zip > w.zip
C Exercises

Question 1

Question:

Write a program to print Hello World!.

Solution:

```c
#include <stdio.h>

int main()
{
    printf("Hello, world!");
    return 0;
}
```

Question 2

Question:

Write a program to find the area of a circle.
Solution:

```c
#include <stdio.h>
int main()
{
    int r, area;
    r = 2;
    area = 3.14 * r * r;
    printf("The area of the circle = \%d", area);
    return 0;
}
```

Question 3

Question:

Write a program to find the sum of two numbers.

Solution:

```c
#include <stdio.h>
int main()
{
    int a, b, sum;
    a=1;
    b=2;
    sum = a + b;
    printf("The sum of a and b = \%d", sum);
    return 0;
}
```
Question 4

Question:

Write a program to find the square of a number.

Solution:

```c
#include<stdio.h>
#include<math.h>

int main()
{
    int a, b;
    a=2;
    b = pow((a), 2);
    printf("The square of a = %d", b);
    return 0;
}
```

Question 5

Question:

Write a program to find the greatest of two numbers.
Solution:

```c
#include <stdio.h>
int main()
{
    int a, b;
    a = 2;
    b = 3;
    if (a > b)
    {
        printf("a is greater than b");
    }
    else
    {
        printf("b is greater than a");
    }
    return 0;
}
```

Question 6

Question:

Write a program to print the average of the elements in the array.

Solution:

```c
#include <stdio.h>
int main()
{
    int i, avg, sum = 0;
```
```c
for(i=0; i<5; i++)
sum = sum + num[i];
avg = sum/5;
printf("Sum of the Elements in the array = %d", sum);
printf("Average of the elements in the array= %d", avg);
return 0;
}
```

---

**Question 7**

**Question:**

Write a program such that a Switch (case) allows to make a decision from the number of choices, i.e., from the number of cases.

**Solution:**

```c
#include<stdio.h>
int main()
{
    char ch;
    printf("Enter any character:");
    scanf("%c", &ch);
    switch(ch)
    {
        case 'R':
            printf("Red");
            break;
        case 'W':
            printf("White");
```
break;
case 'Y':
    printf("Yellow");
    break;
case 'G':
    printf("Green");
    break;
default:
    printf("Error");
    break;
}
return 0;
}

---

### Question 8

#### Question:

Write a program to find the greatest of two numbers using pointers.

#### Solution:

```c
#include<stdio.h>
int main()
{
    int x, y, *p, *q;
    printf("Enter any integer:");
    scanf("%d", &x);
    printf("Enter any integer:");
    scanf("%d", &y);
    p = &x;
```
q = &y;
if(*p>*q)
{
    printf("x is greater than y");
}
if(*q>*p)
{
    printf("y is greater than x");
}
return 0;

---

Question 9

Question:

Write a program to print the address of x and the value assigned to x.

Solution:

#include <stdio.h>
int main()
{
    int x, *p;
    x = 1;
    p = &x;
    printf("The address of the variable x =%d", p);
    printf("The value of the variable x =%d", *p);
    return 0;
}
Question 10

Question:

Write a program to print the first 10 numbers starting from one together with their squares and cubes.

Solution:

```c
#include<stdio.h>
int main()
{
    int i;
    for( i=1; i<=10; i++)
        printf("Number=%d its square=%d its cube=%d\n", i, i*i, i*i*i);
    return 0;
}
```

Question 11

Question:

Write a program:
If you enter a character M
Output must be: ch = M.
Solution:

```c
#include<stdio.h>
int main()
{
    char M;
    printf("Enter any character:");
    scanf("%c", &M);
    printf("ch=%c", M);
    return 0;
}
```

Question 12

Question:

Write a program to print the multiplication table of a number.

Solution:

```c
#include<stdio.h>
int main()
{
    int n, i;
    printf("Enter any number:");
    scanf("%d", &n);
    for( i=1; i<=5; i++)
        printf("%d * %d = %d\n", n, i, n*i);
    return 0;
}
Question 13

Question:

Write a program to print the product of the first 10 digits.

Solution:

```c
#include <stdio.h>

int main()
{
    int i, product = 1;
    for( i=1; i<=10; i++)
    product = product * i;
    printf("The product of the first 10 digits =\%d", product);
    return 0;
}
```

Question 14

Question:

Write a program to print whether the given number is positive or negative.
Solution:

```c
#include<stdio.h>
int main()
{
    int a;
    a = -35;
    if(a>0)
    {
        printf("Number is positive");
    }
    else
    {
        printf("Number entered is negative");
    }
    return 0;
}
```

Question 15

Question:

Write a program to check the equivalence of two numbers.

Solution:

```c
#include<stdio.h>
int main()
{
    int x, y;
    printf("Enter any number:");
```
```c
#include <stdio.h>

int main()
{
    int a, b, c;
    printf("Enter any number:");
    scanf("%d", &a);
    printf("Enter any number:");
    scanf("%d", &b);
    c = a%b;

    scanf("%d", &x);
    printf("Enter any number:");
    scanf("%d", &y);
    if(x-y==0)
    {
        printf("The two numbers are equivalent");
    }
    else
    {
        printf("The two numbers are not equivalent");
    }
    return 0;
}
```

**Question 16**

**Question:**

Write a program to print the remainder of two numbers.

**Solution:**

```c
#include<stdio.h>
int main()
{
    int a, b, c;
    printf("Enter any number:");
    scanf("%d", &a);
    printf("Enter any number:");
    scanf("%d", &b);
    c = a%b;
```
```c
printf("The remainder of a and b = \%d", c);
return 0;
}
```

Question 17

Question:

Write a program to print the given number is even or odd.

Solution:

```c
#include<stdio.h>
int main()
{
    int a;
    printf("Enter any number:");
    scanf("\%d", &a);
    if(a%2 == 0)
    {
        printf("The number is even");
    }
    else
    {
        printf("The number is odd");
    }
    return 0;
}
```
Question 18

Question:

Write a program to print the characters from A to Z.

Solution:

#include<stdio.h>
int main()
{
    char a;
    for( a='A'; a<='Z'; a++)
        printf("%c\n", a);
    return 0;
}

Question 19

Question:

Write a program to find the incremented and decremented values of two numbers.

Solution:

#include<stdio.h>
int main()
{
    int a, b, c, d, e, f;
    a = 10;
    b=12;
    c=a+1;
    d=b+1;
    e=a-1;
    f=b-1;
    printf("The incremented value of a =\%d", c);
    printf("The incremented value of b =\%d", d);
    printf("The decremented value of a =\%d", e);
    printf("The decremented value of b =\%d", f);
    return 0;
}

Question 20

Question:

Write a program to calculate the simple interest.

Solution:

#include<stdio.h>
int main()
{
    int P,T, R, SI;
    P = 1000;
    T = 2;
    R = 3;
SI = P*T*R/100;
printf("The simple interest = %d", SI);
return 0;
}

Question 21

Question:

Write a program to Find the largest of three numbers.

Solution:

#include<stdio.h>
int main()
{
  int a, b, c;
  printf("Enter any number:");
  scanf("%d", &a);
  printf("Enter any number:");
  scanf("%d", &b);
  printf("Enter any number:");
  scanf("%d", &c);
  if(a>b&&a>c)
  {
    printf("%d is greater than %d and %d", a, b, c);
  }
  else if (b>a&&b>c)
  {
    printf("%d is greater than %d and %d", b, a, c);
  }
else
{
printf("%d is greater than %d and %d", c, b, a);
}
return 0;
}

Question 22

Question:
Write a program to print the factorial of the entered number.

Solution:

#include<stdio.h>
int main()
{
int i, n, fact=1;
printf("Enter any number:");
scanf("%d", &n);
for(i=1; i<=n; i++)
fact = fact *i;
printf("\n Entered number is: %d", n);
printf("\n The factorial of the entered number %d is: %d", n, fact);
return 0;
}
Question 23

Question:

Write a program to print the length of the entered string.

Solution:

```c
#include <stdio.h>
#include <string.h>

int main()
{
    char ch[4];
    printf("Enter any word: ");
    scanf("%c", &ch);
    printf("The length of the string = %d", strlen(ch));
    return 0;
}
```

Question 24

Question:

Write a program to print the ASCII value of the entered character.

Solution:

```c
#include <stdio.h>
```
int main()
{
    char ch = 'A';
    printf("The ASCII value of ch is: %d", ch);
    return 0;
}

---

Question 25

Question:

Write a program to check whether the entered character is a lower case letter or not.

Solution:

#include <stdio.h>
int main()
{
    char ch = 'a';
    if(islower(ch))
        printf("you have entered the lower case letter");
    else
        printf("you have entered the upper case letter");
    return 0;
}
Question 26

Question:

Write a program to check whether the entered character is a upper case letter or not.

Solution:

#include<stdio.h>
int main()
{
    char ch = 'a';
    if(isupper(ch))
        printf("you have entered the upper case letter");
    else
        printf("you have entered the lower case letter");
    return 0;
}

Question 27

Question:

Write a program to convert the lower case letter to upper case letter.

Solution:
#include<stdio.h>
int main()
{
char ch = 'a';
char b = toupper(ch);
printf("lower case letter %c is converted to upper case letter %c", ch, b);
return 0;
}

---

**Question 28**

**Question:**

Write a program to print the output:

Einstein [0] = E  
Einstein [1] = I  
Einstein [2] = N  
Einstein [3] = S  
Einstein [5] = E  
Einstein [6] = I  
Einstein [7] = N

**Solution:**

#include<stdio.h>
int main()
{

```c
int i;
char name [8] = {'E', 'I', 'N', 'S', 'T', 'E', 'I', 'N'};
for(i=0; i<8; i++)
printf("\n Element [\%d] = %c", i, name[i]);
return 0;
}
```

---

Question 29

**Question:**

Write a program to print the output:

Name of the book = B

Price of the book = 135.00

Number of pages = 300

Edition = 8

using structures.

---

**Solution:**

```
#include<stdio.h>
int main()
{
 struct book {
 char name;
 float price;
 int pages;
 int edition;
 };
```
struct book b1;
b1.name = 'B';
b1.price = 135.00;
b1.pages = 300;
b1.edition = 8;
printf("\n Name of the book = %c", b1.name);
printf("\n Price of the book = %f", b1.price);
printf("\n Number of pages = %d", b1.pages);
printf("\n Edition of the book = %d", b1.edition);
return 0;
}

Question 30

Question:

Write a program to find square of a number using functions.

Solution:

#include<stdio.h>
int square();
int main()
{
    int answer;
    answer = square();
    printf("Square of the given number=%d", answer);
    return(0);
}
int square()
{
```c
int x;
printf("Enter any integer:");
scanf("%d", &x);
return x*x;
}
```

Question 31

Question:

Write a program To print "hello world" 10 times.

Solution:

```c
#include<stdio.h>
int main()
{
int i;
for (i =1; i<=10; i++)
printf("hello world \n");
return 0;
}
```

Question 32

Question:

Write a program to print first 5 numbers using do while loop statement.
Solution:

```c
#include <stdio.h>
int main()
{
    int i = 1;
    do
    {
        printf("%d\n", i);
        i++;
    } while (i <= 5);
    return 0;
}
```

Question 33

Question:

Write a program to print the output:

body [b] = b
body [o] = o
body [d] = d
body [y] = y

Solution:

```c
#include <stdio.h>
int main()
```
Question 34

Question:

What will be the output of the below program:

```c
#include <stdio.h>
#include <stdlib.h>
int main () {
    printf("linux\n");
    exit (0);
    printf("php\n");
    return 0;
}
```

Solution:

```
linux
```
Question 35

Question:

Write a program to check whether a character is an alphabet or not.

Solution:

```c
#include <stdio.h>
#include <ctype.h>
int main()
{
    int a = 2;
    if(isalpha(a))
    {
        printf("The character a is an alphabet");
    }
    else
    {
        printf("The character a is not an alphabet");
    }
    return 0;
}
```

Question 36

Question:
Write a program to calculate the discounted price and the total price after discount

Given:
If purchase value is greater than 1000, 10% discount
If purchase value is greater than 5000, 20% discount
If purchase value is greater than 10000, 30% discount.

Solution:

```c
#include<stdio.h>
int main()
{
    double PV;
    printf("Enter purchased value:");
    scanf("%lf", &PV);
    if(PV>1000)
    {
        printf("Discount=%lf", PV* 0.1);
        printf("Total=%lf", PV - PV* 0.1);
    }
    else if(PV>5000)
    {
        printf("Discount =%lf", PV* 0.2);
        printf("Total=%lf", PV - PV* 0.1);
    }
    else
    {
        printf("Discount=%lf", PV* 0.3);
        printf("Total=%lf", PV - PV* 0.1);
    }
    return 0;
}
```
Question 37

Question:

Write a program to print the first ten natural numbers using while loop statement.

Solution:

```c
#include<stdio.h>
int main()
{
    int i = 1;
    while (i<=10)
    {
        printf("%d\n", i++);
    }
    return 0;
}
```

Question 38

Question:

What will be the output of the below program:
#include <stdio.h>

int main()
{
    int i;
    for (i=1; i<=5; i++)
    {
        if (i==3)
        {
            continue;
        }
        printf("%d \n ", i);
    }
    return 0;
}

Solution:

1
2
4
5

Question 39

Question:

Write a program to find the size of an array.

Solution:
#include <stdio.h>
int main()
{
    int num [] = {11, 22, 33, 44, 55, 66};
    int n;

    /* Calculating the size of the array with this formula.
    * n = sizeof(array_name) / sizeof(array_name[0])
    * This is a universal formula to find number of elements in
    * an array, which means it will work for arrays of all data
    * types such as int, char, float etc.
    */
    n = sizeof(num) / sizeof(num [0]);
    printf("Size of the array is: \%d\n", n);
    return 0;
}

Question 40

Question:

What would be the output of the following programs:

#include <stdio.h>
int main()
{
    int i;
    for (i=1; i<=5; i++)
    {
        if (i==3)
        {
            90
```c
#include <stdio.h>

int main()
{
    int i;
    for(i=1;i<=5;i++)
    {
        if(i==3)
        {
            goto HAI;
        }
        printf("\n %d ",i);
    }
    HAI : printf("\n Linux");
}
```
```c
#include<stdio.h>
int main()
{
    int i = 54;
    int y = i<<1;
    printf("The value of y = %d", y);
    return 0;
}
```

**Solution:**

The value of $y = 108$

---

```c
#include<stdio.h>
int main()
{
    int i = 54;
    int y = i>>1;
    printf("The value of y = %d", y);
    return 0;
}
```

**Solution:**

The value of $y = 27$
```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int a, b;
    a = -2;
    b = abs(a);
    printf("Absolute value = %d", b);
    return 0;
}

Solution:

Absolute value = 2
```

```c
#include <stdio.h>

int main()
{
    for( ; ; )
    {
        printf("This loop will run forever.\n");
    }
    return 0;
}

Solution:

This loop will run forever.
```
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever. ........

#include <stdio.h>
int main()
{
    printf("Hello, world!");
    return 0;
    printf("Hello, world!");
}

Solution:
Hello, world!

Question 41

Question:
Write a program to check whether the person is a senior citizen or not.

Solution:
#include <stdio.h>
int main()
{
    int age;
    printf("Enter age:");
    scanf("%d", &age);
    if(age>=60)
    {
        printf("senior citizen");
    }
    else
    {
        printf("not a senior citizen");
    }
    return 0;
}

---

**Question 42**

**Question:**

Write a program to Display Fibonacci Sequence.

**Solution:**

```c
#include <stdio.h>
int main()
{
    int i, n, t1 = 0, t2 = 1, nextTerm;
    printf("Enter the number of terms: ");
    scanf("%d", &n);
    printf("Fibonacci Series: ");
```
for (i = 1; i <= n; ++i) {
    printf("%d, ", t1);
    nextTerm = t1 + t2;
    t1 = t2;
    t2 = nextTerm;
}

return 0;
}

Question 43

Question:

Write a program to Find GCD of two Numbers.

Solution:

#include <stdio.h>
int main()
{
    int n1, n2, i, gcd;

    printf("Enter two integers: ");
    scanf("%d %d", &n1, &n2);

    for(i=1; i <= n1 && i <= n2; ++i)
    {
        // Checks if i is factor of both integers
        if(n1%i==0 && n2%i==0)
            gcd = i;
    }
Question 44

Question:

Write a program to Find LCM of two Numbers.

Solution:

```c
#include <stdio.h>

int main() {
    int n1, n2, max;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);

    // maximum number between n1 and n2 is stored in min
    max = (n1 > n2) ? n1 : n2;

    while (1) {
        if (max % n1 == 0 && max % n2 == 0) {
            printf("The LCM of %d and %d is %d.", n1, n2, max);
            break;
        }
        ++max;
    }
}
```
Question 45

Question:

Write a program to Reverse a Sentence Using Recursion.

Solution:

#include <stdio.h>
void reverseSentence();
int main() {
    printf("Enter a sentence: ");
    reverseSentence();
    return 0;
}

void reverseSentence() {
    char c;
    scanf("%c", &c);
    if (c != '\n') {
        reverseSentence();
        printf("%c", c);
    }
}

return 0;
}
Question 46

Question:

Write a program to Swap Numbers in Cyclic Order Using Call by Reference.

Solution:

```c
#include <stdio.h>
void cyclicSwap(int *a, int *b, int *c);
int main() {
    int a, b, c;

    printf("Enter a, b and c respectively: ");
    scanf("%d %d %d", &a, &b, &c);

    printf("Value before swapping:\n");
    printf("a = %d \nb = %d \nc = %d\n", a, b, c);

    cyclicSwap(&a, &b, &c);

    printf("Value after swapping:\n");
    printf("a = %d \nb = %d \nc = %d\n", a, b, c);

    return 0;
}

void cyclicSwap(int *n1, int *n2, int *n3) {
    int temp;
    // swapping in cyclic order
    temp = *n2;
    *n2 = *n1;
    *n1 = *n3;
```
Question 47

Question:

Write a program to Find Largest Number Using Dynamic Memory Allocation.

Solution:

```c
#include <stdio.h>
#include <stdlib.h>

int main() {
    int num;
    float *data;
    printf("Enter the total number of elements: ");
    scanf("%d", &num);

    // Allocating memory for num elements
    data = (float *)malloc(num, sizeof(float));
    if (data == NULL) {
        printf("Error!!! memory not allocated.");
        exit(0);
    }

    // Storing numbers entered by the user.
    for (int i = 0; i < num; ++i) {
        printf("Enter Number %d: ", i + 1);
        scanf("%f", data + i);
    }
}
```
// Finding the largest number
for (int i = 1; i < num; ++i) {
    if (*data < *(data + i))
        *data = *(data + i);
}
printf("Largest number = %.2f", *data);

return 0;
}

Question 48

Question:

Write a program to Sort Elements in Lexicographical Order.

Solution:

#include <stdio.h>
#include <string.h>

int main() {
    char str[5][50], temp[50];
    printf("Enter 5 words: ");

    // Getting strings input
    for (int i = 0; i < 5; ++i) {
        fgets(str[i], sizeof(str[i]), stdin);
    }
// storing strings in the lexicographical order
for (int i = 0; i < 5; ++i) {
    for (int j = i + 1; j < 5; ++j) {

        // swapping strings if they are not in the lexicographical order
        if (strcmp(str[i], str[j]) > 0) {
            strcpy(temp, str[i]);
            strcpy(str[i], str[j]);
            strcpy(str[j], temp);
        }
    }
}

printf("\nIn the lexicographical order: \n");
for (int i = 0; i < 5; ++i) {
    fputs(str[i], stdout);
}
return 0;

---

**Question 49**

**Question:**

Write a program to Add Two Complex Numbers by Passing Structure to a Function.

**Solution:**

```
#include <stdio.h>

typedef struct complex {
```
float real;
float imag;
} complex;

complex add(complex n1, complex n2);

int main() {
    complex n1, n2, result;
    
    printf("For 1st complex number \n");
    printf("Enter the real and imaginary parts: ");
    scanf("%f %f", &n1.real, &n1.imag);
    printf("\nFor 2nd complex number \n");
    printf("Enter the real and imaginary parts: ");
    scanf("%f %f", &n2.real, &n2.imag);
    
    result = add(n1, n2);
    
    printf("Sum = %.1f + %.1fi", result.real, result.imag);
    return 0;
}

complex add(complex n1, complex n2) {
    complex temp;
    temp.real = n1.real + n2.real;
    temp.imag = n1.imag + n2.imag;
    return (temp);
}

Question 50

Question:
Write a program to Write a Sentence to a File.

Solution:

```c
#include <stdio.h>
#include <stdlib.h>

int main() {
    char sentence[1000];

    // creating file pointer to work with files
    FILE *fptr;

    // opening file in writing mode
    fptr = fopen("program.txt", "w");

    // exiting program
    if (fptr == NULL) {
        printf("Error!");
        exit(1);
    }
    printf("Enter a sentence:\n");
    fgets(sentence, sizeof(sentence), stdin);
    fprintf(fptr, "%s", sentence);
    fclose(fptr);
    return 0;
}
```

Question 51

Question:
Write a program to Read a Line From a File and Display it.

Solution:

```c
#include <stdio.h>
#include <stdlib.h> // For exit() function
int main() {
    char c[1000];
    FILE *fptr;
    if ((fptr = fopen("program.txt", "r")) == NULL) {
        printf("Error! opening file");
        // Program exits if file pointer returns NULL.
        exit(1);
    }

    // reads text until newline is encountered
    fscanf(fptr, "%[^\n]", c);
    printf("Data from the file:\n%s", c);
    fclose(fptr);

    return 0;
}
```

Question 52

Question:

Write a program to Store Data in Structures Dynamically.
Solution:

```c
#include <stdio.h>
#include <stdlib.h>

struct course {
    int marks;
    char subject[30];
};

int main() {
    struct course *ptr;
    int i, noOfRecords;
    printf("Enter the number of records: ");
    scanf("%d", &noOfRecords);

    // Memory allocation for noOfRecords structures
    ptr = (struct course *)malloc(noOfRecords * sizeof(struct course));
    for (i = 0; i < noOfRecords; ++i) {
        printf("Enter the name of the subject and marks respectively:\n");
        scanf("%s %d", (ptr + i)->subject, &(ptr + i)->marks);
    }

    printf("Displaying Information:\n");
    for (i = 0; i < noOfRecords; ++i)
        printf("%s\t%d\n", (ptr + i)->subject, (ptr + i)->marks);

    return 0;
}
```

Question 53

Question:
Write a program to Check if a Matrix is Invertible.

Solution:

```c
#include<stdio.h>

int main(){

    int a[3][3], i, j;

    long determinant;
    printf("Enter the 9 elements of matrix: ");
    for(i = 0 ;i < 3; i++)
        for(j = 0; j < 3; j++)
            scanf("%d", &a[i][j]);

    printf("\nThe matrix is\n");
    for(i = 0; i < 3; i++){
        printf("\n");
        for(j = 0; j < 3; j++)
            printf("%d\t", a[i][j]);
    }

    determinant = a[0][0] * ((a[1][1]*a[2][2]) - (a[2][1]*a[1][2])) - a[0][1] * (a[1][0] * a[2][2] - a[2][0] * a[1][2]) + a[0][2] * (a[1][0] * a[2][1] - a[2][0] * a[1][1]);
    if ( determinant == 0 )
        printf("\nMatrix is NOT invertible");
    else
        printf("\nThe given matrix has an inverse!!!");
    return 0;
}
```
Question 54

Question:

Write a program to Compute Determinant of a Matrix.

Solution:

```c
#include <stdio.h>

int main(){

    int a[3][3], i, j;

    long determinant;
    printf("Enter the 9 elements of matrix: ");
    for(i = 0; i < 3; i++)
        for(j = 0; j < 3; j++)
            scanf("%d", &a[i][j]);

    printf("nThe matrix is
");
    for(i = 0; i < 3; i++)
        printf("\n");
    for(j = 0; j < 3; j++)
        printf("%d\t", a[i][j]);
}

determinant = a[0][0] * ((a[1][1]*a[2][2]) - (a[2][1]*a[1][2])) - a[0][1] * (a[1][0] * a[2][2] - a[2][0] * a[1][2]) + a[0][2] * (a[1][0] * a[2][1] - a[2][0] * a[1][1]);

    printf("\nDeterminant of 3X3 matrix: %ld", determinant);
```
Question 55

Question:

Write a program to Implement Hash Tables.

Solution:

```c
#include <stdio.h>
#include <stdlib.h>

struct data
{
    int key;
    int value;
};

struct data *array;
int capacity = 10;
int size = 0;

/* this function gives a unique hash code to the given key */
int hashcode(int key)
{
    return (key % capacity);
}

/* it returns prime number just greater than array capacity */
```
int get_prime(int n)
{
    if (n % 2 == 0)
    {
        n++;
    }
    for (; !if_prime(n); n += 2);

    return n;
}

/* to check if given input (i.e n) is prime or not */
int if_prime(int n)
{
    int i;
    if ( n == 1 || n == 0)
    {
        return 0;
    }
    for (i = 2; i < n; i++)
    {
        if (n % i == 0)
        {
            return 0;
        }
    }
    return 1;
}

void init_array()
{
    int i;
    capacity = get_prime(capacity);
    array = (struct data*) malloc(capacity * sizeof(struct data));
    for (i = 0; i < capacity; i++)
    {
        array[i].key = 0;
    }
array[i].value = 0;
}

void insert(int key)
{
    int index = hashcode(key);
    if (array[index].value == 0)
    {
        /* key not present, insert it */
        array[index].key = key;
        array[index].value = 1;
        size++;
        printf("\n Key (%d) has been inserted \n", key);
    }
    else if(array[index].key == key)
    {
        /* updating already existing key */
        printf("\n Key (%d) already present, hence updating its value \n", key);
        array[index].value += 1;
    }
    else
    {
        /* key cannot be insert as the index is already containing some other key */
        printf("\n ELEMENT CANNOT BE INSERTED \n");
    }
}

void remove_element(int key)
{
    int index = hashcode(key);
    if(array[index].value == 0)
    {

```c
printf("\n This key does not exist \n");
}
else {
    array[index].key = 0;
    array[index].value = 0;
    size--;
    printf("\n Key (%d) has been removed \n", key);
}
}

/* to display all the elements of a hash table */
void display()
{
    int i;
    for (i = 0; i < capacity; i++)
    {
        if (array[i].value == 0)
        {
            printf("\n Array[%d] has no elements \n");
        }
        else
        {
            printf("\n array[%d] has elements -:
        key(%d) and value(%d)
        \t", i, array[i].key, array[i].value);
        }
    }
}

int size_of_hashtable()
{
    return size;
}

void main()
{
    int choice, key, value, n, c;
    clrscr();
```
init_array();

do {
    printf("\n Implementation of Hash Table in C \n\n");
    printf("MENU-
    \n1.Inserting item in the Hash Table\n2.Removing item from the Hash Table\n3.Check the size of Hash Table\n4.Display a Hash Table\n\n Please enter your choice -:");

    scanf("%d", &choice);

    switch(choice) {

    case 1:
        printf("Inserting element in Hash Table\n");
        printf("Enter key -:");
        scanf("%d", &key);
        insert(key);
        break;

    case 2:
        printf("Deleting in Hash Table \n Enter the key to delete-:");
        scanf("%d", &key);
        remove_element(key);
        break;

    case 3:
        n = size_of_hashtable();
        printf("Size of Hash Table is-:%d\n", n);
break;

case 4:

display();

break;

default:

printf("Wrong Input\n");

}

printf("\n Do you want to continue-:(press 1 for yes)\t");
scanf("%d", &c);

}while(c == 1);

getch();

}

Question 56

Question:

Write a program to Check if a Matrix is a Sparse Matrix.

Solution:
#include <stdio.h>

void main ()
{
    int matrix[10][10];
    int i, j, m, n;
    int sparse_counter = 0;

    printf("Enter the order of the matrix \n");
    scanf("%d %d", &m, &n);
    printf("Enter the elements of the matrix \n");
    for (i = 0; i < m; ++i)
    {
        for (j = 0; j < n; ++j)
        {
            scanf("%d", &matrix[i][j]);
            if (matrix[i][j] == 0)
            {
                ++sparse_counter;
            }
        }
    }
    if (sparse_counter > ((m * n) / 2))
    {
        printf("The given matrix is Sparse Matrix !!! \n");
    }
    else
    {
        printf("The given matrix is not a Sparse Matrix \n");
        printf("There are %d number of Zeros.", sparse_counter);
    }
}

Question 57

Question:
Write a program to Perform Complex Number Multiplication.

Solution:

```c
#include<stdio.h>
typedef struct COMPLEX{
    int a;
    int b;
}Complex;
Complex multiply(Complex, Complex);
int main(){
    int a1, b1, a2, b2;
    Complex x, y, z;
    printf("Enter first complex number : ");
    scanf("%d+%di", &a1, &b1);
    printf("\nEnter second complex number : ");
    scanf("%d+%di", &a2, &b2);
    x.a = a1;
    x.b = b1;
    y.a = a2;
    y.b = b2;
    z = multiply(x, y);
    printf("\nAfter multiplication: %d+%di", z.a, z.b);
    return 0;
}
Complex multiply(Complex x, Complex y){
    Complex z;
    z.a = x.a * y.a - x.b * y.b;
    z.b = x.a * y.b + x.b * y.a;
    return z;
}
```
Question 58

Question:

Write a program to Generate Random Hexadecimal Bytes.

Solution:

```c
#include <time.h>
#include <stdio.h>
#include <stdlib.h>

int main(void)
{
    int length;
    char str[] = "0123456789ABCDEF";
    /* Seed number for rand() */
    srand((unsigned int) time(0) + getpid());
    length = rand() % 15 + 8;

    while(length--){
        putchar(str[rand() % 16]);
        srand(rand());
    }
    printf("\n");

    return EXIT_SUCCESS;
}
```
C++ Exercises

Question 1

Question:

Write a program to print Hello World!.

Solution:

```cpp
#include <iostream>

int main()
{
    std::cout << "Hello World!";
    return 0;
}
```

Solution:

```cpp
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello World!";
    return 0;
}
```
Question 2

Question:

Write a program to find the area of a circle.

Solution:

```cpp
#include<iostream>
using namespace std;
int main()
{
    float r, area;
    cout<<"Enter any number:";
    cin>>r;
    area = 3.14 * r * r;
    cout<<"The area of the circle = "<< area;
    return 0;
}
```

Question 3

Question:

Write a program to find the sum of two numbers.
**Solution:**

```cpp
#include<iostream>
using namespace std;
int main()
{
  float a, b, sum;
  cout<<"Enter any two numbers:";
  cin>>a;
  cin>>b;
  sum = a + b;
  cout<<"The sum of a and b = "<< sum;
  return 0;
}
```

**Question 4**

**Question:**

Write a program to find the square of a number.

**Solution:**

```cpp
#include<iostream>
using namespace std;
int main()
{
  int a, b;
  a = 2;
  b = a * a;
  cout<<"The square of a = "<< b;
```
return 0;
}

Question 5

Question:

Write a program to find the greatest of two numbers.

Solution:

#include <iostream>
using namespace std;
int main()
{
    int a, b;
    a = 2;
    b = 3;
    if(a>b)
    {
        cout<<"a is greater than b";
    }
    else
    {
        cout<<"b is greater than a";
    }
    return 0;
}
Question 6

Question:

Write a program to print the average of the elements in the array.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int i, avg, sum = 0;
    for(i=0; i<5; i++)
        sum = sum + num[i];
    avg = sum/5;
    cout<<"Sum of the Elements in the array = "<< sum <<endl;
    cout<<"Average of the elements in the array= "<< avg<<endl;
    return 0;
}
```

Question 7

Question:

Write a program such that a Switch (case) allows to make a decision from the number of choices, i.e., from the number of cases.
Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    char ch;
    cout << "Enter any character:";  
    cin >> ch;
    switch (ch)
    {
    case 'R':
        cout << "Red";
        break;
    case 'W':
        cout << "White";
        break;
    case 'Y':
        cout << "Yellow";
        break;
    case 'G':
        cout << "Green";
        break;
    default:
        cout << "Error";
        break;
    }
    return 0;
}
```
**Question 8**

**Question:**

Write a program to find the greatest of two numbers using pointers.

**Solution:**

```cpp
#include<iostream>
using namespace std;
int main()
{
    int x, y, *p, *q;
    cout<<"Enter any integer:";
    cin>> x;
    cout<<"Enter any integer:";
    cin>> y;
    p = &x;
    q = &y;
    if(*p>*q)
    {
        cout<<"x is greater than y";
    }
    else
    {
        cout<<"y is greater than x";
    }
    return 0;
}
```
Question 9

Question:

Write a program to print the address of x and the value assigned to x.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int x, *p;
    cout << "Enter any integer:";
    cin >> x;
    p = &x;
    cout << "The address of the variable x = " << p << endl;
    cout << "The value of the variable x = " << *p << endl;
    return 0;
}
```

Question 10

Question:

Write a program to print the first 10 numbers starting from one together with their squares and cubes.
Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int i;
    for (i = 1; i <= 10; i++)
        cout << "number = " << i << " its square = " << i*i << " its cube = " << i*i*i << endl;
    return 0;
}
```

Question 11

Question:

Write a program:
If you enter a character M
Output must be: ch = M.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    char M;
    cout << "Enter any character: ";
    cin >> M;
    cout << "ch = " << M;
```
Question 12

Question:

Write a program to print the multiplication table of a number.

Solution:

```cpp
#include<iostream>
using namespace std;
int main()
{
   int n, i;
   cout<<"Enter any number:";  
cin>>n;
   for( i=1; i<=5; i++)
      cout<< n << " * " << i << " = " << n*i <<endl;
   return 0;
}
```

Question 13

Question:

Write a program to print the product of the first 10 digits.
Solution:

```cpp
#include<iostream>
using namespace std;
int main()
{
    int i, product = 1;
    for( i=1; i<=10; i++)
        product = product * i;
    cout<<"The product of the first 10 digits = " << product;
    return 0;
}
```

Question 14

Question:

Write a program to print whether the given number is positive or negative.

Solution:

```cpp
#include<iostream>
using namespace std;
int main()
{
    int a;
    a = -35;
    if(a>0)
    {
```
cout<<"Number is positive";
}
else
{
cout<<"Number entered is negative";
}
return 0;
}

Question 15

Question:
Write a program to check the equivalence of two numbers.

Solution:

#include<iostream>
using namespace std;
int main()
{
 int x, y;
cout<<"Enter any number:"
; cin>>x;
cout<<"Enter any number:"
; cin>>y;
if(x-y==0)
{
cout<<"The two numbers are equivalent";
}
else
Question 16

Question:

Write a program to print the remainder of two numbers.

Solution:

```cpp
#include<iostream>
using namespace std;
int main()
{
    int a, b, c;
    cout<<"Enter any number:";
    cin>>a;
    cout<<"Enter any number:";
    cin>>b;
    c = a % b;
    cout<<"The remainder of a and b = " << c;
    return 0;
}
```
Question 17

Question:

Write a program to print the given number is even or odd.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int a;
    cout<<"Enter any number:";
    cin>>a;
    if(a%2 == 0)
    {
        cout<<"The number is even";
    }
    else
    {
        cout<<"The number is odd";
    }
    return 0;
}
```

Question 18

Question:
Write a program to print the characters from A to Z.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    char a = 'A';
    while (a<='Z')
    {
        cout<< "\n"<< a++;
    }
    return 0;
}
```

Question 19

Question:

Write a program to find the incremented and decremented values of two numbers.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
```
```cpp
{ 
    int a, b, c, d, e, f;
    a = 10;
    b=12;
    c=a+1;
    d=b+1;
    e=a-1;
    f=b-1;
    cout<<"The incremented value of a = "<< c << endl;
    cout<<"The incremented value of b = "<< d << endl;
    cout<<"The decremented value of a = "<< e << endl;
    cout<<"The decremented value of b = "<< f << endl;
    return 0;
}

Question 20

Question:

Write a program to calculate the simple interest.

Solution:

#include<iostream>
using namespace std;
int main()
{
    int P,T, R, SI;
    cout<<"Enter principal amount:"
    cin>>P;
    cout<<"Enter time:"
```
cin>>T;
cout<<"Enter rate of interest:"

cin>>R;
SI = P*T*R/100;
cout<<"the simple interest = "<<SI;
return 0;
}

Question 21

Question:

Write a program to Find the largest of three numbers.

Solution:

#include<iostream>
using namespace std;

int main()
{
    int a, b, c;
    cout<<"Enter any number:"
    cin>>a;
    cout<<"Enter any number:"
    cin>>b;
    cout<<"Enter any number:"
    cin>>c;
    if(a>b&&a>c)
    {
        cout<< a<<" is greater than "<< b<<" and "<<c;
    }
else if (b>a&&b>c)
{
    cout<< b" is greater than " a " and " c;
}
else
{
    cout<< c" is greater than " b " and " a;
}
return 0;

---

**Question 22**

**Question:**

Write a program to print the factorial of the entered number.

**Solution:**

```cpp
#include<iosstream>
using namespace std;
int main()
{
    int i, n, fact=1;
    cout<<"Enter any number:";
    cin>>n;
    for(i=1; i<=n; i++)
        fact = fact * i;
    cout<<"\n Entered number is: " n;
    cout<<"\n The factorial of the entered number " n " is: " fact;
    return 0;
```
Question 23

Question:

Write a program to print the length of the entered string.

Solution:

```cpp
#include <iostream>
#include <string.h>
using namespace std;

int main()
{
    char ch[4];
    cout<<"Enter any word: ";
    cin>>ch;
    cout<<"The length of the string = "<< strlen(ch);
    return 0;
}
```

Question 24

Question:

Write a program to print the ASCII value of the entered character.
**Solution:**

```cpp
#include <iostream>
using namespace std;
int main()
{
    char c;
    cout << "Enter a character: ";
    cin >> c;
    cout << "ASCII Value of " << c << " is " << int(c);
    return 0;
}
```

**Question 25**

**Question:**

Write a program to check whether the entered character is a lower case letter or not.

**Solution:**

```cpp
#include <iostream>
using namespace std;
int main()
{
    char ch = 'a';
    if(islower(ch))
        cout<<"you have entered the lower case letter";
```
else
    cout<<"you have entered the upper case letter";
    return 0;
}

Question 26

Question:

Write a program to check whether the entered character is a upper case letter or not.

Solution:

#include<iostream>
using namespace std;
int main()
{
    char ch = 'a';
    if(isupper(ch))
        cout<<"you have entered the upper case letter";
    else
        cout<<"you have entered the lower case letter";
    return 0;
}
Question 27

Question:

Write a program to convert the lower case letter to upper case letter.

Solution:

```cpp
#include<iostream>
using namespace std;

int main()
{
    char ch = 'a';
    char b = toupper(ch);
    cout<<" lower case letter "<<ch<<" is converted to upper case letter "<<b;
    return 0;
}
```

Question 28

Question:

Write a program to print the output:

Einstein [0] = E
Einstein [1] = I
Einstein [2] = N
Einstein [3] = S
Einstein [5] = E
Einstein [6] = I
Einstein [7] = N

Solution:

```cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    char name[8] = {'E', 'I', 'N', 'S', 'T', 'E', 'I', 'N'};
    for (i = 0; i < 8; i++)
        cout << "Element [" << i << "] = " << name[i] << endl;
    return 0;
}
```

Question 29

Question:

Write a program to print the output:
Name of the book = B
Price of the book = 135.00
Number of pages = 300
Edition = 8
using structures.
Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    struct book {
        char name;
        float price;
        int pages;
        int edition;
    };
    struct book b1 = {'B', 135.00, 300, 8};
    cout << "Name of the book = " << b1.name << endl;
    cout << "Price of the book = " << b1.price << endl;
    cout << "Number of pages = " << b1.pages << endl;
    return 0;
}
```

Question 30

Question:

Write a program to find square of a number using functions.

Solution:

```cpp
#include <iostream>
```
using namespace std;
int square();
int main()
{
    int answer;
    answer = square();
    cout<<"Square of the given number = "<< answer;
    return 0;
}
int square()
{
    int x;
    cout<<"Enter any integer:"
    cin>>x;
    return x*x;
}

Question 31

Question:
Write a program To print "hello world" 10 times.

Solution:
#include<iostream>
using namespace std;
int main()
{
    int i;
    for (i =1; i<=10; i ++)
cout<<"\n hello world";
 return 0;
}

Question 32

Question:

Write a program to print first 5 numbers using do while loop statement.

Solution:

#include<iostream>
using namespace std;
int main()
{
    int i =1;
    do
    {
        cout<< " \n i= " << i++;
    } while (i<=5);
    return 0;
}

Question 33

Question:
Write a program to print the output:

body [b] = b
body [o] = o
body [d] = d
body [y] = y

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    char i;
    char body [4] = {'b', 'o', 'd', 'y'};
    for(i=0; i<4; i++)
        cout << "body [" << body[i] << "] = " << body[i] << endl;
    return 0;
}
```

Question 34

Question:

What will be the output of the below program:

```cpp
#include <iostream>
using namespace std;
```
int main()
{
    cout<<"linux\n";
    exit (0);
    cout<<"php\n";
    return 0;
}

Solution:

linux

Question 35

Question:

Write a program to check whether a character is an alphabet or not.

Solution:

#include <iostream>
using namespace std;
int main()
{
    int a =2;
    if(isalpha(a))
    {
        cout<<"The character a is an alphabet";
    }
}
else
{
cout<<"The character a is not an alphabet";
}
return 0;
}

Question 36

Question:

Write a program to calculate the discounted price and the total price after discount

Given:
If purchase value is greater than 1000, 10% discount
If purchase value is greater than 5000, 20% discount
If purchase value is greater than 10000, 30% discount.

Solution:

#include<iostream>
using namespace std;
int main()
{
double PV;
cout<<"Enter purchased value:";
cin>>PV;
if(PV>1000)
{


cout<<"Discount = "<< PV* 0.1 << endl;
cout<<"Total= "<< PV - PV* 0.1 << endl;
}
else if(PV>5000)
{
cout<<"Discount = "<< PV* 0.2 << endl;
cout<<"Total= "<< PV - PV* 0.1 << endl;
}
else
{
cout<<"Discount = "<< PV* 0.3 << endl;
cout<<"Total= "<< PV - PV* 0.1 << endl;
}
return 0;
}

Question 37

Question:

Write a program to print the first ten natural numbers using while loop statement.

Solution:

#include<iostream>
using namespace std;
int main()
{
    int i = 1;
    while (i<=10)
    {
        cout<<i<<" ";
    }
```cpp
#include <iostream>
using namespace std;
int main()
{
  int i;
  for (i=1; i<=5; i++)
  {
    if (i==3)
    {
      continue;
    }
    cout<<"\n "<< i;
  }
  return 0;
}
```

**Question 38**

**Question:**

What will be the output of the below program:

```
#include <iostream>
using namespace std;
int main()
{
  int i;
  for (i=1; i<=5; i++)
  {
    if (i==3)
    {
      continue;
    }
    cout<<"\n "<< i;
  }
  return 0;
}
```

**Solution:**

148
Question 39

Question:

Write a program to find the size of an array.

Solution:

```cpp
#include <iostream>
using namespace std;
int main()
{
    int num[] = {11, 22, 33, 44, 55, 66};
    int n;

    // Calculating the size of the array with this formula.
    // n = sizeof(array_name) / sizeof(array_name[0])
    // This is a universal formula to find number of elements in
    // an array, which means it will work for arrays of all data
    // types such as int, char, float etc.
    n = sizeof(num) / sizeof(num[0]);
    cout << "Size of the array is:" << n;
    return 0;
}
```
Question 40

Question:

What would be the output of the following programs:

```cpp
#include <iostream>
using namespace std;
int main()
{
  int i;
  for (i=1; i<=5; i++)
  {
    if (i==3)
      break;
  }
  cout<<"\n "<< i;
  return 0;
}
```

Solution:

```
1
2
```
```cpp
#include <iostream>
using namespace std;

int main()
{
    int i;
    for (i = 1; i <= 5; i++)
    {
        if (i == 3)
        {
            goto HAI;
        }
        cout << "n " << i;
    }
    HAI : cout << "n Linux";
}
```

**Solution:**

```
1
2
Linux
```

```cpp
#include <iostream>
using namespace std;

int main()
{
    int i = 54;
    int y = i << 1;
    cout << "The value of y = " << y;
    return 0;
}
```
Solution:

The value of y = \textbf{108}

```cpp
#include<iostream>
using namespace std;

int main()
{
    int i = 54;
    int y = i>>1;
    cout<<"The value of y = " << y;
    return 0;
}
```

Solution:

The value of y = \textbf{27}

```cpp
#include<iostream>
#include<cmath>
using namespace std;

int main()
{
    int a, b;
    a= -2;
    b= abs(a);
    cout<<" Absolute value = " << b << endl;
    return 0;
}
```
Solution:

Absolute value = 2

#include <iostream>
using namespace std;
int main()
{
    for( ; ; ) {
        cout<<"This loop will run forever.\n";
    }
    return 0;
}

Solution:

This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever. ........

#include<iostream>
using namespace std;
int main()
Solution:

Hello, world!

Question 41

Question:

Write a program to check whether the person is a senior citizen or not.

Solution:

```cpp
#include <iostream>
using namespace std;

int main()
{
    int age;
    age = 20;
    if (age >= 60)
    {
        cout << "Senior citizen";
    }
    if (age < 60)
    {
        cout << "Not a senior citizen";
    }
    return 0;
}
```
Question 42

Question:

Write a program to compute inverse of tan x.

Solution:

```cpp
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
    int x = 20;
    cout << "Inverse of tan x = " << atan(x);
    return 0;
}
```

Question 43

Question:
Write a program to Find All Roots of a Quadratic Equation.

Solution:

```cpp
#include <iostream>
#include <cmath>
using namespace std;

int main() {

    float a, b, c, x1, x2, discriminant, realPart, imaginaryPart;
    cout << "Enter coefficients a, b and c: ";
    cin >> a >> b >> c;
    discriminant = b*b - 4*a*c;

    if (discriminant > 0) {
        x1 = (-b + sqrt(discriminant)) / (2*a);
        x2 = (-b - sqrt(discriminant)) / (2*a);
        cout << "Roots are real and different." << endl;
        cout << "x1 = " << x1 << endl;
        cout << "x2 = " << x2 << endl;
    }

    else if (discriminant == 0) {
        cout << "Roots are real and same." << endl;
        x1 = -b/(2*a);
        cout << "x1 = x2 =" << x1 << endl;
    }

    else {
        realPart = -b/(2*a);
        imaginaryPart = sqrt(-discriminant)/(2*a);
        cout << "Roots are complex and different." << endl;
        cout << "x1 = " << realPart << "+" << imaginaryPart << "i" << endl;
    }
}
```
cout << "x2 = " << realPart << "-" << imaginaryPart << "i" << endl;
}

return 0;
}

---

**Question 44**

**Question:**

Write a program to Display Fibonacci Series.

**Solution:**

```cpp
#include <iostream>
using namespace std;

int main()
{
    int n, t1 = 0, t2 = 1, nextTerm = 0;

    cout << "Enter the number of terms: ";
    cin >> n;

    cout << "Fibonacci Series: ";

    for (int i = 1; i <= n; ++i)
    {
        // Prints the first two terms.
        if(i == 1)
        {
        
```
cout << " " << t1;
    continue;
}
if(i == 2)
{
    cout << t2 << " ";
    continue;
}
nextTerm = t1 + t2;
t1 = t2;
t2 = nextTerm;
    cout << nextTerm << " ";
}  
return 0;
}

Question 45

Question:
Write a program to Find GCD.

Solution:

#include <iostream>
using namespace std;

int main()
{
    int n1, n2;
cout << "Enter two numbers: ";
cin >> n1 >> n2;

while(n1 != n2)
{
    if(n1 > n2)
        n1 -= n2;
    else
        n2 -= n1;
}

cout << "HCF = " << n1;
return 0;

Question 46

Question:

Write a program to Find LCM.

Solution:

#include <iostream>
using namespace std;

int main()
{
    int n1, n2, max;

cout << "Enter two numbers: ";
cin >> n1 >> n2;

// maximum value between n1 and n2 is stored in max
max = (n1 > n2) ? n1 : n2;

do
{
    if (max % n1 == 0 && max % n2 == 0)
    {
        cout << "LCM = " << max;
        break;
    }
    else
        ++max;
} while (true);

return 0;
```cpp
int main() {
    int low, high, i;
    bool isPrime = true;

    cout << "Enter two numbers (intervals): ";
    cin >> low >> high;

    cout << "\nPrime numbers between " << low << " and " << high << " are: " << endl;

    while (low < high) {
        isPrime = true;
        if (low == 0 || low == 1) {
            isPrime = false;
        } else {
            for (i = 2; i <= low / 2; ++i) {
                if (low % i == 0) {
                    isPrime = false;
                    break;
                }
            }
        }

        if (isPrime)
            cout << low << " ";

        ++low;
    }

    return 0;
}
```
Question 48

Question:

Write a program to Check Armstrong Number.

Solution:

```cpp
#include <iostream>
using namespace std;

int main() {
    int num, originalNum, remainder, result = 0;
    cout << "Enter a three-digit integer: ";
    cin >> num;
    originalNum = num;

    while (originalNum != 0) {
        // remainder contains the last digit
        remainder = originalNum % 10;

        result += remainder * remainder * remainder;

        // removing last digit from the original number
        originalNum /= 10;
    }

    if (result == num)
        cout << num << " is an Armstrong number.";
    else
        cout << num << " is not an Armstrong number.";

    return 0;
}
```
Question 49

Question:

*Write a program to print half pyramid using .

Solution:

```cpp
#include <iostream>
using namespace std;

int main()
{
    int rows;

    cout << "Enter number of rows: ";
    cin >> rows;

    for(int i = 1; i <= rows; ++i)
    {
        for(int j = 1; j <= i; ++j)
        {
            cout << "* ";
        }
        cout << "\n";
    }
    return 0;
}
```
Question 50

Question:

Write a program to print half pyramid using numbers.

Solution:

```cpp
#include <iostream>
using namespace std;

int main()
{
    int rows;

    cout << "Enter number of rows: ";
    cin >> rows;

    for(int i = 1; i <= rows; ++i)
    {
        for(int j = 1; j <= i; ++j)
        {
            cout << j << " ";
        }
        cout << "\n";
    }
    return 0;
}
```
Question 51

Question:

Write a program to Find Sum of Natural Numbers using Recursion.

Solution:

```cpp
#include <iostream>
using namespace std;

int add(int n);

int main()
{
    int n;

    cout << "Enter a positive integer: ";
    cin >> n;

    cout << "Sum = " << add(n);

    return 0;
}

int add(int n)
{
    if(n != 0)
        return n + add(n - 1);
    return 0;
}
```
Question 52

Question:

Write a program to Access Elements of an Array Using Pointer.

Solution:

```cpp
#include <iostream>
using namespace std;

int main()
{
    int data[5];
    cout << "Enter elements: ";
    
    for(int i = 0; i < 5; ++i)
        cin >> data[i];

    cout << "You entered: ";
    for(int i = 0; i < 5; ++i)
        cout << endl <<*(data + i);

    return 0;
}
```

Question 53

Question:
Write a program to Store Information of a Student in a Structure.

Solution:

```cpp
#include <iostream>
using namespace std;

struct student
{
    char name[50];
    int roll;
    float marks;
};

int main()
{
    student s;
    cout << "Enter information," << endl;
    cout << "Enter name: ";
    cin >> s.name;
    cout << "Enter roll number: ";
    cin >> s.roll;
    cout << "Enter marks: ";
    cin >> s.marks;

    cout << "Displaying Information," << endl;
    cout << "Name: " << s.name << endl;
    cout << "Roll: " << s.roll << endl;
    cout << "Marks: " << s.marks << endl;
    return 0;
}
```
Question 54

Question:

Write a program to Store and Display Information Using Structure.

Solution:

```cpp
#include <iostream>
using namespace std;

struct student
{
    char name[50];
    int roll;
    float marks;
} s[10];

int main()
{
    cout << "Enter information of students: " << endl;

    // storing information
    for(int i = 0; i < 10; ++i)
    {
        s[i].roll = i+1;
        cout << "For roll number" << s[i].roll << "," << endl;

        cout << "Enter name: ";
        cin >> s[i].name;

        cout << "Enter marks: ";
        cin >> s[i].marks;
    }
}
```
cout << endl;
}

cout << "Displaying Information: " << endl;

// Displaying information
for(int i = 0; i < 10; ++i)
{
    cout << "\nRoll number: " << i+1 << endl;
    cout << "Name: " << s[i].name << endl;
    cout << "Marks: " << s[i].marks << endl;
}

return 0;
}
} complexNumber;

complexNumber addComplexNumbers(complex, complex);

int main() {
    complexNumber num1, num2, complexSum;
    char signOfImag;

    cout << "For 1st complex number," << endl;
    cout << "Enter real and imaginary parts respectively:" << endl;
    cin >> num1.real >> num1.imag;

    cout << endl
    << "For 2nd complex number," << endl;
    cout << "Enter real and imaginary parts respectively:" << endl;
    cin >> num2.real >> num2.imag;

    // Call add function and store result in complexSum
    complexSum = addComplexNumbers(num1, num2);

    // Use Ternary Operator to check the sign of the imaginary number
    signOfImag = (complexSum.imag > 0) ? '+' : '-';

    // Use Ternary Operator to adjust the sign of the imaginary number
    complexSum.imag = (complexSum.imag > 0) ? complexSum.imag : -complexSum.imag;

    cout << "Sum = " << complexSum.real << signOfImag << complexSum.imag << "i";

    return 0;
}

complexNumber addComplexNumbers(complex num1, complex num2) {
    complex temp;
    temp.real = num1.real + num2.real;
    temp.imag = num1.imag + num2.imag;
    return (temp);
}
Java Exercises

Question 1

Question:

Write a program to print Hello World!.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

Question 2

Question:

Write a program to find the area of a circle.

Solution:
public class MyClass {
    public static void main (String [] args) {
        int r, area;
        r = 2;
        area = 3.14 * r * r;
        System.out.println("The area of the circle = " + area);
    }
}

Question 3

Question:

Write a program to find the sum of two numbers.

Solution:

public class MyClass {
    public static void main(String [] args) {
        int a, b, sum;
        a=1;
        b=2;
        sum = a + b;
        System.out.println("The sum of a and b = " + sum);
    }
}
Question 4

Question:

Write a program to find the square of a number.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int a, b;
        a = 2;
        b = a * a;
        System.out.println("The square of a = " + b);
    }
}
```

Question 5

Question:

Write a program to find the greatest of two numbers.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int a, b;
```
Question 6

Question:

Write a program to print the average of the elements in the array.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int i, avg, sum = 0;
        int[] num = {16, 18, 20, 25, 36};
        for (i = 0; i < 5; i++)
            sum = sum + num[i];
        avg = sum / 5;
        System.out.println("Sum of the Elements in the array = " + sum);
        System.out.println("Average of the Elements in the array = " + avg);
    }
}
Question 7

Question:

Write a program such that a Switch (case) allows to make a decision from the number of choices, i.e., from the number of cases.

Solution:

```java
class MyClass {
    public static void main(String[] args) throws Exception {
        char ch;
        System.out.print("Enter a character:");
        ch = (char)System.in.read();
        switch(ch) {
            case 'R':
                System.out.print("Red");
                break;
            case 'W':
                System.out.print("White");
                break;
            case 'Y':
                System.out.print("Yellow");
                break;
            case 'G':
                System.out.print("Green");
                break;
            default:
```
System.out.print("Error");
break;
}
}

Question 8

Question:

Write a program to read 10 numbers from the keyboard and find their sum and average.

Solution:

import java.util.Scanner;
public class MyClass {
    public static void main(String [] args) {
        int N1, N2, N3, N4, N5, N6, N7, N8, N9, N10, sum;
        float X;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter any ten Numbers: ");
        N1 = scan.nextInt();
        N2 = scan.nextInt();
        N3 = scan.nextInt();
        N4 = scan.nextInt();
        N5 = scan.nextInt();
        N6 = scan.nextInt();
        N7 = scan.nextInt();
        N8 = scan.nextInt();
        N9 = scan.nextInt();
        N10 = scan.nextInt();
        }
\[ N10 = \text{scan.nextInt();} \]
\[ \text{sum} = N1 + N2 + N3 + N4 + N5 + N6 + N7 + N8 + N9 + N10; \]
\[ X = \text{sum} / 10; \]
System.out.println("The sum of 10 numbers = " + sum);
System.out.println("The average of 10 numbers = " + X);
}
}

---

**Question 9**

**Question:**

Write a program to print the first 10 numbers starting from one together with their squares and cubes.

**Solution:**

```java
public class MyClass {
    public static void main(String[] args) throws Exception {
        int i;
        for (i=1; i<=10; i++)
            System.out.println(" \n number = " + i + " its square = " + i*i + " its cube = " + i*i*i);
    }
}
```
Question 10

Question:

Write a program:
If you enter a character M
Output must be: ch = M.

Solution:

```java
public class MyClass {
    public static void main(String[] args) throws Exception {
        char c;
        System.out.print("Enter a character:");
        c = (char) System.in.read();
        System.out.println("ch = "+ c);
    }
}
```

Question 11

Question:

Write a program to print the multiplication table of a number.

Solution:
import java.util.Scanner;

public class MyClass {
    public static void main(String [] args) {
        int n, i;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number: ");
        n = scan.nextInt();
        for (i=1; i<=5; i++)
            System.out.println (n + " * " + i + " = " + n * i);
    }
}

Question 12

Question:
Write a program to print the product of the first 10 digits.

Solution:

public class MyClass {
    public static void main(String [] args) {
        int i, product = 1;
        for (i=1; i<=10; i++)
            product = product * i;
        System.out.println("The product of the first 10 digits = " + product);
    }
}
Question 13

Question:

Write a program to print whether the given number is positive or negative.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int a;
        a = -35;
        if(a>0) {
            System.out.println("Number is positive");
        } else {
            System.out.println("Number entered is negative");
        }
    }
}
```

Question 14

Question:

Write a program to check the equivalence of two numbers.
Solution:

```java
import java.util.Scanner;
public class MyClass {
    public static void main(String[] args) {
        int x, y;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number: ");
        x = scan.nextInt();
        System.out.println("Enter a number: ");
        y = scan.nextInt();
        if (x - y == 0) {
            System.out.println("The two numbers are equivalent");
        } else {
            System.out.println("The two numbers are not equivalent");
        }
    }
}
```

Question 15

Question:

Write a program to print the remainder of two numbers.

Solution:

```java
import java.util.Scanner;
```
public class MyClass {
    public static void main(String [] args) {
        int a, b, c;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number: ");
        a = scan.nextInt();
        System.out.println("Enter a number: ");
        b = scan.nextInt();
        c = a % b;
        System.out.println("The remainder of a and b = " + c);
    }
}

Question 16

Question:

Write a program to print the given number is even or odd.

Solution:

import java.util.Scanner;
public class MyClass {
    public static void main(String [] args) {
        int a;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter a number: ");
        a = scan.nextInt();
        if(a%2 == 0)
        {
            System.out.println("The number is even");
        }
{
    else
    {
        System.out.println("The number is odd");
    }
}

Question 17

Question:

Write a program to print the characters from A to Z.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        char a;
        for (a = 'A'; a <= 'Z'; a++)
            System.out.println("\n " + a);
    }
}
```

Question 18

Question:
Write a program to find the incremented and decremented values of two numbers.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int a, b, c, d, e, f;
        a = 10;
        b = 12;
        c = a + 1;
        d = b + 1;
        e = a - 1;
        f = b - 1;
        System.out.print("The incremented value of a = "+ c);
        System.out.print("The incremented value of b = "+ d);
        System.out.print("The decremented value of a = "+ e);
        System.out.print("The decremented value of b = "+ f);
    }
}
```

Question 19

Question:

Write a program to calculate the simple interest.

Solution:
public class MyClass {
public static void main(String [] args) {
    int P,T, R, SI;
P = 1000;
T = 2;
R = 3;
SI = P*T*R/100;
System.out.println("The simple interest = " + SI);
}
}

Question 20

Question:

Write a program to Find the largest of three numbers.

Solution:

import java.util.Scanner;
public class MyClass {
public static void main(String [] args) {
    int a, b, c;
Scanner scan = new Scanner(System.in);
System.out.println("Enter any number:");
a = scan.nextInt();
System.out.println("Enter any number:");
b = scan.nextInt();
System.out.println("Enter any number:");
c = scan.nextInt();
if(a>b&&a>c)
```java
{ System.out.println("a is greater than b and c"); } else if (b>a&&b>c) { System.out.println("b is greater than a and c"); } else { System.out.println("c is greater than b and a"); }
}
```

**Question 21**

**Question:**

Write a program to print the factorial of the entered number.

**Solution:**

```java
import java.util.Scanner;
public class MyClass {
    public static void main(String []args){
        int i, n, fact=1;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter any number:");
        n = scan.nextInt();
        for(i = 1; i <= n; i++)
        {
```
import java.util.Scanner;
public class MyClass {
    public static void main(String[] args) {
        String a;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Your Name : ");
        a = scan.nextLine();
        System.out.println("The length of the String is: "+ a.length());
    }
}

Question 22

Question:
Write a program to print the length of the entered string.

Solution:

import java.util.Scanner;
public class MyClass {
    public static void main(String[] args) {
        String a;
        Scanner scan = new Scanner(System.in);
        System.out.print("Enter Your Name : ");
        a = scan.nextLine();
        System.out.println("The length of the String is: "+ a.length());
    }
}

Question 23

Question:
Write a program to print the output:

Einstein [0] = E
Einstein [1] = I
Einstein [2] = N
Einstein [3] = S
Einstein [5] = E
Einstein [6] = I
Einstein [7] = N

Solution:

```java
public class MyClass {
    public static void main(String[] args) throws Exception{
        int i;
        char [] num = {'E', 'I', 'N', 'S', 'T', 'E', 'I', 'N'};
        for(i=0; i<8; i++)
            System.out.println("Einstein [" + i + "] = " + num[i]);
    }
}
```

Question 24

Question:

Write a program to find square of a number using method.
Solution:

```java
import java.util.Scanner;

public class MyClass {
    public static void main(String[] args) {
        int x;
        Scanner scan = new Scanner(System.in);
        System.out.println("Enter any number: ");
        x = scan.nextInt();
        System.out.println("Square of the number = " + square(x));
    }
    public static int square(int x){
        return x*x;
    }
}
```

Question 25

Question:

Write a program To print "hello world" 10 times.

Solution:

```java
public class MyClass {
    public static void main(String [] args) {
        int i;
        for (i =1; i<=10; i++)
            System.out.println("\n hello world");
    }
}
```
Question 26

Question:

Write a program to print first 5 numbers using do while loop statement.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int i = 1;
        do {
            System.out.println("n "+ i);
        } while (i<=5);
    }
}
```

Question 27

Question:

Write a program to print the output:

```java
body [b] = b
body [o] = o
```
body [d] = d
body [y] = y

Solution:

```java
public class MyClass {
    public static void main(String[] args) throws Exception {
        int i;
        char[] body = {'b', 'o', 'd', 'y'};
        for (i = 0; i < 4; i++)
            System.out.println("body [" + body[i] + "] = " + body[i]);
    }
}
```

Question 28

Question:

Write a program to print the first ten natural numbers using while loop statement.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int i = 1;
        while (i <= 10)
            System.out.println(\"n \" + i++);
    }
```
Question 29

Question:
What will be the output of the below program:

```java
public class MyClass {
    public static void main(String []args) {
        int i;
        for (i=1; i<=5; i++) {
            if (i==3) {
                continue;
            }
            System.out.println("" + i);
        }
    }
}
```

Solution:

1
2
4
5
Question 30

Question:

Write a program to find the size of an array.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int num[] = {11, 22, 33, 44, 55, 66};
        System.out.println("Size of the array is: " + num.length);
    }
}
```

Question 31

Question:

What would be the output of the following programs:

```java
public class MyClass {
    public static void main(String []args) {
        int i;
        for (i=1; i<=5; i++) {
```
```java
if (i==3) {
    break;
}
System.out.println("" + i);
}
}

Solution:

1
2

public class MyClass {
    public static void main(String [] args) {
        int x = 2;
        System.out.println(" Square of a number = " + Math.pow(x, 2));
    }
}

Solution:

Square of a number = 4.0

public class MyClass {
    public static void main(String [] args) {
        int i = 54;
        int y = i<<1;
    }
}
System.out.println("The value of y = " + y);
}
}

Solution:
The value of y = 108

```java
public class MyClass {
    public static void main(String [] args) {
        int i = 54;
        int y = i>>1;
        System.out.println("The value of y = " + y);
    }
}
```

Solution:
The value of y = 27

```java
import java.util.Scanner;
public class MyClass {
    public static void main(String [] args) {
        String m;
        Scanner in = new Scanner(System.in);
        System.out.print("Enter the name: ");
        m = in.nextLine();
        System.out.println("The name you entered = " + m);
    }
}
```
Solution:

Enter the name:
Dennis
The name you entered = Dennis

```java
public class MyClass {
    public static void main(String[] args) {
        for( ; ; ) {
            System.out.println("This loop will run forever.\n");
        }
    }
}
```

Solution:

This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever.
This loop will run forever. .........

```java
public class MyClass {
```
public static void main(String[] args) {
    System.out.println("Hello, World!");
    System.exit(0);
    System.out.println("Hello, World!");
}
Question 33

Question:

Write a program to Find ASCII Value of a character.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {

        char ch = 'a';
        int ascii = ch;
        // You can also cast char to int
        int castAscii = (int) ch;

        System.out.println("The ASCII value of " + ch + " is: " + ascii);
        System.out.println("The ASCII value of " + ch + " is: " + castAscii);
    }
}
```
Question 34

Question:

Write a program to Swap Two Numbers.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {

        float first = 12.0f, second = 24.5f;

        System.out.println("--Before swap--");
        System.out.println("First number = " + first);
        System.out.println("Second number = " + second);

        first = first - second;
        second = first + second;
        first = second - first;

        System.out.println("--After swap--");
        System.out.println("First number = " + first);
        System.out.println("Second number = " + second);
    }
}
```
Question 35

Question:
Write a program to Display Fibonacci Series.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int n = 10, t1 = 0, t2 = 1;
        System.out.print("First " + n + " terms: ");

        for (int i = 1; i <= n; ++i) {
            System.out.print(t1 + " + ");

            int sum = t1 + t2;
            t1 = t2;
            t2 = sum;
        }
    }
}
```

Question 36

Question:
Write a program to Find GCD of two Numbers.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        int n1 = 81, n2 = 153, gcd = 1;
        for (int i = 1; i <= n1 && i <= n2; ++i) {
            // Checks if i is factor of both integers
            if (n1 % i == 0 && n2 % i == 0) {
                gcd = i;
            }
        }
        System.out.printf("G.C.D of %d and %d is %d", n1, n2, gcd);
    }
}
```

Question 37

Question:

Write a program to Find LCM of two Numbers.

Solution:
public class MyClass {
    public static void main(String[] args) {

        int n1 = 72, n2 = 120, lcm;

        // maximum number between n1 and n2 is stored in lcm
        lcm = (n1 > n2) ? n1 : n2;

        // Always true
        while (true) {
            if (lcm % n1 == 0 && lcm % n2 == 0) {
                System.out.printf("The LCM of %d and %d is %d.", n1, n2, lcm);
                break;
            }
            ++lcm;
        }
    }
}

Question 38

Question:

Write a program to Count Number of Digits in an Integer.

Solution:

public class MyClass {
    public static void main(String[] args) {

        // Count number of digits
        // Code here
    }
}
int count = 0, num = 3452;

while (num != 0)
{
    // num = num/10
    num /= 10;
    ++count;
}

System.out.println("Number of digits: "+count);

Question 39

Question:

Write a program to Reverse a Number.

Solution:

public class MyClass {

    public static void main(String[] args) {

        int num = 1234, reversed = 0;

        while (num != 0) {
            int digit = num % 10;
            reversed = reversed * 10 + digit;
            num /= 10;
        }

        System.out.println("Reversed number: "+reversed);
    }
}

203
System.out.println("Reversed Number: " + reversed);
}

---

**Question 40**

**Question:**

*Write a program to Display Prime Numbers Between Two Intervals.*

---

**Solution:**

```java
public class MyClass {

    public static void main(String[] args) {

        int low = 20, high = 50;

        while (low < high) {
            boolean flag = false;

            for(int i = 2; i <= low/2; ++i) {
                // condition for nonprime number
                if(low % i == 0) {
                    flag = true;
                    break;
                }
            }

            // Print prime numbers
            if(flag == false) {
                System.out.print(low + " ");
            }
        }
    }
}
```
if (!flag && low != 0 && low != 1)
    System.out.print(low + " ");

++low;
}
}

Question 41

Question:

Write a program to Check Armstrong Number.

Solution:

public class MyClass {

    public static void main(String[] args) {

        int number = 371, originalNumber, remainder, result = 0;

        originalNumber = number;

        while (originalNumber != 0)
        {
            remainder = originalNumber % 10;
            result += Math.pow(remainder, 3);
            originalNumber /= 10;
        }

        System.out.print(result + " ");
    }
}
if(result == number)
    System.out.println(number + " is an Armstrong number.");
else
    System.out.println(number + " is not an Armstrong number.");
}

Question 42

Question:

Write a program to Find G.C.D Using Recursion.

Solution:

public class MyClass {

    public static void main(String[] args) {
        int n1 = 366, n2 = 60;
        int hcf = hcf(n1, n2);

        System.out.printf("G.C.D of %d and %d is %d.", n1, n2, hcf);
    }

    public static int hcf(int n1, int n2)
    {
        if (n2 != 0)
            return hcf(n2, n1 % n2);
        else
            return n1;
    }
}
Question 43

Question:

Write a program to Reverse a Sentence Using Recursion.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {
        String sentence = "Go work";
        String reversed = reverse(sentence);
        System.out.println("The reversed sentence is: "+ reversed);
    }

    public static String reverse(String sentence) {
        if (sentence.isEmpty())
            return sentence;

        return reverse(sentence.substring(1)) + sentence.charAt(0);
    }
}
```
Question 44

Question:

Write a program to Calculate Standard Deviation.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {
        double[] numArray = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };
        double SD = calculateSD(numArray);

        System.out.format("Standard Deviation = %.6f", SD);
    }

    public static double calculateSD(double numArray[]) {
        double sum = 0.0, standardDeviation = 0.0;
        int length = numArray.length;

        for(double num : numArray) {
            sum += num;
        }

        double mean = sum/length;

        for(double num: numArray) {
            standardDeviation += Math.pow(num - mean, 2);
        }

        return Math.sqrt(standardDeviation/length);
    }

```
Question 45

Question:

Write a program to Add Two Matrix Using Multi-dimensional Arrays.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {
        int rows = 2, columns = 3;
        int[][] firstMatrix = {{2, 3, 4}, {5, 2, 3}};
        int[][] secondMatrix = {{-4, 5, 3}, {5, 6, 3}};

        // Adding Two matrices
        int[][] sum = new int[rows][columns];
        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < columns; j++) {
                sum[i][j] = firstMatrix[i][j] + secondMatrix[i][j];
            }
        }

        // Displaying the result
        System.out.println("Sum of two matrices is: ");
        for (int[] row : sum) {
            for (int column : row) {
                System.out.print(column + "  ");
            }
        }
    }
}
```
Question 46

Question:

Write a program to Multiply to Matrix Using Multi-dimensional Arrays.

Solution:

```java
public class MyClass {

    public static void main(String[] args) {
        int r1 = 2, c1 = 3;
        int r2 = 3, c2 = 2;
        int[][] firstMatrix = { {3, -2, 5}, {3, 0, 4} };
        int[][] secondMatrix = { {2, 3}, {-9, 0}, {0, 4} };

        // Multiplying Two matrices
        int[][] product = new int[r1][c2];
        for (int i = 0; i < r1; i++) {
            for (int j = 0; j < c2; j++) {
                for (int k = 0; k < c1; k++) {
                    product[i][j] += firstMatrix[i][k] * secondMatrix[k][j];
                }
            }
        }
    }
}
```
// Displaying the result
System.out.println("Sum of two matrices is: ");
for(int[] row : product) {
    for (int column : row) {
        System.out.print(column + " ");
    }
    System.out.println();
}

Question 47

Question:

Write a program to Find Transpose of a Matrix.

Solution:

public class MyClass {

    public static void main(String[] args) {
        int row = 2, column = 3;
        int[][] matrix = { {2, 3, 4}, {5, 6, 4} };

        // Display current matrix
        display(matrix);

        // Transpose the matrix
        int[][] transpose = new int[column][row];
for(int i = 0; i < row; i++) {
    for (int j = 0; j < column; j++) {
        transpose[j][i] = matrix[i][j];
    }
}

// Display transposed matrix
display(transpose);

public static void display(int[][] matrix) {
    System.out.println("The matrix is: ");
    for(int[] row : matrix) {
        for (int column : row) {
            System.out.print(column + " ");
        }
        System.out.println();
    }
}

Question 48

Question:

Write a program to find the occurrence (Frequency) of a character in a given string.

Solution:

public class MyClass {


```java
public static void main(String[] args) {
    String str = "This website is awesome."
    char ch = 'e'
    int frequency = 0;

    for(int i = 0; i < str.length(); i++) {
        if(ch == str.charAt(i)) {
            ++frequency;
        }
    }

    System.out.println("Frequency of " + ch + " = " + frequency);
}
}
```

---

**Question 49**

**Question:**

**Write a program to Remove All Whitespaces from a String.**

**Solution:**

```java
public class MyClass {

    public static void main(String[] args) {
        String sentence = "This is better."
        System.out.println("Original sentence: " + sentence);

        sentence = sentence.replaceAll("\s", "");

    }
}
```
System.out.println("After replacement: " + sentence);

}
}

Question 50

Question:

Write a program to Get Current Date/Time.

Solution:

public class MyClass {

    public static void main(String[] args) {
        LocalDateTime current = LocalDateTime.now();

        System.out.println("Current Date and Time is: " + current);
    }
}

Question 51

Question:

Write a program to Get Current Working Directory.
Solution:

```java
public class MyClass {

    public static void main(String[] args) {

        String path = System.getProperty("user.dir");

        System.out.println("Working Directory = " + path);
    }
}
```

Question 52

Question:

Write a program to Append Text to an Existing File.

Solution:

```java
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.nio.file.StandardOpenOption;

public class MyClass {

    public static void main(String[] args) {

    }
}
```
String path = System.getProperty("user.dir") + "\src\test.txt";
String text = "Added text";

try {
    Files.write(Paths.get(path), text.getBytes(), StandardOpenOption.APPEND);
} catch (IOException e) {
}
}

Question 53

Question:

Write a program to compare two strings.

Solution:

public class MyClass {

    public static void main(String[] args) {

        String style = "Bold";
        String style2 = "Bold";

        if(style == style2)
            System.out.println("Equal");
        else
            System.out.println("Not Equal");
    }
}
Question 54

Question:

Write a program to Create Directories.

Solution:

```java
import java.io.File;

class MyClass {
    public static void main(String[] args) {

        // creates a file object with specified path
        File file = new File("Java Example\directory");

        // tries to create a new directory
        boolean value = file.mkdir();
        if(value) {
            System.out.println("The new directory is created.");
        }
        else {
            System.out.println("The directory already exists.");
        }
    }
}
```
Question 55

Question:

Write a program to Rename File.

Solution:

```java
import java.io.File;

class MyClass {
    public static void main(String[] args) {

        // create a file object
        File file = new File("oldName");

        // create a file
        try {
            file.createNewFile();
        } catch(Exception e) {
            e.printStackTrace();
        }

        // create an object that contains the new name of file
        File newFile = new File("newName");

        // change the name of file
        boolean value = file.renameTo(newFile);

        if(value) {
            System.out.println("The name of the file is changed.");
        }
    }
}
```
else {
    System.out.println("The name cannot be changed.");
}

Question 56

Question:

Write a program to Get all Files Present in a Directory.

Solution:

import java.io.File;

class MyClass {
    public static void main(String[] args) {

        // creates a file object
        File file = new File("C:\Users\Guest User\Desktop\Java File\List Method");

        // returns an array of all files
        String[] fileList = file.list();

        for(String str : fileList) {
            System.out.println(str);
        }
    }
}

219
Question 57

Question:

Write a program to Copy File.

Solution:

```java
import java.io.FileInputStream;
import java.io.FileOutputStream;

class MyClass {
    public static void main(String[] args) {
        byte[] array = new byte[50];
        try {
            FileInputStream sourceFile = new FileInputStream("input.txt");
            FileOutputStream destFile = new FileOutputStream("newFile");

            // reads all data from input.txt
            sourceFile.read(array);

            // writes all data to newFile
            destFile.write(array);
            System.out.println("The input.txt file is copied to newFile.");

            // closes the stream
            sourceFile.close();
            destFile.close();
        } catch (Exception e) {
```
```
Question 58

Question:

Write a program to Implement Bubble Sort algorithm.

Solution:

// import the Class
import java.util.Arrays;
import java.util.Scanner;

class MyClass {

    // create an object of scanner
    // to take input from the user
    Scanner input = new Scanner(System.in);

    // method to perform bubble sort
    void bubbleSort(int array[]) {
        int size = array.length;

        // for ascending or descending sort
        System.out.println("Choose Sorting Order:");
        System.out.println("1 for Ascending \n2 for Descending");
        int sortOrder = input.nextInt();
// run loops two times
// first loop access each element of the array
for (int i = 0; i < size - 1; i++)

// second loop performs the comparison in each iteration
for (int j = 0; j < size - i - 1; j++)

// sort the array in ascending order
if (sortOrder == 1) {
  // compares the adjacent element
  if (array[j] > array[j + 1]) {
    // swap if left element is greater than right
    int temp = array[j];
    array[j] = array[j + 1];
    array[j + 1] = temp;
  }
}

// sort the array in descending order
else {
  // compares the adjacent element
  if (array[j] < array[j + 1]) {
    // swap if left element is smaller than right
    int temp = array[j];
    array[j] = array[j + 1];
    array[j + 1] = temp;
  }
}

// driver code
public static void main(String args[]) {
// create an array
int[] data = { -2, 45, 0, 11, -9 };

// create an object of Main class
Main bs = new Main();

// call the method bubbleSort using object bs
// pass the array as the method argument
bs.bubbleSort(data);
System.out.println("Sorted Array in Ascending Order:");

// call toString() of Arrays class
// to convert data into the string
System.out.println(Arrays.toString(data));

---

**Question 59**

**Question:**

Write a program to Implement Quick Sort Algorithm.

**Solution:**

```java
import java.util.Arrays;

class MyClass {

    // divide the array on the basis of pivot
    int partition(int array[], int low, int high) {
```
// select last element as pivot
int pivot = array[high];

// initialize the second pointer
int i = (low - 1);

// Put the elements smaller than pivot on the left and
// greater than pivot on the right of pivot
for (int j = low; j < high; j++) {

    // compare all elements with pivot
    // swap the element greater than pivot
    // with element smaller than pivot
    // to sort in descending order
    // if (array[j] >= pivot)
    if (array[j] <= pivot) {

        // increase the second pointer if
        // smaller element is swapped with greater
        i++;
        int temp = array[i];
        array[i] = array[j];
        array[j] = temp;
    }
}

// put pivot in position
// so that element on left are smaller
// element on right are greater than pivot
int temp = array[i + 1];
array[i + 1] = array[high];
array[high] = temp;
return (i + 1);

void quickSort(int array[], int low, int high) {  

if (low < high) {

    // Select pivot position and put all the elements smaller
    // than pivot on the left and greater than pivot on right
    int pi = partition(array, low, high);

    // sort the elements on the left of the pivot
    quickSort(array, low, pi - 1);

    // sort the elements on the right of pivot
    quickSort(array, pi + 1, high);
}

// Driver code
public static void main(String args[]) {

    // create an unsorted array
    int[] data = { 8, 7, 2, 1, 0, 9, 6 };  
    int size = data.length;

    // create an object of the Main class
    Main qs = new Main();

    // pass the array with the first and last index
    qs.quickSort(data, 0, size - 1);
    System.out.println("Sorted Array: ");
    System.out.println( Arrays.toString(data));
}

---

**Question 60**

**Question:**
Write a program to Implement Merge Sort Algorithm.

Solution:

```java
import java.util.Arrays;

class MyClass {

    // Merge two sub arrays L and M into array
    void merge(int array[], int p, int q, int r) {
        int n1 = q - p + 1;
        int n2 = r - q;

        int L[] = new int[n1];
        int M[] = new int[n2];

        // fill the left and right array
        for (int i = 0; i < n1; i++)
            L[i] = array[p + i];
        for (int j = 0; j < n2; j++)
            M[j] = array[q + 1 + j];

        // Maintain current index of sub-arrays and main array
        int i, j, k;
        i = 0;
        j = 0;
        k = p;

        // Until we reach either end of either L or M, pick larger among
        // elements L and M and place them in the correct position at A[p..r]
        // for sorting in descending
        // use if(L[i] >= M[j])
    }
}
```
while (i < n1 && j < n2) {
    if (L[i] <= M[j]) {
        array[k] = L[i];
        i++;
    } else {
        array[k] = M[j];
        j++;
    }
    k++;
}

// When we run out of elements in either L or M,
// pick up the remaining elements and put in A[p..r]
while (i < n1) {
    array[k] = L[i];
    i++;
    k++;
}

while (j < n2) {
    array[k] = M[j];
    j++;
    k++;
}

// Divide the array into two sub arrays, sort them and merge them
void mergeSort(int array[], int left, int right) {
    if (left < right) {

        // m is the point where the array is divided into two sub arrays
        int mid = (left + right) / 2;

        // recursive call to each sub arrays
        mergeSort(array, left, mid);
        mergeSort(array, mid + 1, right);
    }
// Merge the sorted sub arrays
merge(array, left, mid, right);

public static void main(String args[]) {

    // created an unsorted array
    int[] array = { 6, 5, 12, 10, 9, 1 };

    Main ob = new Main();

    // call the method mergeSort()
    // pass argument: array, first index and last index
    ob.mergeSort(array, 0, array.length - 1);

    System.out.println("Sorted Array:");
    System.out.println(Arrays.toString(array));
}

Question 61

Question:

Write a program to Implement Binary Search Algorithm.

Solution:

import java.util.Scanner;
class MyClass {
    int binarySearch(int array[], int element, int low, int high) {

        // Repeat until the pointers low and high meet each other
        while (low <= high) {

            // get index of mid element
            int mid = low + (high - low) / 2;

            // if element to be searched is the mid element
            if (array[mid] == element)
                return mid;

            // if element is less than mid element
            // search only the left side of mid
            if (array[mid] < element)
                low = mid + 1;

            // if element is greater than mid element
            // search only the right side of mid
            else
                high = mid - 1;

        }

        return -1;
    }

    public static void main(String args[]) {

        // create an object of Main class
        Main obj = new Main();

        // create a sorted array
        int[] array = {3, 4, 5, 6, 7, 8, 9};
        int n = array.length;

        // get input from user for element to be searched
Scanner input = new Scanner(System.in);

System.out.println("Enter element to be searched:");

// element to be searched
int element = input.nextInt();
input.close();

// call the binary search method
// pass arguments: array, element, index of first and last element
int result = obj.binarySearch(array, element, 0, n - 1);
if (result == -1)
    System.out.println("Not found");
else
    System.out.println("Element found at index " + result);
}
}

Question 62

Question:
Write a program to Load File as Input Stream.

Solution:

import java.io.InputStream;
import java.io.FileInputStream;

public class MyClass {

}
public static void main(String args[]) {
    try {
        // file input.txt is loaded as input stream
        // input.txt file contains:
        // This is a content of the file input.txt
        FileInputStream input = new FileInputStream("input.txt");

        System.out.println("Data in the file: ");

        // Reads the first byte
        int i = input.read();

        while(i != -1) {
            System.out.print((char)i);

            // Reads next byte from the file
            i = input.read();
        }
        input.close();
    }
    catch(Exception e) {
        e.printStackTrace();
    }
}

Question 63

Question:
Write a program to Create File and Write to the File.

Solution:

import java.io.File;

class MyClass {
    public static void main(String[] args) {

        // create a file object for the current location
        File file = new File("JavaFile.java");

        try {

            // create a new file with name specified
            // by the file object
            boolean value = file.createNewFile();
            if (value) {
                System.out.println("New Java File is created.");
            } else {
                System.out.println("The file already exists.");
            }
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
}
Question 64

Question:

Write a program to Read the Content of a File Line by Line.

Solution:

```java
import java.io.BufferedInputStream;
import java.io.FileInputStream;

class MyClass {
    public static void main(String[] args) {
        try {
            // Creates a FileInputStream
            FileInputStream file = new FileInputStream("input.txt");

            // Creates a BufferedInputStream
            BufferedInputStream input = new BufferedInputStream(file);

            // Reads first byte from file
            int i = input.read();

            while (i != -1) {
                System.out.print((char) i);

                // Reads next byte from the file
                i = input.read();
            }
        }
        input.close();
    }
}
```
```java
import java.io.File;

class MyClass {
    public static void main(String[] args) {

        // creates a file object
        File file = new File("JavaFile.java");

        // deletes the file
        boolean value = file.delete();
        if(value) {
            System.out.println("JavaFile.java is successfully deleted.");
        } else {
            System.out.println("File doesn't exist");
        }
    }
}
```
Question 66

Question:

Write a program to Get the File Extension.

Solution:

```java
import java.io.File;

class MyClass {

    public static void main(String[] args) {
        File file = new File("Test.java");

        // convert the file name into string
        String fileName = file.toString();

        int index = fileName.lastIndexOf('.');
        if(index > 0) {
            String extension = fileName.substring(index + 1);
            System.out.println("File extension is " + extension);
        }
    }
}
```
Question 67

Question:

Write a program to Count number of lines present in the file.

Solution:

```java
import java.io.File;
import java.util.Scanner;

class MyClass {
    public static void main(String[] args) {

        int count = 0;

        try {
            // create a new file object
            File file = new File("input.txt");

            // create an object of Scanner
            // associated with the file
            Scanner sc = new Scanner(file);

            // read each line and
            // count number of lines
            while(sc.hasNextLine()) {
                sc.nextLine();
                count++;
            }
            System.out.println("Total Number of Lines: " + count);

            // close scanner
        }
    }
}
```
sc.close();
}
catch (Exception e) {
    e.getStackTrace();
}
}

Question 68

Question:

Write a program to Merge two lists.

Solution:

```java
import java.util.ArrayList;
import java.util.List;

class MyClass {
    public static void main(String[] args) {

        // create first list
        List<Integer> prime = new ArrayList<>();
        prime.add(2);
        prime.add(3);
        prime.add(5);
        System.out.println("First List: " + prime);

        // create second list
        List<Integer> even = new ArrayList<>();
        even.add(4);
```
even.add(6);
System.out.println("Second List: " + even);

// create merged list
List<Integer> numbers = new ArrayList<>();
numbers.addAll(prime);
numbers.addAll(even);

System.out.println("Merged List: " + numbers);

Question 69

Question:

Write a program to Remove duplicate elements from Array List.

Solution:

import java.util.ArrayList;
import java.util.Arrays;
import java.util.LinkedHashSet;
import java.util.Set;

class MyClass {
    public static void main(String[] args) {

        // create an arraylist from the array
        // using asList() method of the Arrays class

}
```java
ArrayList<Integer> numbers = new ArrayList<>(Arrays.asList(1, 2, 3, 4, 1, 3));
System.out.println("ArrayList with duplicate elements: " + numbers);

// convert the arraylist into a set
Set<Integer> set = new LinkedHashSet<>();
set.addAll(numbers);

// delete all elements of arraylist
numbers.clear();

// add element from set to arraylist
numbers.addAll(set);
System.out.println("ArrayList without duplicate elements: " + numbers);
```

---

**Question 70**

**Question:**

Write a program to Calculate union of two sets.

**Solution:**

```java
import java.util.HashSet;
import java.util.Set;

class MyClass {
    public static void main(String[] args) {

        // create the first set
```
```java
Set<Integer> evenNumbers = new HashSet<>();
evenNumbers.add(2);
evenNumbers.add(4);
System.out.println("Set1: " + evenNumbers);

// create second set
Set<Integer> numbers = new HashSet<>();
numbers.add(1);
numbers.add(3);
System.out.println("Set2: " + numbers);

// Union of two sets
numbers.addAll(evenNumbers);
System.out.println("Union is: " + numbers);
```

Question 71

**Question:**

Write a program to determine whether one string is a rotation of another.

**Solution:**

```java
public class MyClass {
    public static void main(String[] args) {
        String str1 = "abcde", str2 = "deabc";

        if(str1.length() != str2.length()){
            System.out.println("Second string is not a rotation of first string");
        }
    }
}
else {
    //Concatenate str1 with str1 and store it in str1
    str1 = str1.concat(str1);
    //Check whether str2 is present in str1
    if(str1.indexOf(str2) != -1)
        System.out.println("Second string is a rotation of first string");
    else
        System.out.println("Second string is not a rotation of first string");
}

Question 72

Question:

Write a program to find the duplicate characters in a string.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        String string1 = "Great responsibility";
        int count;

        //Converts given string into character array
        char string[] = string1.toCharArray();

        System.out.println("Duplicate characters in a given string: ");
```
Question 73

Question:

Write a program to find the duplicate words in a string.

Solution:

```java
public class MyClass {
    public static void main(String[] args) {
        String string = "Big black bug bit a big black dog on his big black nose";
        int count;

        //Converts the string into lowercase
```
string = string.toLowerCase();

//Split the string into words using built-in function
String words[] = string.split(" ");

System.out.println("Duplicate words in a given string :");
for(int i = 0; i < words.length; i++) {
    count = 1;
    for(int j = i+1; j < words.length; j++) {
        if(words[i].equals(words[j])) {
            count++;
            //Set words[j] to 0 to avoid printing visited word
            words[j] = "0";
        }
    }
    //Displays the duplicate word if count is greater than 1
    if(count > 1 && words[i] != "0")
        System.out.println(words[i]);
}
}

Question 74

Question:

Write a program to find maximum and minimum occurring character in a string.

Solution:
public class MyClass {

    public static void main(String[] args) {
        String str = "grass is greener on the other side";
        int[] freq = new int[str.length()];
        char minChar = str.charAt(0), maxChar = str.charAt(0);
        int i, j, min, max;

        //Converts given string into character array
        char string[] = str.toCharArray();

        //Count each word in given string and store in array freq
        for(i = 0; i < string.length; i++) {
            freq[i] = 1;
            for(j = i+1; j < string.length; j++) {
                if(string[i] == string[j] && string[i] != ' ' && string[i] != '0') {
                    freq[i]++;
                    //Set string[j] to 0 to avoid printing visited character
                    string[j] = '0';
                }
            }
        }

        //Determine minimum and maximum occurring characters
        min = max = freq[0];
        for(i = 0; i < freq.length; i++) {

            //If min is greater than frequency of a character
            //then, store frequency in min and corresponding character in minChar
            if(min > freq[i] && freq[i] != '0') {
                min = freq[i];
                minChar = string[i];
            }

            //If max is less than frequency of a character
            //then, store frequency in max and corresponding character in maxChar
            if(max < freq[i]) {

        }
```java
max = freq[i];
maxChar = string[i];
}
}

System.out.println("Minimum occurring character: " + minChar);
System.out.println("Maximum occurring character: " + maxChar);
}
}
```

**Question 75**

**Question:**

Write a program to separate the Individual Characters from a String.

**Solution:**

```java
public class MyClass {
    public static void main(String[] args) {
        String string = "characters ";

        //Displays individual characters from given string
        System.out.println("Individual characters from given string: ");

        //Iterate through the string and display individual character
        for(int i = 0; i < string.length(); i++){
            System.out.print(string.charAt(i) + " ");
        }
    }
}
```
Python Exercises

Question 1

Question:

Write a program to Add Two Numbers.

Solution:

```python
a = 1
b = 2
c = a + b
print(c)
```

```python
a = int(input("enter a number: "))
b = int(input("enter a number: "))
c = a + b
print(c)
```

Question 2

Question:
Write a program to find whether a given number (accept from the user) is even or odd, print out an appropriate message to the user.

Solution:

```python
a = int(input("enter a number: "))
if a % 2 == 0:
    print("This is an even number.")
else:
    print("This is an odd number.")
```

Question 3

Question:

Write a program to check whether a number entered by the user is positive, negative or zero.

Solution:

```python
a = int(input("Enter a number: "))
if a > 0:
    print("Positive number")
elif a == 0:
    print("Zero")
else:
    print("Negative number")
```
Question 4

Question:

Write a program to display the calendar of a given date.

Solution:

```python
import calendar
yy = int(input("Enter year: "))
mm = int(input("Enter month: "))
print(calendar.month(yy, mm))
```

Question 5

Question:

Write a program to ask the user to enter the string and print that string as output of the program.

Solution:

```python
string = input("Enter string: ")
print("You entered:",string)
```
Question 6

**Question:**

Write a program to Concatenate Two Strings.

**Solution:**

```python
string1 = input("Enter first string to concatenate: ")
string2 = input("Enter second string to concatenate: ")
string3 = string1 + string2
print("String after concatenation = ",string3)
```

Question 7

**Question:**

Write a program to Check if an item exists in the list.

**Solution:**

```python
list_of_items = ["ball", "book", "pencil"]
item = input("Type item to check: ")
if item in list_of_items:
    print("Item exists in the list.")
```
else:
    print("Item does not exist in the list.")

---

Question 8

**Question:**

Write a program to Join two or more lists.

**Solution:**

```python
list1 = ["This", "is", "a", "sample", "program"]
list2 = [10, 2, 45, 3, 5, 7, 8, 10]
finalList = list1 + list2
print(finalList)
```

---

Question 9

**Question:**

Write a program to Calculate Cube of a Number.

**Solution:**

```python
import math
```
Question 10

Question:
Write a program to Calculate Square root of a Number.

Solution:

```python
import math
a = int(input("Enter a number: "))
b = math.sqrt(a)
print (b)
```

Question 11

Question:
Write a program that takes a list of numbers (for example, a = [5, 10, 15, 20, 25]) and makes a new list of only the first and last elements of the given list.

Solution:

```python
import math
a = int(input("Enter a number: "))
b = math.sqrt(a)
print (b)
```
a = [5, 10, 15, 20, 25]
print([a[0], a[4]])

Question 12

Question:

Take a list, say for example this one: a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89] and write a program that prints out all the elements of the list that are less than 5.

Solution:

a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
for i in a:
    if i < 5:
        print(i)

Question 13

Question:

Let’s say I give you a list saved in a variable: a = [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]. Write one line of Python that takes this list ‘a’ and makes a new list that has only the even elements of this list in it.
Solution:

```python
a = [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
b = [number for number in a if number % 2 == 0]
print(b)
```

Question 14

Question:

Ask the user for a string and print out whether this string is a palindrome or not (A palindrome is a string that reads the same forwards and backwards).

Solution:

```python
a = input("Please enter a word: ")
c = a.casefold()
b = reversed(c)
if list(c) == list(b):
    print("It is palindrome")
else:
    print("It is not palindrome")
```

Question 15

Question:
Take two lists, say for example these two: \( a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89] \) \( b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13] \) and write a program that returns a list that contains only the elements that are common between the lists (without duplicates). Make sure your program works on two lists of different sizes.

Solution:

\[
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]\\nb = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]\\nresult = [i for i in set(a) if i in b]\\nprint(result)
\]

Question 16

Question:

Write a program to add a string to text file.

Solution:

```python
file = open("testfile.txt","w")
file.write("Hello World")
file.write("This is our new text file")
file.write("and this is another line.")
file.write("Why? Because we can.")
file.close()
```
Question 17

Question:

Write a program to read a file and display its contents on console.

Solution:

```python
with open('testfile.txt') as f:
    line = f.readline()
    while line:
        print(line)
        line = f.readline()
```

Question 18

Question:

Take two sets, say for example these two: a = {1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89} b = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13} and write a program that returns a set that contains only the elements that are common between the sets.

Solution:

```python
a = {1, 1, 2, 2, 3, 5, 8, 13, 21, 34, 55, 89}
```
Question 19

Question:
Write a program to split the characters of the given string into a list.

Solution:

```python
s = "mystring"
l = list(s)
print(l)
```

Question 20

Question:
Create a program that asks the user for a number and then prints out a list of all the divisors of that number.

Solution:
n=int(input("Enter an integer: "))
print("The divisors of the number are: ")
for i in range(1,n+1):
    if(n%i==0):
        print(i)

Question 21

Question:
Write a program to Find the largest of three numbers.

Solution:

a = int(input("Enter first number: "))
b = int(input("Enter second number: "))
c = int(input("Enter third number: "))
if (a > b) and (a > c):
    largest = a
elif (b > a) and (b > c):
    largest = b
else:
    largest = c
print("The largest number is", largest)

Question 22

Question:
Write a Program to Find Absolute value of a Number.

Solution:

```python
num = int(input("Enter a number: "))
if num >= 0:
    print(num)
else:
    print(-num)
```

Question 23

Question:

Write a program to Find the length of a String.

Solution:

```python
print("Enter 'y' for exit.")
string = input("Enter a string: ")
if string == 'y':
    exit()
else:
    print("Length of the string =", len(string))
```
Question 24

Question:

Write a program to Print Natural Numbers from 1 to N.

Solution:

```python
N = int(input("Please Enter any Number: "))
for i in range(1, N+1):
    print (i)
```

Question 25

Question:

Write a program to calculate the sum and average of Natural Numbers from 1 to N.

Solution:

```python
N = int(input("Please Enter any Number: "))
sum = 0
for i in range(1,N+1):
    sum = sum + i
print(sum)
average = sum / N
```
Question 26

Question:

Write a program to Print a Statement Any Number of Times.

Solution:

```python
n = int(input("Please Enter any Number: ")
for i in range(n):
    print("hello world")
```

Question 27

Question:

Write a program To Multiply Two Numbers Using Function.

Solution:

```python
def my_function():
    a = int(input("enter a number: "))
    b=int(input("enter a number: "))
```
c = a*b
    return c
d = my_function()
print(d)

---

Question 28

Question:

Write a program To add an item to the end of the list.

Solution:

list1 = ["pen", "book", "ball"]
list1.append("bat")
print(list1)

---

Question 29

Question:

Write a program To remove an item from the list.

Solution:
list1 = ["pen", "book", "ball"]
list1.remove("ball")
print(list1)

Question 30

Question:

Write a program To print the number of elements in an array.

Solution:

list1 = ["pen", "book", "ball"]
a = len(list1)
print(a)

Question 31

Question:

Write a program To calculate the variance and standard deviation of the elements of the list.

Solution:
import numpy as np
a = [2, 6, 8, 12, 18, 24, 28, 32]
variance = np.var(a)
std = np.std(a)
print(variance)
print(std)

Question 32

Question:

Write a program to get the difference between the two lists.

Solution:

list1 = [4, 5, 6, 7]
list2 = [4, 5]
print(list(set(list1) - set(list2)))

Question 33

Question:

Write a program to select an item randomly from a list.
Solution:

```python
import random
list = ['Paper', 'Pencil', 'Book', 'Bag', 'Pen']
print(random.choice(list))
```

---

**Question 34**

**Question:**

Write a program that prints all the numbers from 0 to 6 except 2 and 6.

**Solution:**

```python
for x in range(6):
    if (x == 2 or x==6):
        continue
    print(x)
```

---

**Question 35**

**Question:**

Write a program that takes input from the user and displays that input back in upper and lower cases.
Solution:

a = input("What's your name? ")
print(a.upper())
print(a.lower())

Question 36

Question:

Write a program to check whether a string starts with specified characters.

Solution:

string = "myw3schools.com"
print(string.startswith("w3s"))

Question 37

Question:

Write a program to create the multiplication table (from 1 to 10) of a number.

Solution:
n = int(input("Enter a number: "))
for i in range(1,11):
    print(n,'x',i,'=',n*i)

---

**Question 38**

**Question:**

Write a program to check a triangle is equilateral, isosceles or scalene.

**Solution:**

```python
print("Enter lengths of the triangle sides: ")
a = int(input("a: "))
b = int(input("b: "))
c = int(input("c: "))
if a == b == c:
    print("Equilateral triangle")
elif a==b or b==c or c==a:
    print("isosceles triangle")
else:
    print("Scalene triangle")
```

---

**Question 39**

**Question:**

---
Write a program to sum of two given integers. However, if the sum is between 15 to 20 it will return 20.

Solution:

```python
a = int(input("enter a number: "))
b = int(input("enter a number: "))
c = a+b
if c in range(15, 20):
    print(20)
else:
    print(c)
```

Question 40

Question:

Write a program to convert degree to radian.

Solution:

```python
pi=22/7
degree = int(input("Input degrees: "))
radian = degree*(pi/180)
print(radian)
```
Question 41

Question:

Write a program to Generate a Random Number.

Solution:

```python
import random
print(random.randint(0, 9))
```

---

Question 42

Question:

Write a Program to find the semi-perimeter of triangle.

Solution:

```python
a = int(input('Enter first side: '))
b = int(input('Enter second side: '))
c = int(input('Enter third side: '))
s = (a + b + c) / 2
print(s)
```
Question 43

Question:

Given a list of numbers, Iterate it and print only those numbers which are divisible of 2.

Solution:

```
List = [10, 20, 33, 46, 55]
for i in List:
    if (i % 2 == 0):
        print(i)
```

Question 44

Question:

Write a program to Multiply all numbers in the list.

Solution:

```
import numpy
list = [1, 2, 3]
result = numpy.prod(list)
print(result)
```
Question 45

Question:

Write a program to print ASCII Value of a character.

Solution:

```python
a = 'j'
print("The ASCII value of '" + a +"' is", ord(a))
```

Question 46

Question:

Write a program to list files in a directory.

Solution:

```python
# Import os module to read directory
import os

# Set the directory path
path = 'C:/Users/Manju/.spyder-py3/
```
# Read the content of the file
files = os.listdir(path)

# Print the content of the directory
for file in files:
    print(file)

---

**Question 47**

**Question:**

Write a program to Read and Write File.

**Solution:**

#Assign the filename
filename = "languages.txt"

# Open file for writing
fileHandler = open(filename, "w")

# Add some text
fileHandler.write("Bash\n")
fileHandler.write("Python\n")
fileHandler.write("PHP\n")

# Close the file
fileHandler.close()

# Open file for reading
fileHandler = open(filename, "r")
# Read a file line by line
for line in fileHandler:
    print(line)

# Close the file
fileHandler.close()

---

**Question 48**

**Question:**

Write a program to add and search data in the dictionary.

**Solution:**

# Define a dictionary
customers = {'1':'Mehzabin Afroze','2':'Md. Ali',
'm3':'Mosarof Ahmed','4':'Mila Hasan', '5':'Yaqub Ali'}

# Append a new data
customers['6'] = 'Mehboba Ferdous'

print("The customer names are:")
# Print the values of the dictionary
for customer in customers:
    print(customers[customer])

# Take customer ID as input to search
name = input("Enter customer ID:"
# Search the ID in the dictionary
for customer in customers:
    if customer == name:
        print(customers[customer])
        break

---

**Question 49**

**Question:**

Write a program to add and search data in the set.

**Solution:**

```python
# Define the number set
numbers = {23, 90, 56, 78, 12, 34, 67}

# Add a new data
numbers.add(50)
# Print the set values
print(numbers)

message = "Number is not found"

# Take a number value for search
search_number = int(input("Enter a number:"))
# Search the number in the set
for val in numbers:
    if val == search_number:
```

---

273
print(message)

Question 50

Question:

Write a program to demonstrate throw and catch exception.

Solution:

# Try block
try:
    # Take a number
    number = int(input("Enter a number: "))
    if number % 2 == 0:
        print("Number is even")
    else:
        print("Number is odd")

# Exception block
except (ValueError):
    # Print error message
    print("Enter a numeric value")
Question 51

Question:

Write a program to illustrate password authentication.

Solution:

```python
# import getpass module
import getpass

# Take password from the user
passwd = getpass.getpass('Password: ')

# Check the password
if passwd == "python":
    print("You are authenticated")
else:
    print("You are not authenticated")
```

Question 52

Question:

Write a program to calculate the average of numbers in a given list.
Solution:

```python
n=int(input("Enter the number of elements to be inserted: "))
a=[]
for i in range(0,n):
    elem=int(input("Enter element: "))
    a.append(elem)
avg=sum(a)/n
print("Average of elements in the list",round(avg,2))
```

---

**Question 53**

**Question:**

Write a program to exchange the values of two numbers without using a temporary variable.

---

**Solution:**

```python
a=int(input("Enter value of first variable: "))
b=int(input("Enter value of second variable: "))
a=a+b
b=a-b
a=a-b
print("a is:"+a," b is:"+b)
```
**Question 54**

**Question:**

Write a program to reverse a given number.

**Solution:**

```python
n = int(input("Enter number: "))
rev = 0
while (n > 0):
    dig = n % 10
    rev = rev * 10 + dig
    n = n // 10
print("Reverse of the number: ", rev)
```

**Question 55**

**Question:**

Write a program to take in the marks of 5 subjects and display the grade.

**Solution:**

```python
sub1 = int(input("Enter marks of the first subject: "))
sub2 = int(input("Enter marks of the second subject: "))
sub3 = int(input("Enter marks of the third subject: "))`
sub4 = int(input("Enter marks of the fourth subject: "))
sub5 = int(input("Enter marks of the fifth subject: "))

avg = (sub1 + sub2 + sub3 + sub4 + sub5) / 5

if (avg >= 90):
    print("Grade: A")
elif (avg >= 80 & avg < 90):
    print("Grade: B")
elif (avg >= 70 & avg < 80):
    print("Grade: C")
elif (avg >= 60 & avg < 70):
    print("Grade: D")
else:
    print("Grade: F")

Question 56

Question:

Write a program to print all numbers in a range divisible by a given number.

Solution:

lower = int(input("Enter lower range limit: "))
upper = int(input("Enter upper range limit: "))
n = int(input("Enter the number to be divided by: "))

for i in range(lower, upper + 1):
    if (i % n == 0):
        print(i)
Question 57

Question:

Write a program to read two numbers and print their quotient and remainder.

Solution:

```python
a = int(input("Enter the first number: "))
b = int(input("Enter the second number: "))
quotient = a // b
remainder = a % b
print("Quotient is:", quotient)
print("Remainder is:", remainder)
```

Question 58

Question:

Write a program to accept three distinct digits and print all possible combinations from the digits.

Solution:
a = int(input("Enter first number:"))
b = int(input("Enter second number:"))
c = int(input("Enter third number:"))
d = []
d.append(a)
d.append(b)
d.append(c)
for i in range(0, 3):
    for j in range(0, 3):
        for k in range(0, 3):
            if (i != j) & (j != k) & (k != i):
                print(d[i], d[j], d[k])

Question 59

Question:

Write a program to print odd numbers within a given range.

Solution:

lower = int(input("Enter the lower limit for the range:"))
upper = int(input("Enter the upper limit for the range:"))
for i in range(lower, upper + 1):
    if (i % 2) != 0:
        print(i)
Question 60

Question:

*Write a program to find the smallest divisor of an integer.*

Solution:

```python
n = int(input("Enter an integer:")

a = []
for i in range(2, n+1):
    if n % i == 0:
        a.append(i)

a.sort()
print("Smallest divisor is:", a[0])
```

---

Question 61

Question:

*Write a program to count the number of digits in a number.*

Solution:
n=int(input("Enter number:"))
count=0
while(n>0):
    count=count+1
    n=n//10
print("The number of digits in the number are:",count)

---

Question 62

**Question:**

Write a program to read a number \( n \) and print and compute the series

\[1+2+...+n=\].

---

**Solution:**

```python
n=int(input("Enter a number: "))
a=[]
for i in range(1,n+1):
    print(i,sep=" ",end=" ")
    if(i<n):
        print("+",sep=" ",end=" ")
a.append(i)
print("=",sum(a))
```
Question 63

Question:

Write a program to read a number n and print the natural numbers summation pattern.

Solution:

```python
n = int(input("Enter a number: "))
for j in range(1, n+1):
    a = []
    for i in range(1, j+1):
        print(i, sep=" ", end=" ")
        if i < j:
            print(+", sep=" ", end=" ")
            a.append(i)
    print(=", sum(a))

print()
```

Question 64

Question:
Write a program to read a number n and print an identity matrix of the desired size.

Solution:

```python
n=int(input("Enter a number: "))
for i in range(0,n):
    for j in range(0,n):
        if(i==j):
            print("1",sep=" ",end=" ")
        else:
            print("0",sep=" ",end=" ")
    print()
```

Question 65

Question:

Write a program to read a number n and print an inverted star pattern of the desired size.

Solution:

```python
n=int(input("Enter number of rows: "))
for i in range (n,0,-1):
    print((n-i) * ' ' + i * '*')
```
Question 66

Question:

Write a program to print prime numbers in a range using Sieve of Eratosthenes.

Solution:

```python
n = int(input("Enter upper limit of range: "))
sieve = set(range(2, n+1))
while sieve:
    prime = min(sieve)
    print(prime, end="\t")
    sieve -= set(range(prime, n+1, prime))

print()
```

Question 67

Question:

Write a program to find the largest number in a list.
Solution:

```python
a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
a.sort()
print("Largest element is:",a[n-1])
```

Question 68

**Question:**

Write a program to find the second largest number in a list.

**Solution:**

```python
a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
a.sort()
print("Second largest element is:",a[n-2])
```
Question 69

Question:

Write a program to put the even and odd elements in a list into two different lists.

Solution:

```python
a = []
n = int(input("Enter number of elements:"))
for i in range(1, n+1):
    b = int(input("Enter element:"))
    a.append(b)
even = []
odd = []
for j in a:
    if (j%2==0):
        even.append(j)
    else:
        odd.append(j)
print("The even list", even)
print("The odd list", odd)
```

Question 70

Question:

Write a program to sort the list according to the second element in the sublist.
Solution:

```python
a=[['A', 34], ['B', 21], ['C', 26]]
for i in range(len(a)):
    for j in range(len(a)-i-1):
        if a[j][1]>a[j+1][1]:
            temp=a[j]
            a[j]=a[j+1]
            a[j+1]=temp

print(a)
```

Question 71

**Question:**

Write a program to find the second largest number in a list using bubble sort.

**Solution:**

```python
a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
for i in range(len(a)):
    for j in range(len(a)-i-1):
        if a[j][1]>a[j+1][1]:
            temp=a[j]
            a[j]=a[j+1]
            a[j+1]=temp
```
if(a[j]>a[j+1]):
    temp=a[j]
    a[j]=a[j+1]
    a[j+1]=temp
print('Second largest number is:',a[n-2])

Question 72

Question:
Write a program to sort a list according to the length of the elements.

Solution:

a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=input("Enter element:")
    a.append(b)
a.sort(key=len)
print(a)

Question 73

Question:
Write a program to create a list of tuples with the first element as the number and the second element as the square of the number.

Solution:

```python
l_range = int(input("Enter the lower range: "))
u_range = int(input("Enter the upper range: "))
a = [(x, x**2) for x in range(l_range, u_range + 1)]
print(a)
```

Question 74

Question:

Write a program to create a list of all numbers in a range which are perfect squares and the sum of the digits of the number is less than 10.

Solution:

```python
l = int(input("Enter lower range: "))
u = int(input("Enter upper range: "))
a = []
a = [x for x in range(l, u + 1) if (int(x**0.5))**2 == x and sum(list(map(int, str(x)))) < 10]
print(a)
```
Question 75

Question:

Write a program to find the cumulative sum of a list where the i\textsuperscript{th} element is the sum of the first i+1 elements from the original list.

Solution:

```python
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
    element=int(input("Enter element" +str(x+1)+ ":"))
    a.append(element)
b=[sum(a[0:x+1]) for x in range(0,len(a))]
print("The original list is: ",a)
print("The new list is: ",b)
```

Question 76

Question:

Write a program to generate random numbers from 1 to 20 and append them to the list.
Solution:

```python
import random
a=[]
n=int(input("Enter number of elements:"))
for j in range(n):
    a.append(random.randint(1,20))
print('Randomised list is: ',a)
```

Question 77

Question:

Write a program to sort a list of tuples in increasing order by the last element in each tuple.

Solution:

```python
def last(n):
    return n[-1]

def sort(tuples):
    return sorted(tuples, key=last)

a=input("Enter a list of tuples:")
print("Sorted:")
```
Question 78

Question:

Write a program to swap the first and last value of a list.

Solution:

```python
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
    element=int(input("Enter element" + str(x+1) + ":"))
    a.append(element)
temp=a[0]
a[0]=a[n-1]
a[n-1]=temp
print("New list is:")
print(a)
```

Question 79

Question:

Write a program to remove the duplicate items from a list.
Solution:

```python
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
    element=int(input("Enter element" + str(x+1) + ":"))
    a.append(element)
b = set()
unique = []
for x in a:
    if x not in b:
        unique.append(x)
        b.add(x)
print("Non-duplicate items:")
print(unique)
```

Question 80

Question:

Write a program to read a list of words and return the length of the longest one.

Solution:

```python
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
    element=input("Enter element" + str(x+1) + ":")
```
```python
max1 = len(a[0])
temp = a[0]
for i in a:
    if len(i) > max1:
        max1 = len(i)
temp = i
print("The word with the longest length is:")
print(temp)
```

**Question 81**

**Question:** Write a program to remove the i\textsuperscript{th} occurrence of the given word in list where words can repeat.

**Solution:**

```python
a = []
n = int(input("Enter the number of elements in list:"))
for x in range(0, n):
    element = input("Enter element " + str(x + 1) + ":")
    a.append(element)
print(a)
c = []
count = 0
b = input("Enter word to remove: ")
n = int(input("Enter the occurrence to remove: "))
for i in a:
```
if(i==b):
    count=count+1
    if(count!=n):
        c.append(i)
    else:
        c.append(i)
if(count==0):
    print("Item not found ")
else:
    print("The number of repetitions is: ",count)
    print("Updated list is: ",c)
    print("The distinct elements are: ",set(a))

---

Question 82

Question:

Write a program to solve the maximum subarray problem using divide and conquer technique.

Solution:

def find_max_subarray(alist, start, end):
    """Returns (l, r, m) such that alist[l:r] is the maximum subarray in
    end."""
    # base case
    if start == end - 1:
        return start, end, alist[start]
else:
    mid = (start + end)//2
    left_start, left_end, left_max = find_max_subarray(alist, start, mid)
    right_start, right_end, right_max = find_max_subarray(alist, mid, end)
    cross_start, cross_end, cross_max = find_max_crossing_subarray(alist, start, mid, end)
    if (left_max > right_max and left_max > cross_max):
        return left_start, left_end, left_max
    elif (right_max > left_max and right_max > cross_max):
        return right_start, right_end, right_max
    else:
        return cross_start, cross_end, cross_max

def find_max_crossing_subarray(alist, start, mid, end):
    """Returns (l, r, m) such that alist[l:r] is the maximum subarray within
    alist with start <= l < mid <= r < end with sum m. The arguments start, mid,
    end must satisfy start <= mid <= end.""
    sum_left = float('-inf')
    sum_temp = 0
    cross_start = mid
    for i in range(mid - 1, start - 1, -1):
        sum_temp = sum_temp + alist[i]
        if sum_temp > sum_left:
            sum_left = sum_temp
            cross_start = i

    sum_right = float('-inf')
    sum_temp = 0
    cross_end = mid + 1
    for i in range(mid, end):
        sum_temp = sum_temp + alist[i]
        if sum_temp > sum_right:
            sum_right = sum_temp
            cross_end = i + 1
    return cross_start, cross_end, sum_left + sum_right

alist = input('Enter the list of numbers: ')

297
Question 83

Question:

Write a program to solve the maximum subarray problem using Kadane’s algorithm.

Solution:

def find_max_subarray(alist, start, end):
    """Returns (l, r, m) such that alist[l:r] is the maximum subarray in
    end."""
    max_ending_at_i = max_seen_so_far = alist[start]
    max_left_at_i = max_left_so_far = start
    # max_right_at_i is always i + 1
    max_right_so_far = start + 1
    for i in range(start + 1, end):
        if max_ending_at_i > 0:
            max_ending_at_i += alist[i]
        else:
            max_ending_at_i = alist[i]
            max_left_at_i = i
if max_ending_at_i > max_seen_so_far:
    max_seen_so_far = max_ending_at_i
    max_left_so_far = max_left_at_i
    max_right_so_far = i + 1
return max_left_so_far, max_right_so_far, max_seen_so_far

alist = input('Enter the list of numbers: ')
alist = alist.split()
alist = [int(x) for x in alist]
start, end, maximum = find_max_subarray(alist, 0, len(alist))
print('The maximum subarray starts at index {}, ends at index {}' ' and has sum {}.').format(start, end - 1, maximum)

Question 84

Question:

Write a program to find the element that occurs odd number of times in a list.

Solution:

def find_odd_occuring(alist):
    """Return the element that occurs odd number of times in alist.

    alist is a list in which all elements except one element occurs an even number of times.
    """
    ans = 0
for element in alist:
    ans ^= element

return ans

alist = input('Enter the list: ').split()
alist = [int(i) for i in alist]
ans = find_odd_occuring(alist)
print('The element that occurs odd number of times:', ans)

---

**Question 85**

**Question:**

Write a program to check if a date is valid and print the incremented date if it is.

**Solution:**

date=input("Enter the date: ")
dd,mm,yy=date.split('/'

dd=int(dd)
mm=int(mm)
yy=int(yy)
if(mm==1 or mm==3 or mm==5 or mm==7 or mm==8 or mm==10 or mm==12):
    max1=31
elif(mm==4 or mm==6 or mm==9 or mm==11):
    max1=30

...
elif(\(yy \% 4==0 \text{ and } yy \% 100!=0 \text{ or } yy \% 400==0\)):
    max1=29
else:
    max1=28
if(mm<1 or mm>12):
    print("Date is invalid.")
elif(dd<1 or dd>max1):
    print("Date is invalid.")
elif(dd==max1 and mm!=12):
    dd=1
    mm=mm+1
    print("The incremented date is: ",dd,mm,yy)
elif(dd==31 and mm==12):
    dd=1
    mm=1
    yy=yy+1
    print("The incremented date is: ",dd,mm,yy)
else:
    dd=dd+1
    print("The incremented date is: ",dd,mm,yy)

Question 86

Question:

Write a program to compute simple interest given all the required values.

Solution:

principle=float(input("Enter the principle amount:"))
Question 87

Question:

Write a program to check whether a given year is a leap year or not.

Solution:

```python
time=int(input("Enter the time(years):"))
rate=float(input("Enter the rate:"))
simple_interest=(principle*time*rate)/100
print("The simple interest is:", simple_interest)
```

year=int(input("Enter year to be checked:"))
if(year%4==0 and year%100!=0 or year%400==0):
    print("The year is a leap year!")
else:
    print("The year isn't a leap year!")
```

Question 88

Question:

Write a program to compute prime factors of an integer.
Solution:

```python
n=int(input("Enter an integer:"))
print("Factors are:")
i=1
while(i<=n):
    k=0
    if(n%i==0):
        j=1
        while(j<=i):
            if(i%j==0):
                k=k+1
                j=j+1
        if(k==2):
            print(i)
    i=i+1
```

Question 89

Question:

Write a program to generate all the divisors of an integer.

Solution:

```python
n=int(input("Enter an integer:"))
```
print("The divisors of the number are:")
for i in range(1,n+1):
    if(n%i==0):
        print(i)

Question 90

Question:
Write a program to print the table of a given number.

Solution:

n=int(input("Enter the number to print the tables for:"))
for i in range(1,11):
    print(n,"x",i,"=",n*i)

Question 91

Question:
Write a program to check if a number is an Armstrong number.
**Solution:**

```python
n=int(input("Enter any number: "))
a=list(map(int,str(n)))
b=list(map(lambda x:x**3,a))
if(sum(b)==n):
    print("The number is an armstrong number. ")
else:
    print("The number isn't an armstrong number. ")
```

---

**Question 92**

**Question:**

Write a program to print the Pascal's triangle for n number of rows given by the user.

---

**Solution:**

```python
n=int(input("Enter number of rows: "))
a=[]
for i in range(n):
    a.append([])
a[i].append(1)
    for j in range(1,i):
        a[i].append(a[i-1][j-1]+a[i-1][j])
    if(n!=0):
        a[i].append(1)
```
for i in range(n):
    print("   "*(n-i),end=" ",sep=" ")
for j in range(0,i+1):
    print('{}:6'.format(a[i][j]),end=" ",sep=" ")
print()

---

Question 93

Question:

Write a program to check if a number is a Perfect number.

Solution:

n = int(input("Enter any number: "))
sum1 = 0
for i in range(1, n):
    if(n % i == 0):
        sum1 = sum1 + i
if (sum1 == n):
    print("The number is a Perfect number!")
else:
    print("The number is not a Perfect number!")
Question 94

Question:

Write a program to find the LCM of two numbers.

Solution:

```python
a = int(input("Enter the first number:"))
b = int(input("Enter the second number:"))
if (a>b):
    min1 = a
else:
    min1 = b
while (1):
    if (min1 % a == 0 and min1 % b == 0):
        print("LCM is:", min1)
        break
    min1 = min1 + 1
```

Question 95

Question:

Write a program to find the GCD of two numbers.
Solution:

```python
import math
a=int(input("Enter the first number:"))
b=int(input("Enter the second number:"))
print("The GCD of the two numbers is",math.gcd(a,b))
```

---

Question 96

Question:

Write a program to compute a polynomial equation given that the coefficients of the polynomial are stored in the list.

Solution:

```python
import math
print("Enter the coefficients of the form ax^3 + bx^2 + cx + d")
lst=[]
for i in range(0,4):
    a=int(input("Enter coefficient:"))
    lst.append(a)
x=int(input("Enter the value of x:"))
sum1=0
j=3
for i in range(0,3):
    while(j>0):
        sum1=sum1+(lst[i]*math.pow(x,j))
    break
```
Question 97

Question:

Write a program to check if two numbers are amicable numbers.

Solution:

```python
x=int(input('Enter number 1: '))
y=int(input('Enter number 2: '))
sum1=0
sum2=0
for i in range(1,x):
    if x%i==0:
        sum1+=i
for j in range(1,y):
    if y%j==0:
        sum2+=j
if(sum1==y and sum2==x):
    print('Amicable!')
else:
    print('Not Amicable!')
```
Question 98

Question:

Write a program to find the area of a triangle given all three sides.

Solution:

```python
import math

a = int(input("Enter first side: "))
b = int(input("Enter second side: "))
c = int(input("Enter third side: "))
s = (a + b + c) / 2
area = math.sqrt(s * (s - a) * (s - b) * (s - c))
print("Area of the triangle is: ", round(area, 2))
```

Question 99

Question:

Write a program to find the gravitational force acting between two objects.

Solution:

```python
m1 = float(input("Enter the first mass: "))
m2 = float(input("Enter the second mass: "))
r = float(input("Enter the distance between the centres of the masses: "))
```
\[ G = 6.673 \times (10^{-11}) \]
\[ f = \frac{G m_1 m_2}{r^2} \]
\[ \text{print}(\text{"Hence, the gravitational force is: ", round(f,2),\"N\")} \]

**Question 100**

**Question:**

Write a program to find the sum of sine series.

**Solution:**

```python
import math

def sin(x, n):
    sine = 0
    for i in range(n):
        sign = (-1)**i
        pi=22/7
        y=x*(pi/180)
        sine = sine + ((y**(2.0*i+1))/math.factorial(2*i+1))*sign
    return sine

x=int(input("Enter the value of x in degrees:"))
n=int(input("Enter the number of terms:"))
print(round(sin(x,n),2))
```
Question 101

Question:

Write a program to find the sum of cosine series.

Solution:

```python
import math
def cosine(x, n):
    cosx = 1
    sign = -1
    for i in range(2, n, 2):
        pi = 22/7
        y = x*(pi/180)
        cosx = cosx + (sign*(y**i))/math.factorial(i)
        sign = -sign
    return cosx

x = int(input("Enter the value of x in degrees:"))
n = int(input("Enter the number of terms:"))
print(round(cosine(x, n), 2))
```

Question 102

Question:

Write a program to find the sum of first N Natural Numbers.
Solution:

```python
n=int(input("Enter a number: "))
sum1 = 0
while(n > 0):
    sum1=sum1+n
    n=n-1
print("The sum of first n natural numbers is",sum1)
```

---

Question 103

**Question:**

**Write a program to find the sum of series: 1 + 1/2 + 1/3 + ..... + 1/N.**

---

**Solution:**

```python
n=int(input("Enter the number of terms: "))
sum1=0
for i in range(1,n+1):
    sum1=sum1+(1/i)
print("The sum of series is",round(sum1,2))
```
Question 104

Question:

Write a program to determine all Pythagorean triplets till the upper limit.

Solution:

```python
limit=int(input("Enter upper limit:"))
c=0
m=2
while(c<limit):
    for n in range(1,m+1):
        a=m*m-n*n
        b=2*m*n
        c=m*m+n*n
        if(c>limit):
            break
        if(a==0 or b==0 or c==0):
            break
        print(a,b,c)
    m=m+1
```

Question 105

Question:
Write a program to search the number of times a particular number occurs in a list.

Solution:

```python
a=[]
n=int(input("Enter number of elements:"))
for i in range(1,n+1):
    b=int(input("Enter element:"))
    a.append(b)
k=0
num=int(input("Enter the number to be counted:"))
for j in a:
    if(j==num):
        k=k+1
print("Number of times",num,"appears is",k)
```

Question 106

Question:

Write a program to test Collatz conjecture for a given number.

Solution:

```python
def collatz(n):
```
while n > 1:
    print(n, end=' ')
    if (n % 2):
        # n is odd
        n = 3*n + 1
    else:
        # n is even
        n = n//2
    print(1, end='')

n = int(input('Enter n: '))
print('Sequence: ', end='')
collatz(n)

Question 107

Question:

Write a program to count set bits in a number.

Solution:

def count_set_bits(n):
    count = 0
    while n:
        n &= n - 1
        count += 1
return count

n = int(input('Enter n: '))
print('Number of set bits:', count_set_bits(n))

---

Question 108

Question:

Write a program to find whether a number is a power of two.

Solution:

```python
def is_power_of_two(n):
    """Return True if n is a power of two."""
    if n <= 0:
        return False
    else:
        return n & (n - 1) == 0

n = int(input('Enter a number: '))

if is_power_of_two(n):
    print('{} is a power of two.'.format(n))
else:
    print('{} is not a power of two.'.format(n))
```
Question 109

Question:

Write a program to clear the rightmost set bit of a number.

Solution:

def clear_rightmost_set_bit(n):
    """Clear rightmost set bit of n and return it."""
    return n & (n - 1)

n = int(input('Enter a number: '))
ans = clear_rightmost_set_bit(n)
print('n with its rightmost set bit cleared equals:', ans)

Question 110

Question:

Write a program to generate all gray codes using recursion.
Solution:

```python
def get_gray_codes(n):
    """Return n-bit Gray code in a list."""
    if n == 0:
        return ['']
    first_half = get_gray_codes(n - 1)
    second_half = first_half.copy()

    first_half = ['0' + code for code in first_half]
    second_half = ['1' + code for code in reversed(second_half)]

    return first_half + second_half

n = int(input('Enter the number of bits: '))
codes = get_gray_codes(n)
print('All {}-bit Gray Codes:'.format(n))
print(codes)
```

Question 111

Question:

Write a program to convert Gray code to binary.

Solution:
def gray_to_binary(n):
    """Convert Gray codeword to binary and return it."""
    n = int(n, 2) # convert to int

    mask = n
    while mask != 0:
        mask >>= 1
        n ^= mask

    # bin(n) returns n's binary representation with a '0b' prefixed
    # the slice operation is to remove the prefix
    return bin(n)[2:]

g = input('Enter Gray codeword: ')
b = gray_to_binary(g)
print('In binary:', b)

---

**Question 112**

**Question:**

Write a program to convert binary to Gray code.

**Solution:**

def binary_to_gray(n):
    """Convert Binary to Gray codeword and return it."""
    n = int(n, 2) # convert to int
n ^= (n >> 1)

# bin(n) returns n's binary representation with a '0b' prefixed
# the slice operation is to remove the prefix
return bin(n)[2:]

g = input('Enter binary number: ')
b = binary_to_gray(g)
print('Gray codeword:', b)

---

Question 113

Question:

Write a program to replace all occurrences of 'a' with '$' in a string.

Solution:

string=input("Enter string:")
string=string.replace('a','$')
string=string.replace('A','$')
print("Modified string:")
print(string)
Question 114

Question:

Write a program to remove the n\textsuperscript{th} index character from a non-empty string.

Solution:

```python
def remove(string, n):
    first = string[:n]
    last = string[n+1:]
    return first + last

string = input("Enter the string:")
n = int(input("Enter the index of the character to remove:"))
print("Modified string:")
print(remove(string, n))
```

Question 115

Question:

Write a program to detect if two strings are anagrams.

Solution:

```python
s1 = input("Enter first string:")
s2 = input("Enter second string:")
```
if(sorted(s1)==sorted(s2)):
    print("The strings are anagrams.")
else:
    print("The strings aren't anagrams.")

Question 116

Question:
Write a program to form a string where the first character and the last character have been exchanged.

Solution:

def change(string):
    return string[-1:] + string[1:-1] + string[1:]
string=input("Enter string:")
print("Modified string:")
print(change(string))

Question 117

Question:
Write a program to count the number of vowels in a string.

Solution:

```python
string=input("Enter string:")
vowels=0
for i in string:
    if(i=='a' or i=='e' or i=='i' or i=='o' or i=='u' or i=='A' or i=='E' or i=='I' or i=='O' or i=='U'):
        vowels=vowels+1
print("Number of vowels are:")
print(vowels)
```

Question 118

Question:

Write a program to take a string and replace every blank space with a hyphen.

Solution:

```python
string=input("Enter string:")
string=string.replace(' ','-')
print("Modified string:")
print(string)
```
Question 119

Question:

Write a program to calculate the length of a string without using library functions.

Solution:

```python
string = input("Enter string:")
count = 0
for i in string:
    count = count + 1
print("Length of the string is:"),
print(count)
```

Question 120

Question:

Write a program to remove the characters of odd index values in a string.

Solution:
```python
def modify(string):
    final = ""
    for i in range(len(string)):
        if i % 2 == 0:
            final += string[i]
    return final
string=input("Enter string:")
print("Modified string is:")
print(modify(string))
```

---

**Question 121**

**Question:**

Write a program to calculate the number of words and characters present in a string.

**Solution:**

```python
string=input("Enter string:")
char=0
word=1
for i in string:
    char+=1
    if(i==' '):
        word+=1
print("Number of words in the string:")
print(word)
print("Number of characters in the string:")
```
Question 122

Question:

Write a program to take in two strings and display the larger string without using built-in functions.

Solution:

```python
string1 = input("Enter first string:")
string2 = input("Enter second string:")
count1 = 0
count2 = 0
for i in string1:
    count1 = count1 + 1
for j in string2:
    count2 = count2 + 1
if (count1 < count2):
    print("Larger string is:")
    print(string2)
elif (count1 == count2):
    print("Both strings are equal.")
else:
    print("Larger string is:")
    print(string1)
```
Question 123

Question:

Write a program to count number of lowercase characters in a string.

Solution:

```python
string = input("Enter string:")
count = 0
for i in string:
    if i.islower():
        count = count + 1
print("The number of lowercase characters is:")
print(count)
```

Question 124

Question:

Write a program to count the number of lowercase letters and uppercase letters in a string.
**Solution:**

```python
string = input("Enter string:")
count1 = 0
count2 = 0
for i in string:
    if i.islower():
        count1 = count1 + 1
    elif i.isupper():
        count2 = count2 + 1
print("The number of lowercase characters is:")
print(count1)
print("The number of uppercase characters is:")
print(count2)
```

---

**Question 125**

**Question:**

Write a program to calculate the number of digits and letters in a string.

**Solution:**

```python
string = input("Enter string:")
count1 = 0
count2 = 0
for i in string:
    if i.isdigit():
        count1 = count1 + 1
    count2 = count2 + 1
```
print("The number of digits is:")
print(count1)
print("The number of characters is:")
print(count2)

Question 126

Question:
Write a program to form a new string made of the first 2 characters and last 2 characters from a given string.

Solution:

string=input("Enter string:")
count=0
for i in string:
    count=count+1
new=string[0:2]+string[count-2:count]
print("Newly formed string is:")
print(new)

Question 127

Question:
Write a program to count the occurrences of each word in a given string sentence.

Solution:

```python
string = input("Enter string:")
word = input("Enter word:")
a = []
count = 0
a = string.split(" ")
for i in range(0, len(a)):
    if word == a[i]:
        count = count + 1
print("Count of the word is:")
print(count)
```

Question 128

Question:

Write a program to check if a substring is present in a given string.

Solution:

```python
string = input("Enter string:")
sub_str = input("Enter word:")
if (string.find(sub_str) == -1):
    print("Substring not found in string!")
```
else:
    print("Substring in string!")

Question 129

Question:

Write a program to print all permutations of a string in lexicographic order without using recursion.

Solution:

```python
from math import factorial

def print_permutations_lexicographic_order(s):
    """Print all permutations of string s in lexicographic order."""
    seq = list(s)

    # there are going to be n! permutations where n = len(seq)
    for _ in range(factorial(len(seq))):
        # print permutation
        print(''.join(seq))

        # find p such that seq[p:] is the largest sequence with elements in
        # descending lexicographic order
        p = len(seq) - 1
        while p > 0 and seq[p - 1] > seq[p]:
            p -= 1
```

332
# reverse seq[p:]
seq[p:] = reversed(seq[p:.utils])

if p > 0:
    # find q such that seq[q] is the smallest element in seq[p:] such that
    # seq[q] > seq[p - 1]
    q = p
    while seq[p - 1] > seq[q]:
        q += 1

    # swap seq[p - 1] and seq[q]
    seq[p - 1], seq[q] = seq[q], seq[p - 1]

s = input('Enter the string: ')
print_permutations_lexicographic_order(s)

---

**Question 130**

**Question:**

Write a program to print all permutations of a string in lexicographic order using recursion.

**Solution:**

```python
from math import factorial

def print_permutations_lexicographic_order(s):
```
Print all permutations of string s in lexicographic order.

```python
seq = list(s)
for _ in range(factorial(len(seq))):
    print(''.join(seq))
    nxt = get_next_permutation(seq)
    if nxt is None:
        # then reverse it
        seq.reverse()
    else:
        seq = nxt

def get_next_permutation(seq):
    """Return next greater lexicographic permutation. Return None if cannot.
    This will return the next greater permutation of seq in lexicographic
    order. If seq is the highest permutation then this will return None.
    seq is a list.
    """
    if len(seq) == 0:
        return None

    nxt = get_next_permutation(seq[1:])
    if nxt is None:
        # reverse seq[1:], so that seq[1:] now is in ascending order
        seq[1:] = reversed(seq[1:])
        # find q such that seq[q] is the smallest element in seq[1:] such that
        # seq[q] > seq[0]
        q = 1
        while q < len(seq) and seq[0] > seq[q]:
            q += 1
        # if cannot find q, then seq is the highest permutation
```
if q == len(seq):
    return None

# swap seq[0] and seq[q]
seq[0], seq[q] = seq[q], seq[0]

return seq
else:
    return [seq[0]] + nxt

s = input('Enter the string: ')
print_permutations_lexicographic_order(s)

---

**Question 131**

**Question:**

Write a program to add a key-value pair to a dictionary.

**Solution:**

```python
key=int(input("Enter the key (int) to be added:"))
value=int(input("Enter the value for the key to be added:"))
d={}
d.update({key:value})
print("Updated dictionary is:")
print(d)
```
Question 132

Question:

Write a program to concatenate two dictionaries into one dictionary.

Solution:

d1={'A':1,'B':2}
d2={'C':3}
d1.update(d2)
print("Concatenated dictionary is:")
print(d1)

Question 133

Question:

Write a program to check if a given key exists in a dictionary or not.

Solution:

d={'A':1,'B':2,'C':3}
key=input("Enter key to check:")
```python
if key in d.keys():
    print("Key is present and value of the key is:"")
    print(d[key])
else:
    print("Key isn't present!")
```

---

**Question 134**

**Question:**

Write a program to find the sum all the items in a dictionary.

**Solution:**

```python
d={'A':100,'B':540,'C':239}
print("Total sum of values in the dictionary:"")
print(sum(d.values()))
```

---

**Question 135**

**Question:**

Write a program to multiply all the items in a dictionary.
Solution:

d={'A':10,'B':10,'C':239}
tot=1
for i in d:
    tot=tot*d[i]
print(tot)

Question 136

Question:

Write a program to remove the given key from a dictionary.

Solution:

d = {'a':1,'b':2,'c':3,'d':4}
print("Initial dictionary")
print(d)
key=input("Enter the key to delete(a-d):")
if key in d:
    del d[key]
else:
    print("Key not found!")
    exit(0)
print("Updated dictionary")
Question 137

Question:

Write a program to form a dictionary from an object of a class.

Solution:

class A(object):
    def __init__(self):
        self.A=1
        self.B=2

obj=A()
pren(obj.__dict__)
**Solution:**

```python
keys=[]
values=[]
n=int(input("Enter number of elements for dictionary:"))
print("For keys:")
for x in range(0,n):
    element=int(input("Enter element" + str(x+1) + ":"))
    keys.append(element)
print("For values:")
for x in range(0,n):
    element=int(input("Enter element" + str(x+1) + ":"))
    values.append(element)
d=dict(zip(keys,values))
print("The dictionary is:")
print(d)
```

---

**Question 139**

**Question:**

Write a program to count the frequency of words appearing in a string using a dictionary.

---

**Solution:**

```python
test_string=input("Enter string:")
l=[]
l=test_string.split()
```
Question 140

Question:

Write a program to create a dictionary with key as first character and value as words starting with that character.

Solution:

test_string = input("Enter string:")
l = test_string.split()
d = {}
for word in l:
    if word[0] not in d.keys():
        d[word[0]] = []
        d[word[0]].append(word)
    else:
        if word not in d[word[0]]:
            d[word[0]].append(word)
for k, v in d.items():
    print(k, ":", v)
Question 141

Question:
Write a program to count the number of vowels present in a string using sets.

Solution:

```python
s = input("Enter string:")
count = 0
vowels = set("aeiou")
for letter in s:
    if letter in vowels:
        count += 1
print("Count of the vowels is:" + count)
```

Question 142

Question:
Write a program to check common letters in the two input strings.

Solution:

```python
s1 = input("Enter first string:" + count)
```
s2=input("Enter second string:")
a=list(set(s1)&set(s2))
print("The common letters are:")
for i in a:
    print(i)

Question 143

Question:

Write a program to display which letters are in the first string but not in the second string.

Solution:

s1=input("Enter first string:")
s2=input("Enter second string:")
a=list(set(s1)-set(s2))
print("The letters are:")
for i in a:
    print(i)

Question 144

Question:
Write a program to display which letters is present in both the strings.

Solution:

```python
s1 = input("Enter first string:")
s2 = input("Enter second string:")
a = list(set(s1) | set(s2))
print("The letters are:"
for i in a:
    print(i)
```

Question 145

Question:

Write a program to determine whether a given number is even or odd recursively.

Solution:

```python
def check(n):
    if (n < 2):
        return (n % 2 == 0)
    return (check(n - 2))
n = int(input("Enter number:"))
if (check(n) == True):
    print("Number is even!")
else:
    print("Number is odd!")
```
Question 146

Question:

Write a program to determine how many times a given letter occurs in a string recursively.

Solution:

```python
def check(string, ch):
    if not string:
        return 0
    elif string[0] == ch:
        return 1 + check(string[1:], ch)
    else:
        return check(string[1:], ch)

string = input("Enter string:")
ch = input("Enter character to check:")
print("Count is:")
print(check(string, ch))
```
Question 147

Question:

Write a program to find the fibonacci series using recursion.

Solution:

def fibonacci(n):
    if(n <= 1):
        return n
    else:
        return(fibonacci(n-1) + fibonacci(n-2))

n = int(input("Enter number of terms:"))
print("Fibonacci sequence:")
for i in range(n):
    print (fibonacci(i))

Question 148

Question:

Write a program to find the factorial of a number using recursion.

Solution:
def factorial(n):
    if n <= 1:
        return 1
    else:
        return n*factorial(n-1)

n = int(input("Enter number:"))
print("Factorial:")
print(factorial(n))

---

Question 149

Question:

Write a program to find the sum of elements in a list recursively.

Solution:

def sum_arr(arr, size):
    if size == 0:
        return 0
    else:
        return arr[size-1] + sum_arr(arr, size-1)

n=int(input("Enter the number of elements for list:"))
a=[]
for i in range(0,n):
    element=int(input("Enter element:"))
    a.append(element)
print(\"The list is:\")
print(a)
print(\"Sum of items in list:\")
b=sum_arr(a,n)
print(b)

Question 150

Question:

Write a program to find the binary equivalent of a number recursively.

Solution:

l=[]
def convert(b):
    if(b==0):
        return l
    dig=b%2
    l.append(dig)
    convert(b//2)
a=int(input("Enter a number: "))
convert(a)
l.reverse()
print("Binary equivalent:")
for i in l:
    print (i)
Question 151

Question:

Write a program to find the LCM of two numbers using recursion.

Solution:

```python
def lcm(a, b):
    lcm.multiple = lcm.multiple + b
    if ((lcm.multiple % a == 0) and (lcm.multiple % b == 0)):
        return lcm.multiple;
    else:
        lcm(a, b)
    return lcm.multiple
lcm.multiple = 0
a = int(input("Enter first number:"))
b = int(input("Enter second number:

if (a > b):
    LCM = lcm(b, a)
else:
    LCM = lcm(a, b)
print(LCM)
```

Question 152

Question:
Write a program to find the GCD of two numbers using recursion.

Solution:

def gcd(a, b):
    if b == 0:
        return a
    else:
        return gcd(b, a % b)
a = int(input("Enter first number:"))
b = int(input("Enter second number:"))
GCD = gcd(a, b)
print("GCD is: ")
print(GCD)

Question 153

Question:

Write a program to read the contents of a file.

Solution:

a = str(input("Enter the name of the file with .txt extension:"))
file2 = open(a, 'r')
line = file2.readline()
while(line!=""):
    print(line)
    line=file2.readline()
file2.close()

Question 154

Question:

Write a program to count the number of words in a text file.

Solution:

fname = input("Enter file name: ")

num_words = 0

with open(fname, 'r') as f:
    for line in f:
        words = line.split()
        num_words += len(words)
print("Number of words:")
print(num_words)
Question 155

Question:

Write a program to count the number of lines in a text file.

Solution:

```python
fname = input("Enter file name: ")
num_lines = 0
with open(fname, 'r') as f:
    for line in f:
        num_lines += 1
print("Number of lines:")
print(num_lines)
```

Question 156

Question:

Write a program to read a string from the user and appends it into a file.

Solution:

```python
fname = input("Enter file name: ")
file3=open(fname,"a")
```
```python
c=input("Enter string to append: \n");
file3.write("\n")
file3.write(c)
file3.close()
print("Contents of appended file:");
file4=open(fname,'r')
line1=file4.readline()
while(line1!=""):
    print(line1)
    line1=file4.readline()
file4.close()
```

---

**Question 157**

**Question:**

Write a program to count the occurrences of a word in a text file.

**Solution:**

```python
fname = input("Enter file name: ")
word=input("Enter word to be searched:")
k = 0

with open(fname, 'r') as f:
    for line in f:
        words = line.split()
        for i in words:
            if(i==word):
```


```python
k = k + 1
print("Occurrences of the word:")
print(k)
```

---

**Question 158**

**Question:**

Write a program to copy the contents of one file into another.

**Solution:**

```python
with open("test.txt") as f:
    with open("out.txt", "w") as f1:
        for line in f:
            f1.write(line)
```

---

**Question 159**

**Question:**

Write a program to count the occurrences of a letter in a text file.
Solution:

```python
fname = input("Enter file name: ")
l=input("Enter letter to be searched:")
k = 0

with open(fname, 'r') as f:
    for line in f:
        words = line.split()
        for i in words:
            for letter in i:
                if(letter==l):
                    k=k+1

print("Occurrences of the letter:")
print(k)
```

Question 160

Question:

Write a program to read a text file and print all numbers present in the text file.

Solution:

```python
fname = input("Enter file name: ")

with open(fname, 'r') as f:
    for line in f:
        words = line.split()
```
for i in words:
    for letter in i:
        if(letter.isdigit()):
            print(letter)

Question 161

Question:

Write a program to append the contents of one file to another file.

Solution:

name1 = input("Enter file to be read from: ")
name2 = input("Enter file to be appended to: ")
fin = open(name1, "r")
data2 = fin.read()
fin.close()
fout = open(name2, "a")
fout.write(data2)
fout.close()
Write a program to count the number of blank spaces in a text file.

Solution:

```python
fname = input("Enter file name: ")
k = 0

with open(fname, 'r') as f:
    for line in f:
        words = line.split()
        for i in words:
            for letter in i:
                if(letter.isspace):
                    k=k+1

print("Occurrences of blank spaces:")
print(k)
```

Question 163

Question:

Write a program to read a file and capitalize the first letter of every word in the file.

Solution:

```python
fname = input("Enter file name: ")
```
```python
with open(fname, 'r') as f:
    for line in f:
        l = line.title()
        print(l)
```

**Question 164**

**Question:**

Write a program to read the contents of a file in reverse order.

**Solution:**

```python
filename = input("Enter file name: ")
for line in reversed(list(open(filename))):
    print(line.rstrip())
```

**Question 165**

**Question:**

Write a program to find the area of a rectangle using classes.
Solution:

class rectangle():
    def __init__(self,breadth,length):
        self.breadth=breadth
        self.length=length
    def area(self):
        return self.breadth*self.length

da=int(input("Enter length of rectangle: "))
b=int(input("Enter breadth of rectangle: "))
obj=rectangle(a,b)
print("Area of rectangle:",obj.area())

print()

Question 166

Question:

Write a program to append, delete and display elements of a list using classes.

Solution:

class check():
    def __init__(self):
        self.n=[]
    def add(self,a):
        self.n.append(a)
    def delete(self,a):
        self.n.remove(a)
    def display(self):
        print("Elements of list:",self.n)
def remove(self, b):
    self.n.remove(b)

def dis(self):
    return (self.n)

obj = check()

choice = 1
while choice != 0:
    print("0. Exit")
    print("1. Add")
    print("2. Delete")
    print("3. Display")

    choice = int(input("Enter choice: "))

    if choice == 1:
        n = int(input("Enter number to append: "))
        obj.add(n)
        print("List: ", obj.dis())

    elif choice == 2:
        n = int(input("Enter number to remove: "))
        obj.remove(n)
        print("List: ", obj.dis())

    elif choice == 3:
        print("List: ", obj.dis())

    elif choice == 0:
        print("Exiting!")

    else:
        print("Invalid choice!!")

print()
Question 167

Question:

Write a program to implement a binary heap.

Solution:

```python
class BinaryHeap:
    def __init__(self):
        self.items = []

    def size(self):
        return len(self.items)

    def parent(self, i):
        return (i - 1)//2

    def left(self, i):
        return 2*i + 1

    def right(self, i):
        return 2*i + 2

    def get(self, i):
        return self.items[i]

    def get_max(self):
        if self.size() == 0:
            return None
        return self.items[0]

    def extract_max(self):
```

```python
```
```python
if self.size() == 0:
    return None

largest = self.get_max()

self.items[0] = self.items[-1]
del self.items[-1]

self.max_heapify(0)

return largest

def max_heapify(self, i):
    l = self.left(i)
    r = self.right(i)

    if (l <= self.size() - 1 and self.get(l) > self.get(i)):
        largest = l
    else:
        largest = i

    if (r <= self.size() - 1 and self.get(r) > self.get(largest)):
        largest = r

    if (largest != i):
        self.swap(largest, i)
        self.max_heapify(largest)

def swap(self, i, j):
    self.items[i], self.items[j] = self.items[j], self.items[i]

def insert(self, key):
    index = self.size()
    self.items.append(key)

    while (index != 0):
        p = self.parent(index)
        if self.get(p) < self.get(index):
            self.swap(p, index)
        index = p

bheap = BinaryHeap()
```

print('Menu')
print('insert <data>,'
print('max get')
print('max extract')
print('quit')

while True:
    do = input('What would you like to do? ').split()

    operation = do[0].strip().lower()
    if operation == 'insert':
        data = int(do[1])
        bheap.insert(data)
    elif operation == 'max':
        suboperation = do[1].strip().lower()
        if suboperation == 'get':
            print('Maximum value: {}'.format(bheap.get_max()))
        elif suboperation == 'extract':
            print('Maximum value removed: {}'.format(bheap.extract_max()))
    elif operation == 'quit':
        break

Question 168

Question:
Write a program to implement a binomial tree.

Solution:
class BinomialTree:
    def __init__(self, key):
        self.key = key
        self.children = []
        self.order = 0

    def add_at_end(self, t):
        self.children.append(t)
        self.order = self.order + 1

trees = []

print('Menu')
print('create <key>')
print('combine <index1> <index2>')
print('quit')

while True:
    do = input('What would you like to do? ').split()

    operation = do[0].strip().lower()

    if operation == 'create':
        key = int(do[1])
        btree = BinomialTree(key)
        trees.append(btree)
        print('Binomial tree created.')

    elif operation == 'combine':
        index1 = int(do[1])
        index2 = int(do[2])
        if trees[index1].order == trees[index2].order:
            trees[index1].add_at_end(trees[index2])
            del trees[index2]
            print('Binomial trees combined.')
        else:
            print('Orders of the trees need to be the same.')
elif operation == 'quit':
    break

print('{:>8}{:>12}{:>8}'.format('Index', 'Root key', 'Order'))
for index, t in enumerate(trees):
    print('{:8d}{:12d}{:8d}'.format(index, t.key, t.order))

---

Question 169

Question:

Write a program to implement Tower of Hanoi.

Solution:

def hanoi(disks, source, auxiliary, target):
    if disks == 1:
        print('Move disk 1 from peg {} to peg {}.' .format(source, target))
        return

    hanoi(disks - 1, source, target, auxiliary)
    print('Move disk {} from peg {} to peg {}.' .format(disks, source, target))
    hanoi(disks - 1, auxiliary, source, target)

disks = int(input('Enter number of disks: '))
hanoi(disks, 'A', 'B', 'C')
Question 170

Question:

Write a program to implement birthday dictionary.

Solution:

```python
if __name__ == '__main__':

    birthdays = {
        'Albert Einstein': '03/14/1879',
        'Benjamin Franklin': '01/17/1706',
        'Ada Lovelace': '12/10/1815',
        'Donald Trump': '06/14/1946',
        'Rowan Atkinson': '01/6/1955'}

    print('Welcome to the birthday dictionary. We know the birthdays of:')
    for name in birthdays:
        print(name)

    print('Who\'s birthday do you want to look up?')
    name = input()
    if name in birthdays:
        print('{}\'s birthday is {}.format(name, birthdays[name]))
    else:
        print('Sadly, we don\'t have {}\'s birthday.'.format(name))
```
Question 171

Question:

Write a program to implement guess letters.

Solution:

```python
if __name__ == '__main__':
    print("Welcome to hangman!!")
    word = "EVAPORATE"
    guessed = "_" * len(word)
    word = list(word)
    guessed = list(guessed)
    lstGuessed = []
    letter = input("guess letter: ")
    while True:
        if letter.upper() in lstGuessed:
            letter = ''
            print("Already guessed!!")
        elif letter.upper() in word:
            index = word.index(letter.upper())
            guessed[index] = letter.upper()
            word[index] = '_'
        else:
            print(''.join(guessed))
            if letter is not '':
                lstGuessed.append(letter.upper())
                letter = input("guess letter: ")

if '_' not in guessed:
    print("You won!!")
```
Question 172

Question:

Write a program to implement password generator.

Solution:

```python
import random

s = "abcdefghijklmnopqrstuvwxyz01234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ!@#$%^&*()?"
passlen = 8
p = ".join(random.sample(s,passlen ))
print (p)
```

Question 173

Question:

Write a program to display calendar of the given month and year.
Solution:

```python
# importing calendar module
import calendar

yy = 2014  # year
mm = 11    # month

# To take month and year input from the user
# yy = int(input("Enter year: "))
# mm = int(input("Enter month: 

# display the calendar
print(calendar.month(yy, mm))
```

---

**Question 174**

**Question:**

Write a program to add two matrices.

**Solution:**

```python
X = [[12, 7, 3],
     [4, 5, 6],
     [7, 8, 9]]

Y = [[5, 8, 1],
     [6, 7, 3],
```
Question 175

Question:

Write a program to transpose a matrix.

Solution:

\[
X = \begin{bmatrix}
12 & 7 \\
4 & 5 \\
3 & 8
\end{bmatrix}
\]

result = \[
\begin{bmatrix}
0 & 0 \\
0 & 0 \\
0 & 0
\end{bmatrix}
\]

# iterate through rows
for i in range(len(X)):
    # iterate through columns
    for j in range(len(X[0])):
        result[i][j] = X[i][j] + Y[i][j]

for r in result:
    print(r)
# iterate through rows
for i in range(len(X)):
    # iterate through columns
    for j in range(len(X[0])):
        result[j][i] = X[i][j]

for r in result:
    print(r)

---

**Question 176**

**Question:**

***Write a program to multiply two matrices.***

**Solution:**

# 3x3 matrix
X = [[12, 7, 3],
     [4, 5, 6],
     [7, 8, 9]]

# 3x4 matrix
Y = [[5, 8, 1, 2],
     [6, 7, 3, 0],
     [4, 5, 9, 1]]

# result is 3x4
result = [[0, 0, 0, 0],
          [0, 0, 0, 0],
          [0, 0, 0, 0]]
# iterate through rows of X
for i in range(len(X)):
    # iterate through columns of Y
    for j in range(len(Y[0])):
        # iterate through rows of Y
        for k in range(len(Y)):
            result[i][j] += X[i][k] * Y[k][j]

for r in result:
    print(r)

---

**Question 177**

**Question:**

Write a program to remove punctuations from a string.

**Solution:**

```python
# define punctuation
punctuations = '''!()-[]{};:'"\,<>./?@#$%^&*_~''

my_str = "Hello!!!, he said ---and went."

# To take input from the user
# my_str = input("Enter a string: ")
```
# remove punctuation from the string
no_punct = ""
for char in my_str:
    if char not in punctuations:
        no_punct = no_punct + char

# display the unpunctuated string
print(no_punct)

---

**Question 178**

**Question:**

Write a program to find the hash of a file and display it.

**Solution:**

```python
import hashlib

def hash_file(filename):
    """This function returns the SHA-1 hash of the file passed into it""

    # make a hash object
    h = hashlib.sha1()

    # open file for reading in binary mode
    with open(filename, 'rb') as file:
        # loop till the end of the file
```
chunk = 0
while chunk != b'':
    # read only 1024 bytes at a time
    chunk = file.read(1024)
    h.update(chunk)

    # return the hex representation of digest
return h.hexdigest()

message = hash_file("languages.txt")
print(message)

---

Question 179

**Question:**

Write a program to find the size (resolution) of a image.

**Solution:**

```python
def jpeg_res(filename):
    """This function prints the resolution of the jpeg image file passed into it""

    # open image for reading in binary mode
    with open(filename,'rb') as img_file:

        # height of image (in 2 bytes) is at 164th position
        img_file.seek(163)

        # read the 2 bytes
```
a = img_file.read(2)

# calculate height
height = (a[0] << 8) + a[1]

# next 2 bytes is width
a = img_file.read(2)

# calculate width
width = (a[0] << 8) + a[1]

print("The resolution of the image is", width, "x", height)

jpeg_res("img1.jpg")

---

Question 180

Question:

Write a program to read website source code.

Solution:

import sys

if sys.version_info[0] == 3:
    from urllib.request import urlopen
else:
    # Not Python 3 - today, it is most likely to be Python 2
    # But note that this might need an update when Python 4
from urllib import urlopen

# Your code where you can use urlopen
with urlopen("http://www.myw3schools.com") as url:
    s = url.read()

print(s)

Question 181

Question:

Write a program to get IP address of your computer.

Solution:

import socket
hostname = socket.gethostname()
IPAddr = socket.gethostbyname(hostname)
print("Your Computer Name is:" + hostname)
print("Your Computer IP Address is:" + IPAddr)

Question 182

Question:
Write a program to get all links from a webpage.

Solution:

```python
from bs4 import BeautifulSoup
from urllib.request import Request, urlopen

req = Request("http://www.myw3schools.com")
html_page = urlopen(req)

soup = BeautifulSoup(html_page, "lxml")

links = []
for link in soup.findAll('a'):
    links.append(link.get('href'))

print(links)
```

Question 183

Question:

Write a program to illustrate Dice Roll Simulator.

Solution:

```python
import random
min = 1
max = 6
roll_again = "yes"

while roll_again == "yes" or roll_again == "y":
    print("Rolling the dices...")
    print("The values are...")
    print(random.randint(min, max))
    print(random.randint(min, max))

    roll_again = input("Roll the dices again?")
```
Question 1

Question:

Write a program to print Hello World!.

Solution:

```php
<?php
    echo "Hello World!";
?>
```

Question 2

Question:

Write a program to add two numbers.

Solution:
Question 3

Question:

Write a program to subtract two numbers.

Solution:

```php
<?php
    $num1=5;
    $num2=1;
    $sub = $num1 - $num2;
    echo "difference of the two numbers is : \$sub";
?>
```
Question 4

Question:

Write a program to divide two numbers.

Solution:

```php
<?php
$num1=6;
$num2=2;
$div =$num1 / $num2;
echo "the division of two numbers is : $div";
?>
```
Write a program to multiply two numbers.

Solution:

```php
$num1 = 6;
$num2 = 2;

$mult = $num1 * $num2;

echo "the product of two numbers is : $mult";
?>
```

Question 6

Question:

Write a program to find the area of a circle.

Solution:

```php
$radius = 2.0;

$pi = 3.14159;

$area = $pi * $radius * $radius;

echo("<br> radius = $radius centimeter");

echo("<br> area = $area centimeter square");
```
Question 7

Question:

Write a program to find the square root of a number.

Solution:

```php
<?php
$num1 = 4.0;
$num2 = sqrt($num1);
echo("The square root of a number = $num2");
?>
```
Question 8

Question:

Write a program to find the cube root of a number.

Solution:

```php
<?php

$num1 = 6.0;
$num2 = pow(($num1), 1/3);

echo("The cube root of a number = $num2");

?>
```

Question 9

Question:

Write a program to round off a number.

Solution:

```php
<?php
```
num1 = 4.5;
num2 = round (num1);
echo("The round off of a number = num2");
?>

---

Question 10

Question:

Write a program to find the incremented and decremented values of two numbers.

Solution:

```php
<?php
$num1 = 2;
$num2 = 3;
$num3 = $num1 + 1;
$num4 = $num1 - 1;
$num5 = $num2 + 1;
$num6 = $num2 - 1;
echo ("The incremented value of $num1 = $num3 ");
echo ("The decremented value of $num1 = $num4 ");
echo ("The incremented value of $num2 = $num5 ");
echo ("The decremented value of $num2 = $num6 ");
?>
```
Question 11

Question:

Write a program to find the greatest of two numbers using if – else statement.

Solution:

```php
<?php
$x = 4.5;
y=5;
if($x>$y){
    echo " x is greater than y";
} else {
    echo " y is greater than x"
}
?>
```

Question 12

Question:
Write a program to print the first ten natural numbers using for loop statement.

Solution:

```php
<?php
for ($i=1; $i<=10; $i++)
    echo ($i);
?>
```

Question 13

Question:

Write a program to print the first ten natural numbers using while loop statement.

Solution:

```php
<?php
$i = 1;
while($i <= 10) {
    echo "$i ";
    $i++;
}
?>
```
Question 14

Question:

Write a program to print the first nine natural numbers using do while loop statement.

Solution:

```php
<?php
$i = 1;
do {
    echo "$i ";
    $i++;
} while ($i <= 9);
?>
```
Question 15

Question:

Write a program to print the average of the first 10 numbers using for loop statement.

Solution:

```php
<?php

$i;
$avg;
$sum = 0;
for( $i=1; $i<=10; $i++)
$sum = $sum + $i;
$avg = $sum/10;
echo "sum of the first 10 numbers = $sum ";
echo "average of the first 10 numbers = $avg ";
?>
```

Question 16

Question:
Write a program to add two numbers using PHP function.

Solution:

```php
<?php
function addition($a, $b) {
    return $a + $b;
}

$sum = addition(4, 3);

echo "the sum of two numbers = $sum ";
?>
```

Question 17

Question:

Write a program to display the current date and time.

Solution:

```php
<?php
```
Question 18

Question:

Write a program to calculate area of a triangle with base as 20 and height as 25.

Solution:

```php
<?php
$base = 10;
$height = 15;
echo "area = " . ($base * $height) / 2;
?>
```
Question 19

Question:

Write a program to convert temperature from fahrenheit to celsius.

Solution:

```php
<?php

$far = 89;
$cel=((far - 32) * (5/9));
echo "Temperature in Fahrenheit is : $far". "<br />
echo "Temperature in Celcius is : $cel" ;
?>
```

Question 20

Question:

Write a program to print factorial of a number.

Solution:
<?php
$num = 5;
$fact = 1;
for ($x=$num; $x>=1; $x--)
{
   $fact = $fact * $x;
}
echo "$fact";
?>

Question 21

Question:

Write a program to reverse a given string.

Solution:

<?php
$str = "JavaScript";
.len = strlen($str);
for ($i=(len-1) ; $i >= 0 ; $i--)
{
   echo $str[$i];
}
?>
Question 22

Question:

Write a program to count the number of vowels in a given string.

Solution:

```php
<?php
function vowel_Count($string)
{
    preg_match_all('/[aeiou]/i', $string, $matches);
    return count($matches[0]);
}
print_r(vowel_Count('Python'));
?>
```
Question 23

Question:

Write a program to find sum of array elements.

Solution:

```php
<?php
$a = array(1, 2, 3, 4);
echo array_sum($a);
?>
```

Question 24

Question:

Write a program to check whether a number is Even or Odd.

Solution:

```php
<?php
$num = 253;
if ($num % 2 == 0)
```
Question 25

Question:

Write a program to list the first 15 prime numbers.

Solution:

```php
<?php
$count = 0;
$num = 2;
while ($count < 15 )
{
    $div=0;
    for ( $i=1; $i<=$num; $i++ )
    {
        if (($num%$i)==0)
        {
            $div++;
        }
    }
    if ($div==1 )
        {
            echo "$num is Prime Number"
        }
    else
        {
            echo "$num is Even Number"
        }
    
    $count++;
}?>
```
if ($div<3)
{
   echo $num." 
</br ";
   $count=$count+1;
}
$num=$num+1;
?>

---

**Question 26**

**Question:**

Write a program to reverse a given number.

**Solution:**

```php
<?php
$num = 253;
$rev = 0;
while ($num > 1)
{
   $rem = $num % 10;
   $rev = ($rev * 10) + $rem;
   $num = ($num / 10);
}
```
Question 27

Question:

Write a program to check whether 507 is Armstrong or not.

Solution:

```php
<?php
$num = 507;
$total = 0;
$x = $num;
while ($x != 0) {
    $rem = $x % 10;
    $total += $rem * $rem * $rem;
    $x = intdiv($x, 10);
}
if ($num == $total) {
    echo "Armstrong number";
} else {
    echo "No it is not an armstrong number";
}
```
Question 28

Question:

Write a program to print table of a number.

Solution:

```php
<?php
define('a', 7);
for($i=1; $i<=10; $i++)
{
    echo "7 * $i = ", $i*a;
    echo '<br>';
}
?>
```
Question 29

Question:

Write a program to print the first 12 numbers of a Fibonacci series.

Solution:

```php
<?php
$num = 0;
$n1 = 0;
$n2 = 1;
echo $n1.' '.$n2.' '; 
while ($num < 10 )
{
    $n3 = $n2 + $n1;
    echo $n3.' '; 
    $n1 = $n2;
    $n2 = $n3;
    $num = $num + 1;
}
?>
```

Question 30

Question:
Write a program to swap two numbers 65 and 98 using a third variable.

Solution:

```php
<?php
$a = 65;
b = 98;
c = $a;
a = $b;
b = $c;
echo "After swapping:<br><br>";
echo "$a =".$a."  b=".$b;
?>
```

Question 31

Question:

Write a program to get the PHP version and configuration information.

Solution:

```php
<?php
phpinfo();
?>
```
Question 32

Question:

Write a program to Get the client IP address.

Solution:

```php
<?php
//whether ip is from share internet
if (!empty($_SERVER['HTTP_CLIENT_IP']))
{
    $ip_address = $_SERVER['HTTP_CLIENT_IP'];
}
//whether ip is from proxy
elseif (!empty($_SERVER['HTTP_X_FORWARDED_FOR']))
{
    $ip_address = $_SERVER['HTTP_X_FORWARDED_FOR'];
}
//whether ip is from remote address
else
{
    $ip_address = $_SERVER['REMOTE_ADDR'];
}
echo $ip_address;
?>
```

Question 33

Question:
Write a program to Check whether the page is called from 'https' or 'http'.

Solution:

```php
<?php
if (!empty($_SERVER['HTTPS'])) {
    echo 'https is enabled';
} else {
    echo 'http is enabled'."\n";
}
?>
```

Question 34

Question:

Write a program to Display source code of a webpage.

Solution:

```php
<?php
$all_lines = file('https://www.w3resource.com/');
foreach ($all_lines as $line_num => $line) {
    echo "Line No.-{$line_num}: " . htmlspecialchars($line) . "\n";
}
```
Question 35

Question:

Write a program to delay the program execution for the given number of seconds.

Solution:

```php
<?php
// current time
echo date('h:i:s') . "\n";
// sleep for 5 seconds
sleep(5);
// wake up
echo date('h:i:s')."\n";
?>
```

---

JavaScript Exercises

Question 1

Question:

Write a program to print Hello World!. 
Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
    document.write("Hello World!");
</script>
</body>
</html>
```

Question 2

Question:

Write a program to add two numbers.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<p>A typical addition operation adds two numbers and produces a new number.</p>
<script>
    var x ;
    var y;
    var z;
```
x = 100;
y = 200;
z = x + y;
document.write(" The sum of two numbers is: ", z);
</script>
</body>
</html>

Question 3

Question:

Write a program to subtract two numbers.

Solution:

<!DOCTYPE html>
<html>
<body>
<p>A typical subtraction operation subtracts two numbers and produces a new number.</p>
<script>
var x;
var y;
var z;
x = 300;
y = 200;
z = x - y;
document.write(" The difference of two numbers is: ", z);
</script>
Question 4

Question:

Write a program to divide two numbers.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<p>A typical division operation divides two numbers and produces a new number.</p>
<script>
var x;
var y;
var z;
x=300;
y = 200;
z = x/y;
document.write(" The division of two numbers is: "+ z);
</script>
</body>
</html>
```
Question 5

Question:

Write a program to multiply two numbers.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<p>A typical multiplication operation multiplies two numbers and produces a new number.</p>
<script>
var x;
var y;
var z;
x=300;
y = 200;
z = x* y;
document.write(" The multiplication of two numbers is:   ");
</script>
</body>
</html>
```
Question 6

Question:

Write a program to find the area of a circle.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var r;
var area;
r=3;
area = 3.14*r*r;
document.write(" The area of the circle is: " + area +" centimeter square");
</script>
</body>
</html>
```

Question 7

Question:

Write a program to find the square root of a number.
Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var x;
var z;
x=4;
z = Math.sqrt(x);
document.write(" The square root of a number z is: " + z);
</script>
</body>
</html>
```

Question 8

Question:

Write a program to find the cube root of a number.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
</body>
</html>
```
Question 9

Question:

Write a program to round off a number.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var x ;
var z;
x=4.5;
z = Math.round(x);
document.write(" The round off a number z is: " + z);
</script>
</body>
</html>
```
Question 10

Question:

Write a program to find the incremented and decremented values of two numbers.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var x;
var y;
var z;
var p;
var a;
var b;
x=4;
y=6;
z=x+1;
p=x-1;
a = y+1;
b = y-1;
</script>
</body>
</html>
```
Question 11

Question:

Write a program to find the greatest of two numbers using if – else statement.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var x;
var y;
x=4;
y=6;
if(x>y){
document.write(" x is greater than y");
} else {
document.write(" y is greater than x");
</script>
</body>
</html>
```
Question 12

Question:

Write a program to print the first ten natural numbers using for loop statement.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var i;
for (i=1; i<=10; i++)
document.write("" + i);
</script>
</body>
</html>
```
Question 13

Question:

Write a program to print the first ten natural numbers using while loop statement.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var i=1;
while (i<=10) 
document.write("" + i++);
</script>
</body>
</html>
```

Question 14

Question:

Write a program to print the first nine natural numbers using do while loop statement.
Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var i=1;
do{
document.write("" + i++);
} while (i<10)
</script>
</body>
</html>
```

Question 15

Question:

Write a program to print the average of the first 10 numbers using for loop statement.

Solution:

```html
<!DOCTYPE html>
<html>
```
Question 16

Question:

Write a program to add two numbers using JavaScript function.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
function addition(a, b) {
  return a + b;
}
</script>
</body>
</html>
```
write("" + addition(4, 3));
</script>
</body>
</html>

---

**Question 17**

**Question:**

Write a program to display the current date and time.

**Solution:**

```
<!DOCTYPE html>
<html>
<body>
<button type="button"
onclick="document.getElementById('demo').innerHTML = Date()">
Click me to display Date and Time.</button>
<p id="demo"></p>
</body>
</html>
```
Question 18

Question:

Write a program to find the area of a triangle where lengths of the three of its sides are 5, 6, 7.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var side1 = 5;
var side2 = 6;
var side3 = 7;
var s = (side1 + side2 + side3)/2;
var area = Math.sqrt(s*((s-side1)*(s-side2)*(s-side3)));
document.write(area);
</script>
</body>
</html>
```

418
Question 19

Question:

Write a program to convert temperature from fahrenheit to celsius.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<p id="demo"></p>
<script>
function toCelsius(f) {
    return (5/9) * (f-32);
}
document.getElementById("demo").innerHTML = toCelsius(77);
</script>
</body>
</html>
```

Question 20

Question:

Write a program to get the website URL (loading page).
Question 21

Question:

Write a program to reverse a given string.

Solution:

```javascript
function string_reverse(str) {
    return str.split('').reverse().join('');
}
```
Document.write(string_reverse("JavaScript"));
</script>
</body>
</html>

Question 22

Question:

Write a program to count the number of vowels in a given string.

Solution:

<!DOCTYPE html>
<html>
<body>
<script>
function vowel_Count(str)
{
    return str.replace(/[^aeiou]/g, "").length;
}
document.write(vowel_Count("Python"));
</script>
</body>
</html>
Question 23

Question:

Write a program to find sum of array elements.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
var num = [1,2,3,4]
var sum = 0;
for(var i = 0; i < num.length; i++){
    sum += num[i]
}
document.write(sum);
</script>
</body>
</html>
```

Question 24

Question:

Write a program to create the dot products of two given 3D vectors.
Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
function dot_product(vector1, vector2) {
    var result = 0;
    for (var i = 0; i < 3; i++) {
        result += vector1[i] * vector2[i];
    }
    return result;
}

document.write(dot_product([1,2,3], [1,2,3]))
</script>
</body>
</html>
```

Question 25

Question:

Write a program to find the number of even digits in a given integer.

Solution:

```html
<!DOCTYPE html>
```
function even_digits(num) {
    var ctr = 0;
    while (num) {
        ctr += num % 2 === 0;
        num = Math.floor(num / 10);
    }
    return ctr;
}

document.write(even_digits(124));
</script>
</body>
</html>

---

**Question 26**

**Question:**

Write a program to change the capitalization of all letters in a given string.

**Solution:**

```html
<!DOCTYPE html>
<html>
<body>
<script>
function change_case(txt) {
```
Question 27

Question:

Write a program to remove all characters from a given string that appear more than once.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
function remove_duplicate_chars(str) {
    var arr_char = str.split("\"\");
    var result_arr = [];
    ```
for (var i = 0; i < arr_char.length; i++) {
    if (str.indexOf(arr_char[i]) === str.lastIndexOf(arr_char[i]))
        result_arr.push(arr_char[i]);
}

return result_arr.join('');
}
document.write(remove_duplicate_cchars("abcdabc"));
</script>
</body>
</html>

Question 28

Question:

Write a program to sort an array of all prime numbers between 1 and a given integer.

Solution:

<!DOCTYPE html>
<html>
<body>
<script>
function sort_prime(num) {
var prime_num1 = [],
    prime_num2 = [];
for (var i = 0; i <= num; i++) {
    prime_num2.push(true);
}
for (var i = 2; i <= num; i++) {
    if (prime_num2[i]) {
        prime_num1.push(i);
        for (var j = 1; i * j <= num; j++) {
            prime_num2[i * j] = false;
        }
    }
}

return prime_num1;
}
document.write(sort_prime(11));
</script>
</body>
</html>

---

**Question 29**

**Question:**

*Write a program to check whether there is at least one element which occurs in two given sorted arrays of integers.*
Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
function check_common_element(arr1, arr2) {
  for (var i = 0; i < arr1.length; i++) {
    if (arr2.indexOf(arr1[i]) !== -1)
      return true;
  }
  return false;
}

document.write(check_common_element([1, 2, 3], [3, 4, 5]));
</script>
</body>
</html>
```

Question 30

Question:
Write a program to find the maximum difference between any two adjacent elements of a given array of integers.

Solution:

```html
<!DOCTYPE html>
<html>
<body>
<script>
function max_difference(arr) {
    var max = -1;
    var temp;
    for (var i = 0; i < arr.length - 1; i++) {
        temp = Math.abs(arr[i] - arr[i + 1]);
        max = Math.max(max, temp);
    }
    return max;
}
document.write(max_difference([1, 2, 3, 8, 9]));
</script>
</body>
</html>
```
Advantages and disadvantages of using JavaScript

Advantages:

- Supported by so many browsers.
- Simple to learn (its syntax is close to English) and implement.
- Offer a really high speed i.e., instance response to the visitors: you don’t have to wait for a page reload to get your desire result.
- Versatile: JavaScript can also be used inside scripts written in other languages such as Perl and PHP.

Disadvantages:

- Security issue: can be exploited for malicious purposes.
- Cannot be used for any networking applications.

Timeline of machine learning

<table>
<thead>
<tr>
<th>Year</th>
<th>Event type</th>
<th>Caption</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1763</td>
<td>Discovery</td>
<td>The Underpinnings of Bayes' Theorem</td>
<td>Thomas Bayes's work <em>An Essay towards solving a Problem in the Doctrine of Chances</em> is published two years after his death, having been amended and edited by a friend of Bayes, Richard Price. The essay presents work which underpins Bayes theorem.</td>
</tr>
<tr>
<td>1805</td>
<td>Discovery</td>
<td>Least Square</td>
<td>Adrien-Marie Legendre describes the &quot;méthode des moindres carrés&quot;, known in English as the least squares method. The least squares method is used widely in data fitting.</td>
</tr>
<tr>
<td>1812</td>
<td></td>
<td>Bayes' Theorem</td>
<td>Pierre-Simon Laplace publishes <em>Théorie Analytique des Probabilités</em>, in which he expands upon the work of Bayes and defines what is now known as Bayes' Theorem.</td>
</tr>
<tr>
<td>Year</td>
<td>Discovery/Achievement</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>1913</td>
<td>Discovery</td>
<td>Andrey Markov first describes techniques he used to analyse a poem. The techniques later become known as Markov chains.</td>
<td></td>
</tr>
<tr>
<td>1950</td>
<td>Turing's Learning</td>
<td>Alan Turing proposes a 'learning machine' that could learn and become artificially intelligent. Turing's specific proposal foreshadows genetic algorithms.</td>
<td></td>
</tr>
<tr>
<td>1951</td>
<td>First Neural Network</td>
<td>Marvin Minsky and Dean Edmonds build the first neural network machine, able to learn, the SNARC.</td>
<td></td>
</tr>
<tr>
<td>1952</td>
<td>Machines Playing</td>
<td>Arthur Samuel joins IBM's Poughkeepsie Laboratory and begins working on some of the very first machine learning programs, first creating programs that play checkers.</td>
<td></td>
</tr>
<tr>
<td>1957</td>
<td>Discovery</td>
<td>Frank Rosenblatt invents the perceptron while working at the Cornell Aeronautical Laboratory. The invention of the perceptron generated a great deal of excitement and was widely covered in the media.</td>
<td></td>
</tr>
<tr>
<td>1963</td>
<td>Achievement</td>
<td>Donald Michie creates a 'machine' consisting of 304 match boxes and beads, which uses reinforcement learning to play Tic-tac-toe (also known as noughts and crosses).</td>
<td></td>
</tr>
<tr>
<td>1967</td>
<td>Nearest Neighbor</td>
<td>The nearest neighbor algorithm was created, which is the start of basic pattern recognition. The algorithm was used to map routes.</td>
<td></td>
</tr>
<tr>
<td>1969</td>
<td>Limitations of Neural</td>
<td>Marvin Minsky and Seymour Papert publish their book <em>Perceptrons</em>, describing some of the limitations of perceptrons and neural networks. The interpretation that the book shows that neural networks are fundamentally limited is seen as a hindrance for research into neural networks.</td>
<td></td>
</tr>
<tr>
<td>1970</td>
<td>Automatic Differentiation</td>
<td>Seppo Linna published the general method for automatic differentiation (AD) of discrete connected networks of nested differentiable functions. This corresponds to the modern version of backpropagation, but is not yet named as such.</td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>Category</td>
<td>Event Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>1979</td>
<td>Discovery</td>
<td>Stanford Cart</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Students at Stanford University develop a cart that can navigate and avoid obstacles in a room.</td>
<td></td>
</tr>
<tr>
<td>1979</td>
<td>Discovery</td>
<td>Neocognitron</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Kunihiko Fukushima first publishes his work on the neocognitron, a type of artificial neural network (ANN). Neocognition later inspires convolutional neural networks (CNNs).</td>
<td></td>
</tr>
<tr>
<td>1981</td>
<td>Explanation Based Learning</td>
<td>Gerald Dejong introduces Explanation Based Learning, where a computer algorithm analyses data and creates a general rule it can follow and discard unimportant data.</td>
<td></td>
</tr>
<tr>
<td>1982</td>
<td>Discovery</td>
<td>Recurrent Neural Network</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>John Hopfield popularizes Hopfield networks, a type of recurrent neural network that can serve as content-addressable memory systems.</td>
<td></td>
</tr>
<tr>
<td>1985</td>
<td></td>
<td>NetTalk</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>A program that learns to pronounce words the same way a baby does, is developed by Terry Sejnowski.</td>
<td></td>
</tr>
<tr>
<td>1986</td>
<td>Application</td>
<td>Backpropagation</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Seppo Linnainmaa's reverse mode of automatic differentiation (first applied to neural networks by Paul Werbos) is used in experiments by David Rumelhart, Geoff Hinton and Ronald J. Williams to learn internal representations.</td>
<td></td>
</tr>
<tr>
<td>1989</td>
<td>Discovery</td>
<td>Reinforcement Learning</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Christopher Watkins develops Q-learning, which greatly improves the practicality and feasibility of reinforcement learning.</td>
<td></td>
</tr>
<tr>
<td>1989</td>
<td>Commercialization</td>
<td>Commercialization of Machine Learning on Personal Computers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Axcelis, Inc. releases Evolver, the first software package to commercialize the use of genetic algorithms on personal computers.</td>
<td></td>
</tr>
<tr>
<td>1992</td>
<td>Achievement</td>
<td>Machines Playing Backgammon</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Gerald Tesauro develops TD-Gammon, a computer backgammon program that uses an artificial neural network trained using temporal-difference learning (hence the 'TD' in the name). TD-Gammon is able to rival, but not consistently surpass, the abilities of top human backgammon players.</td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>Category</td>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>----------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>1995</td>
<td>Discovery</td>
<td>Random Forest Algorithm</td>
<td>Tin Kam Ho publishes a paper describing random decision forests.</td>
</tr>
<tr>
<td>1995</td>
<td>Discovery</td>
<td>Support Vector Machines</td>
<td>Corinna Cortes and Vladimir Vapnik publish their work on support vector machines.</td>
</tr>
<tr>
<td>1997</td>
<td>Achievement</td>
<td>IBM Deep Blue Beats Kasparov</td>
<td>IBM’s Deep Blue beats the world champion at chess.</td>
</tr>
<tr>
<td>1998</td>
<td></td>
<td>MNIST database</td>
<td>A team led by Yann LeCun releases the MNIST database, a dataset comprising a mix of handwritten digits from American Census Bureau employees and American high school students. The MNIST database has since become a benchmark for evaluating handwriting recognition.</td>
</tr>
<tr>
<td>2002</td>
<td></td>
<td>Torch Machine Learning Library</td>
<td>Torch, a software library for machine learning, is first released.</td>
</tr>
<tr>
<td>2006</td>
<td></td>
<td>The Netflix Prize</td>
<td>The Netflix Prize competition is launched by Netflix. The aim of the competition was to use machine learning to beat Netflix's own recommendation software's accuracy in predicting a user's rating for a film given their ratings for previous films by at least 10%. The prize was won in 2009.</td>
</tr>
<tr>
<td>2009</td>
<td>Achievement</td>
<td>ImageNet</td>
<td>ImageNet is created. ImageNet is a large visual database envisioned by Fei-Fei Li from Stanford University, who realized that the best machine learning algorithms wouldn't work well if the data didn't reflect the real world. For many, ImageNet was the catalyst for the AI boom of the 21st century.</td>
</tr>
<tr>
<td>2010</td>
<td></td>
<td>Kaggle Competition</td>
<td>Kaggle, a website that serves as a platform for machine learning</td>
</tr>
<tr>
<td>Year</td>
<td>Achievement</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>2010</td>
<td>Wall Street Journal Profiles Machine Learning Investing</td>
<td>The WSJ Profiles new wave of investing and focuses on RebellionResearch.com which would be the subject of author Scott Patterson's Novel, Dark Pools.</td>
<td></td>
</tr>
<tr>
<td>2011</td>
<td>Achievement</td>
<td>Beating Humans in Jeopardy</td>
<td></td>
</tr>
<tr>
<td>2012</td>
<td>Achievement</td>
<td>Recognizing Cats on YouTube</td>
<td></td>
</tr>
<tr>
<td>2014</td>
<td>Achievement</td>
<td>Leaping in Face Recognition</td>
<td></td>
</tr>
<tr>
<td>2014</td>
<td>Sibyl</td>
<td>Researchers from Google detail their work on Sibyl, a proprietary platform for massively parallel machine learning used internally by Google to make predictions about user behavior and provide recommendations.</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>Achievement</td>
<td>Beating Humans in Go</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Google's AlphaGo program becomes the first Computer Go program to beat an unhandicapped professional human player using a combination of machine learning and tree search techniques. Later improved as AlphaGo Zero and then in 2017 generalized to Chess and more two-player games with AlphaZero.</td>
<td></td>
</tr>
</tbody>
</table>

**Logic Statements:**
Basic If Syntax

```java
if (condition) {
    // Execute all statements inside the body if the condition is true
}
```

Else

```java
if (condition) {
    // Execute all statements inside the body if the condition is true
} else {
    // Execute all statements inside the body if the condition is false
}
```

Else if

```java
if (condition1) {
    // Execute all statements inside the body if condition1 is true
} else if (condition2) {
    // Execute all statements inside the body if the condition1 is false and condition2 is true
} else {
    // Execute all statements inside the body if the condition1 is false and condition2 is false
}
```
Break

while (condition) {
    // statements
    if (condition to break) {
        break;
    }
    // statements
}

for (variable initialization; condition; variable update) {
    // statements
    if (condition to break) {
        break;
    }
    // statements
}

do (condition) {
    // statements
    if (condition to break) {
        break;
    }
    // statements
}

while (condition);
Continue

```plaintext
while (condition) {
    // statements
    if (condition to break) {
        continue;
    }
    // statements
}
```

```plaintext
do (condition) {
    // statements
    if (condition to break) {
        continue;
    }
    // statements
}
while (condition);
```

```plaintext
for (variable initialization; condition; variable update) {
    // statements
    if (condition to break) {
        continue;
    }
    // statements
}
```
Switch case

```
switch (expression) {
    case 1:
        // statements
        break;
    case 2:
        // statements
        break;
    default:
        // statements
}
```

Loops

- for Loop

```
for (variable initialization; condition; variable update) {
    // statements
}
```

- while Loop

```
while (condition) {
    // statements
}
```

- do while Loop

```
do {
```
Timeline of artificial intelligence

<table>
<thead>
<tr>
<th>Period</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Antiquity</td>
<td>Greek myths of Hephaestus and Pygmalion incorporated the idea of intelligent robots (such as Talos) and artificial beings (such as Galatea and Pandora). Sacred mechanical statues built in Egypt and Greece were believed to be capable of wisdom and emotion. Hermes Trismegistus would write &quot;they have sensus and spiritus ... by discovering the true nature of the gods, man has been able to reproduce it.&quot; Mosaic law prohibits the use of automatons in religion.</td>
</tr>
<tr>
<td>10th century BC</td>
<td>Yan Shi presented King Mu of Zhou with mechanical men.</td>
</tr>
<tr>
<td>384 BC–322 BC</td>
<td>Aristotle described the syllogism, a method of formal, mechanical thought and theory of knowledge in The Organon.</td>
</tr>
<tr>
<td>1st century</td>
<td>Heron of Alexandria created mechanical men and other automatons.</td>
</tr>
<tr>
<td>260</td>
<td>Porphyry of Tyros wrote Isagoge which categorized knowledge and logic.</td>
</tr>
<tr>
<td>~800</td>
<td>Geber developed the Arabic alchemical theory of Takwin, the artificial creation of life in the laboratory, up to and including human life.</td>
</tr>
<tr>
<td>1206</td>
<td>Al-Jazari created a programmable orchestra of mechanical human beings.</td>
</tr>
<tr>
<td>1275</td>
<td>Ramon Llull, Spanish theologian, invents the Ars Magna, a tool for combining concepts mechanically, based on an Arabic astrological tool, the Zairja. The method would be developed further by Gottfried Leibniz in the 17th century.</td>
</tr>
<tr>
<td>~1500</td>
<td>Paracelsus claimed to have created an artificial man out of magnetism, sperm and alchemy.</td>
</tr>
<tr>
<td>~1580</td>
<td>Rabbi Judah Loew ben Bezalel of Prague is said to have invented the Golem, a clay man brought to life.</td>
</tr>
<tr>
<td>Early 17th century</td>
<td>René Descartes proposed that bodies of animals are nothing more than complex machines (but that mental phenomena are of a different &quot;substance&quot;).</td>
</tr>
<tr>
<td>1620</td>
<td>Sir Francis Bacon developed empirical theory of knowledge and introduced inductive logic in his work The New Organon, a play on Aristotle's title The Organon.</td>
</tr>
<tr>
<td>1623</td>
<td>Wilhelm Schickard drew a calculating clock on a letter to Kepler. This will be the first of five unsuccessful attempts at designing a direct entry calculating clock in the 17th century (including the designs of Tito Burattini, Samuel Morland and René Grillet).</td>
</tr>
<tr>
<td>Year</td>
<td>Event</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>1641</td>
<td>Thomas Hobbes published <em>Leviathan</em> and presented a mechanical, combinatorial theory of cognition. He wrote &quot;...for reason is nothing but reckoning&quot;.</td>
</tr>
<tr>
<td>1642</td>
<td>Blaise Pascal invented the mechanical calculator, the first digital calculating machine.</td>
</tr>
<tr>
<td>1672</td>
<td>Gottfried Leibniz improved the earlier machines, making the Stepped Reckoner to do multiplication and division. He also invented the binary numeral system and envisioned a universal calculus of reasoning (alphabet of human thought) by which arguments could be decided mechanically. Leibniz worked on assigning a specific number to each and every object in the world, as a prelude to an algebraic solution to all possible problems.</td>
</tr>
<tr>
<td>1726</td>
<td>Jonathan Swift published <em>Gulliver's Travels</em>, which includes this description of the Engine, a machine on the island of Laputa: &quot;a Project for improving speculative Knowledge by practical and mechanical Operations &quot; by using this &quot;Contrivance&quot;, &quot;the most ignorant Person at a reasonable Charge, and with a little bodily Labour, may write Books in Philosophy, Poetry, Politicks, Law, Mathematicks, and Theology, with the least Assistance from Genius or study.&quot; The machine is a parody of <em>Ars Magna</em>, one of the inspirations of Gottfried Leibniz' mechanism.</td>
</tr>
<tr>
<td>1750</td>
<td>Julien Offray de La Mettrie published <em>L'Homme Machine</em>, which argued that human thought is strictly mechanical.</td>
</tr>
<tr>
<td>1769</td>
<td>Wolfgang von Kempelen built and toured with his chess-playing automaton, The Turk. The Turk was later shown to be a hoax, involving a human chess player.</td>
</tr>
<tr>
<td>1818</td>
<td>Mary Shelley published the story of <em>Frankenstein; or the Modern Prometheus</em>, a fictional consideration of the ethics of creating sentient beings.</td>
</tr>
<tr>
<td>1822–1859</td>
<td>Charles Babbage and Ada Lovelace worked on programmable mechanical calculating machines.</td>
</tr>
<tr>
<td>1837</td>
<td>The mathematician Bernard Bolzano made the first modern attempt to formalize semantics.</td>
</tr>
<tr>
<td>1854</td>
<td>George Boole set out to &quot;investigate the fundamental laws of those operations of the mind by which reasoning is performed, to give expression to them in the symbolic language of a calculus&quot;, inventing Boolean algebra.</td>
</tr>
<tr>
<td>1863</td>
<td>Samuel Butler suggested that Darwinian evolution also applies to machines, and speculates that they will one day become conscious and eventually supplant humanity.</td>
</tr>
<tr>
<td>1913</td>
<td>Bertrand Russell and Alfred North Whitehead published <em>Principia Mathematica</em>, which revolutionized formal logic.</td>
</tr>
<tr>
<td>1915</td>
<td>Leonardo Torres y Quevedo built a chess automaton, El Ajedrecista, and published speculation about thinking and automata.</td>
</tr>
<tr>
<td>1923</td>
<td>Karel Čapek's play <em>R.U.R. (Rossum's Universal Robots)</em> opened in London. This is the first use of the word &quot;robot&quot; in English.</td>
</tr>
<tr>
<td>1920s and 1930s</td>
<td>Ludwig Wittgenstein and Rudolf Carnap led philosophy into logical analysis of knowledge. Alonzo Church developed Lambda Calculus to investigate computability using recursive functional notation.</td>
</tr>
<tr>
<td>1931</td>
<td>Kurt Gödel showed that sufficiently powerful formal systems, if consistent, permit the formulation of true theorems that are unprovable by any theorem-proving machine deriving all possible theorems.</td>
</tr>
</tbody>
</table>
from the axioms. To do this he had to build a universal, integer-based programming language, which is the reason why he is sometimes called the "father of theoretical computer science".

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1940</td>
<td>Edward Condon displays Nimatron, a digital computer that played Nim perfectly.</td>
</tr>
<tr>
<td>1941</td>
<td>Konrad Zuse built the first working program-controlled computers.</td>
</tr>
<tr>
<td></td>
<td>Arturo Rosenblueth, Norbert Wiener and Julian Bigelow coin the term &quot;cybernetics&quot;. Wiener's popular book by that name published in 1948.</td>
</tr>
<tr>
<td>1945</td>
<td>Game theory which would prove invaluable in the progress of AI was introduced with the 1944 paper, Theory of Games and Economic Behavior by mathematician John von Neumann and economist Oskar Morgenstern.</td>
</tr>
<tr>
<td></td>
<td>Vannevar Bush published <em>As We May Think</em> (The Atlantic Monthly, July 1945) a prescient vision of the future in which computers assist humans in many activities.</td>
</tr>
<tr>
<td>1948</td>
<td>John von Neumann (quoted by E.T. Jaynes) in response to a comment at a lecture that it was impossible for a machine to think: &quot;You insist that there is something a machine cannot do. If you will tell me precisely what it is that a machine cannot do, then I can always make a machine which will do just that!&quot;. Von Neumann was presumably alluding to the Church-Turing thesis which states that any effective procedure can be simulated by a (generalized) computer.</td>
</tr>
<tr>
<td>1950</td>
<td>Alan Turing proposes the Turing Test as a measure of machine intelligence.</td>
</tr>
<tr>
<td></td>
<td>Claude Shannon published a detailed analysis of chess playing as search.</td>
</tr>
<tr>
<td></td>
<td>Isaac Asimov published his Three Laws of Robotics.</td>
</tr>
<tr>
<td>1951</td>
<td>The first working AI programs were written in 1951 to run on the Ferranti Mark 1 machine of the University of Manchester: a checkers-playing program written by Christopher Strachey and a chess-playing program written by Dietrich Prinz.</td>
</tr>
<tr>
<td>1952–1962</td>
<td>Arthur Samuel (IBM) wrote the first game-playing program, for checkers (draughts), to achieve sufficient skill to challenge a respectable amateur. His first checkers-playing program was written in 1952, and in 1955 he created a version that learned to play.</td>
</tr>
<tr>
<td>1956</td>
<td>The Dartmouth College summer AI conference is organized by John McCarthy, Marvin Minsky, Nathan Rochester of IBM and Claude Shannon. McCarthy coins the term <em>artificial intelligence</em> for the conference.</td>
</tr>
<tr>
<td></td>
<td>The first demonstration of the Logic Theorist (LT) written by Allen Newell, J.C. Shaw and Herbert A. Simon (Carnegie Institute of Technology, now Carnegie Mellon University or CMU). This is often called the first AI program, though Samuel's checkers program also has a strong claim.</td>
</tr>
<tr>
<td>1958</td>
<td>John McCarthy (Massachusetts Institute of Technology or MIT) invented the Lisp programming language.</td>
</tr>
<tr>
<td></td>
<td>Herbert Gelernter and Nathan Rochester (IBM) described a theorem prover in geometry that exploits a semantic model of the domain in the form of diagrams of &quot;typical&quot; cases.</td>
</tr>
</tbody>
</table>
Teddington Conference on the Mechanization of Thought Processes was held in the UK and among the papers presented were John McCarthy's *Programs with Common Sense*, Oliver Selfridge's *Pandemonium*, and Marvin Minsky's *Some Methods of Heuristic Programming and Artificial Intelligence*.

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1959</td>
<td>The General Problem Solver (GPS) was created by Newell, Shaw and Simon while at CMU. John McCarthy and Marvin Minsky founded the MIT AI Lab.</td>
</tr>
<tr>
<td>Late 1950s, early 1960s</td>
<td>Margaret Masterman and colleagues at University of Cambridge design semantic nets for machine translation.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1960</td>
<td><em>Man-Computer Symbiosis</em> by J.C.R. Licklider.</td>
</tr>
<tr>
<td>1961</td>
<td>James Slagle (PhD dissertation, MIT) wrote (in Lisp) the first symbolic integration program, SAINT, which solved calculus problems at the college freshman level. In <em>Minds, Machines and Gödel</em>, John Lucas denied the possibility of machine intelligence on logical or philosophical grounds. He referred to Kurt Gödel's result of 1931: sufficiently powerful formal systems are either inconsistent or allow for formulating true theorems unprovable by any theorem-proving AI deriving all provable theorems from the axioms. Since humans are able to &quot;see&quot; the truth of such theorems, machines were deemed inferior. Unimation's industrial robot Unimate worked on a General Motors automobile assembly line.</td>
</tr>
<tr>
<td>1963</td>
<td>Thomas Evans' program, ANALOGY, written as part of his PhD work at MIT, demonstrated that computers can solve the same analogy problems as are given on IQ tests. Edward Feigenbaum and Julian Feldman published <em>Computers and Thought</em>, the first collection of articles about artificial intelligence. Leonard Uhr and Charles Vossler published &quot;A Pattern Recognition Program That Generates, Evaluates, and Adjusts Its Own Operators&quot;, which described one of the first machine learning programs that could adaptively acquire and modify features and thereby overcome the limitations of simple perceptrons of Rosenblatt.</td>
</tr>
<tr>
<td>1964</td>
<td>Danny Bobrow's dissertation at MIT (technical report #1 from MIT's AI group, Project MAC), shows that computers can understand natural language well enough to solve algebra word problems correctly. Bertram Raphael's MIT dissertation on the SIR program demonstrates the power of a logical representation of knowledge for question-answering systems.</td>
</tr>
<tr>
<td>1965</td>
<td>Lotfi Zadeh at U.C. Berkeley publishes his first paper introducing fuzzy logic &quot;Fuzzy Sets&quot; (Information and Control 8: 338–353). J. Alan Robinson invented a mechanical proof procedure, the Resolution Method, which allowed programs...</td>
</tr>
</tbody>
</table>
to work efficiently with formal logic as a representation language.

Joseph Weizenbaum (MIT) built ELIZA, an interactive program that carries on a dialogue in English language on any topic. It was a popular toy at AI centers on the ARPANET when a version that "simulated" the dialogue of a psychotherapist was programmed.

Edward Feigenbaum initiated Dendral, a ten-year effort to develop software to deduce the molecular structure of organic compounds using scientific instrument data. It was the first expert system.

1966
Ross Quillian (PhD dissertation, Carnegie Inst. of Technology, now CMU) demonstrated semantic nets.

Machine Intelligence workshop at Edinburgh – the first of an influential annual series organized by Donald Michie and others.

Negative report on machine translation kills much work in Natural language processing (NLP) for many years.

Dendral program (Edward Feigenbaum, Joshua Lederberg, Bruce Buchanan, Georgia Sutherland at Stanford University) demonstrated to interpret mass spectra on organic chemical compounds. First successful knowledge-based program for scientific reasoning.

1968
Joel Moses (PhD work at MIT) demonstrated the power of symbolic reasoning for integration problems in the Macsyma program. First successful knowledge-based program in mathematics.

Richard Greenblatt (programmer) at MIT built a knowledge-based chess-playing program, MacHack, that was good enough to achieve a class-C rating in tournament play.

Wallace and Boulton's program, Snob (Comp.J. 11(2) 1968), for unsupervised classification (clustering) uses the Bayesian Minimum Message Length criterion, a mathematical realisation of Occam's razor.

1969

Roger Schank (Stanford) defined conceptual dependency model for natural language understanding. Later developed (in PhD dissertations at Yale University) for use in story understanding by Robert Wilensky and Wendy Lehnert, and for use in understanding memory by Janet Kolodner.

Yorick Wilks (Stanford) developed the semantic coherence view of language called Preference Semantics, embodied in the first semantics-driven machine translation program, and the basis of many PhD dissertations since such as Bran Boguraev and David Carter at Cambridge.

First International Joint Conference on Artificial Intelligence (IJCAI) held at Stanford.

Marvin Minsky and Seymour Papert publish *Perceptrons*, demonstrating previously unrecognized limits of this feed-forward two-layered structure, and This book is considered by some to mark the beginning of the AI winter of the 1970s, a failure of confidence and funding for AI. Nevertheless, significant progress in the field continued (see below).

McCarthy and Hayes started the discussion about the frame problem with their essay, "Some Philosophical Problems from the Standpoint of Artificial Intelligence".
<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
</table>
| 1970 | Seppo Linnainmaa publishes the reverse mode of automatic differentiation. This method became later known as backpropagation, and is heavily used to train artificial neural networks.  
Jaime Carbonell (Sr.) developed SCHOLAR, an interactive program for computer assisted instruction based on semantic nets as the representation of knowledge.  
Bill Woods described Augmented Transition Networks (ATN's) as a representation for natural language understanding.  
Patrick Winston’s PhD program, ARCH, at MIT learned concepts from examples in the world of children’s blocks. |
| 1971 | Terry Winograd’s PhD thesis (MIT) demonstrated the ability of computers to understand English sentences in a restricted world of children’s blocks, in a coupling of his language understanding program, SHRDLU, with a robot arm that carried out instructions typed in English.  
Work on the Boyer-Moore theorem prover started in Edinburgh. |
| 1972 | Prolog programming language developed by Alain Colmerauer.  
Earl Sacerdoti developed one of the first hierarchical planning programs, ABSTRIPS. |
| 1973 | The Assembly Robotics Group at University of Edinburgh builds Freddy Robot, capable of using visual perception to locate and assemble models. (Edinburgh Freddy Assembly Robot: a versatile computer-controlled assembly system.)  
The Lighthill report gives a largely negative verdict on AI research in Great Britain and forms the basis for the decision by the British government to discontinue support for AI research in all but two universities. |
| 1974 | Ted Shortliffe's PhD dissertation on the MYCIN program (Stanford) demonstrated a very practical rule-based approach to medical diagnoses, even in the presence of uncertainty. While it borrowed from DENDRAL, its own contributions strongly influenced the future of expert system development, especially commercial systems. |
| 1975 | Earl Sacerdoti developed techniques of partial-order planning in his NOAH system, replacing the previous paradigm of search among state space descriptions. NOAH was applied at SRI International to interactively diagnose and repair electromechanical systems.  
Austin Tate developed the Nonlin hierarchical planning system able to search a space of partial plans characterised as alternative approaches to the underlying goal structure of the plan.  
Marvin Minsky published his widely read and influential article on Frames as a representation of knowledge, in which many ideas about schemas and semantic links are brought together.  
The Meta-Dendral learning program produced new results in chemistry (some rules of mass spectrometry) the first scientific discoveries by a computer to be published in a refereed journal. |
<p>| Mid-1970s | Barbara Grosz (SRI) established limits to traditional AI approaches to discourse modeling. Subsequent work by Grosz, Bonnie Webber and Candace Sidner developed the notion of “centering”, used in establishing focus of discourse and anaphoric references in Natural language processing. |</p>
<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1976</td>
<td>Douglas Lenat's AM program (Stanford PhD dissertation) demonstrated the discovery model (loosely guided search for interesting conjectures). Randall Davis demonstrated the power of meta-level reasoning in his PhD dissertation at Stanford.</td>
</tr>
<tr>
<td>1978</td>
<td>Tom Mitchell, at Stanford, invented the concept of Version spaces for describing the search space of a concept formation program. Herbert A. Simon wins the Nobel Prize in Economics for his theory of bounded rationality, one of the cornerstones of AI known as &quot;satisficing&quot;. The MOLGEN program, written at Stanford by Mark Stefik and Peter Friedland, demonstrated that an object-oriented programming representation of knowledge can be used to plan gene-cloning experiments.</td>
</tr>
<tr>
<td>1979</td>
<td>Bill VanMelle's PhD dissertation at Stanford demonstrated the generality of MYCIN's representation of knowledge and style of reasoning in his EMYCIN program, the model for many commercial expert system &quot;shells&quot;. Jack Myers and Harry Pople at University of Pittsburgh developed INTERNIST, a knowledge-based medical diagnosis program based on Dr. Myers' clinical knowledge. Cordell Green, David Barstow, Elaine Kant and others at Stanford demonstrated the CHI system for automatic programming. The Stanford Cart, built by Hans Moravec, becomes the first computer-controlled, autonomous vehicle when it successfully traverses a chair-filled room and circumnavigates the Stanford AI Lab. BKG, a backgammon program written by Hans Berliner at CMU, defeats the reigning world champion (in part via luck). Drew McDermott and Jon Doyle at MIT, and John McCarthy at Stanford begin publishing work on non-monotonic logics and formal aspects of truth maintenance.</td>
</tr>
<tr>
<td>Late 1970s</td>
<td>Stanford's SUMEX-AIM resource, headed by Ed Feigenbaum and Joshua Lederberg, demonstrates the power of the ARPAnet for scientific collaboration.</td>
</tr>
<tr>
<td>1980</td>
<td>Lisp machines developed and marketed. First expert system shells and commercial applications. First National Conference of the American Association for Artificial Intelligence (AAAI) held at Stanford.</td>
</tr>
<tr>
<td>1981</td>
<td>Danny Hillis designs the connection machine, which utilizes Parallel computing to bring new power to AI, and to computation in general. (Later founds Thinking Machines Corporation)</td>
</tr>
<tr>
<td>1982</td>
<td>The Fifth Generation Computer Systems project (FGCS), an initiative by Japan's Ministry of International Trade and Industry, begun in 1982, to create a &quot;fifth generation computer&quot; (see history of computing hardware) which was supposed to perform much calculation utilizing massive parallelism.</td>
</tr>
<tr>
<td>1983</td>
<td>John Laird and Paul Rosenbloom, working with Allen Newell, complete CMU dissertations on Soar (program). James F. Allen invents the Interval Calculus, the first widely used formalization of temporal events.</td>
</tr>
<tr>
<td>Year</td>
<td>Event</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Mid-1980s</strong></td>
<td>Neural Networks become widely used with the Backpropagation algorithm, also known as the reverse mode of automatic differentiation published by Seppo Linnainmaa in 1970 and applied to neural networks by Paul Werbos.</td>
</tr>
<tr>
<td>1985</td>
<td>The autonomous drawing program, AARON, created by Harold Cohen, is demonstrated at the AAAI National Conference (based on more than a decade of work, and with subsequent work showing major developments).</td>
</tr>
<tr>
<td>1986</td>
<td>The team of Ernst Dickmanns at Bundeswehr University of Munich builds the first robot cars, driving up to 55 mph on empty streets. Barbara Grosz and Candace Sidner create the first computation model of discourse, establishing the field of research.</td>
</tr>
<tr>
<td>1987</td>
<td>Marvin Minsky published <em>The Society of Mind</em>, a theoretical description of the mind as a collection of cooperating agents. He had been lecturing on the idea for years before the book came out (c.f. Doyle 1983). Around the same time, Rodney Brooks introduced the subsumption architecture and behavior-based robotics as a more minimalist modular model of natural intelligence; Nouvelle AI.</td>
</tr>
<tr>
<td>1989</td>
<td>The development of metal–oxide–semiconductor (MOS) very-large-scale integration (VLSI), in the form of complementary MOS (CMOS) technology, enabled the development of practical artificial neural network (ANN) technology in the 1980s. A landmark publication in the field was the 1989 book <em>Analog VLSI Implementation of Neural Systems</em> by Carver A. Mead and Mohammed Ismail. Dean Pomerleau at CMU creates ALVINN (An Autonomous Land Vehicle in a Neural Network).</td>
</tr>
<tr>
<td>1990s</td>
<td>Major advances in all areas of AI, with significant demonstrations in machine learning, intelligent tutoring, case-based reasoning, multi-agent planning, scheduling, uncertain reasoning, data mining, natural language understanding and translation, vision, virtual reality, games, and other topics.</td>
</tr>
<tr>
<td>Early 1990s</td>
<td>TD-Gammon, a backgammon program written by Gerry Tesauro, demonstrates that reinforcement (learning) is powerful enough to create a championship-level game-playing program by competing favorably with world-class players.</td>
</tr>
<tr>
<td>1991</td>
<td>DART scheduling application deployed in the first Gulf War paid back DARPA’s investment of 30 years in AI research.</td>
</tr>
<tr>
<td>Year</td>
<td>Event</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>1992</td>
<td>Carol Stoker and NASA Ames robotics team explore marine life in Antarctica with an undersea robot Telepresence ROV operated from the ice near McMurdo Bay, Antarctica and remotely via satellite link from Moffett Field, California.</td>
</tr>
<tr>
<td>1993</td>
<td>Ian Horswill extended behavior-based robotics by creating Polly, the first robot to navigate using vision and operate at animal-like speeds (1 meter/second). Rodney Brooks, Lynn Andrea Stein and Cynthia Breazeal started the widely publicized MIT Cog project with numerous collaborators, in an attempt to build a humanoid robot child in just five years. ISX corporation wins &quot;DARPA contractor of the year&quot; for the Dynamic Analysis and Replanning Tool (DART) which reportedly repaid the US government's entire investment in AI research since the 1950s.</td>
</tr>
<tr>
<td>1994</td>
<td>Lotfi Zadeh at U.C. Berkeley creates &quot;soft computing&quot; and builds a world network of research with a fusion of neural science and neural net systems, fuzzy set theory and fuzzy systems, evolutionary algorithms, genetic programming, and chaos theory and chaotic systems (&quot;Fuzzy Logic, Neural Networks, and Soft Computing,&quot; Communications of the ACM, March 1994, Vol. 37 No. 3, pages 77-84). With passengers on board, the twin robot cars VaMP and VITA-2 of Ernst Dickmanns and Daimler-Benz drive more than one thousand kilometers on a Paris three-lane highway in standard heavy traffic at speeds up to 130 km/h. They demonstrate autonomous driving in free lanes, convoy driving, and lane changes left and right with autonomous passing of other cars. English draughts (checkers) world champion Tinsley resigned a match against computer program Chinook. Chinook defeated 2nd highest rated player, Lafferty. Chinook won the USA National Tournament by the widest margin ever. Cindy Mason at NASA organizes the First AAAI Workshop on AI and the Environment.</td>
</tr>
<tr>
<td>1995</td>
<td>Cindy Mason at NASA organizes the First International IJCAI Workshop on AI and the Environment. &quot;No Hands Across America&quot;: A semi-autonomous car drove coast-to-coast across the United States with computer-controlled steering for 2,797 miles (4,501 km) of the 2,849 miles (4,585 km). Throttle and brakes were controlled by a human driver. One of Ernst Dickmanns' robot cars (with robot-controlled throttle and brakes) drove more than 1000 miles from Munich to Copenhagen and back, in traffic, at up to 120 mph, occasionally executing maneuvers to pass other cars (only in a few critical situations a safety driver took over). Active vision was used to deal with rapidly changing street scenes.</td>
</tr>
<tr>
<td>1997</td>
<td>The Deep Blue chess machine (IBM) defeats the (then) world chess champion, Garry Kasparov. First official RoboCup football (soccer) match featuring table-top matches with 40 teams of interacting robots and over 5000 spectators. Computer Othello program Logistello defeated the world champion Takeshi Murakami with a score of 6–0.</td>
</tr>
<tr>
<td>1998</td>
<td>Tiger Electronics' Furby is released, and becomes the first successful attempt at producing a type of A.I to reach a domestic environment.</td>
</tr>
<tr>
<td>Year</td>
<td>Event</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>1999</td>
<td>Sony introduces an improved domestic robot similar to a Furby, the AIBO becomes one of the first artificially intelligent &quot;pets&quot; that is also autonomous.</td>
</tr>
<tr>
<td>Late 1990s</td>
<td>Web crawlers and other AI-based information extraction programs become essential in widespread use of the World Wide Web.</td>
</tr>
<tr>
<td></td>
<td>Demonstration of an Intelligent room and Emotional Agents at MIT's AI Lab.</td>
</tr>
<tr>
<td></td>
<td>Initiation of work on the Oxygen architecture, which connects mobile and stationary computers in an adaptive network.</td>
</tr>
<tr>
<td>2000</td>
<td>Interactive robopets (&quot;smart toys&quot;) become commercially available, realizing the vision of the 18th century novelty toy makers.</td>
</tr>
<tr>
<td></td>
<td>Cynthia Breazeal at MIT publishes her dissertation on Sociable machines, describing Kismet (robot), with a face that expresses emotions.</td>
</tr>
<tr>
<td></td>
<td>The Nomad robot explores remote regions of Antarctica looking for meteorite samples.</td>
</tr>
<tr>
<td>2002</td>
<td>iRobot's Roomba autonomously vacuums the floor while navigating and avoiding obstacles.</td>
</tr>
<tr>
<td></td>
<td>DARPA introduces the DARPA Grand Challenge requiring competitors to produce autonomous vehicles for prize money.</td>
</tr>
<tr>
<td></td>
<td>NASA's robotic exploration rovers Spirit and Opportunity autonomously navigate the surface of Mars.</td>
</tr>
<tr>
<td>2005</td>
<td>Honda's ASIMO robot, an artificially intelligent humanoid robot, is able to walk as fast as a human, delivering trays to customers in restaurant settings.</td>
</tr>
<tr>
<td></td>
<td>Recommendation technology based on tracking web activity or media usage brings AI to marketing. See TiVo Suggestions.</td>
</tr>
<tr>
<td></td>
<td>Blue Brain is born, a project to simulate the brain at molecular detail.</td>
</tr>
<tr>
<td>2006</td>
<td>The Dartmouth Artificial Intelligence Conference: The Next 50 Years (AI@50) AI@50 (14–16 July 2006)</td>
</tr>
<tr>
<td>2007</td>
<td>Checkers is solved by a team of researchers at the University of Alberta.</td>
</tr>
<tr>
<td></td>
<td>DARPA launches the Urban Challenge for autonomous cars to obey traffic rules and operate in an urban...</td>
</tr>
<tr>
<td>Year</td>
<td>Event</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td>2008</td>
<td>Cynthia Mason at Stanford presents her idea on Artificial Compassionate Intelligence, in her paper on &quot;Giving Robots Compassion&quot;.</td>
</tr>
<tr>
<td>2009</td>
<td>Google builds autonomous car.</td>
</tr>
<tr>
<td>2010</td>
<td>Microsoft launched Kinect for Xbox 360, the first gaming device to track human body movement, using just a 3D camera and infra-red detection, enabling users to play their Xbox 360 wirelessly. The award-winning machine learning for human motion capture technology for this device was developed by the Computer Vision group at Microsoft Research, Cambridge.</td>
</tr>
<tr>
<td>2011</td>
<td>Mary Lou Maher and Doug Fisher organize the First AAAI Workshop on AI and Sustainability.</td>
</tr>
<tr>
<td>2013</td>
<td>Robot HRP-2 built by SCHAFT Inc of Japan, a subsidiary of Google, defeats 15 teams to win DARPA’s Robotics Challenge Trials. HRP-2 scored 27 out of 32 points in 8 tasks needed in disaster response. Tasks are drive a vehicle, walk over debris, climb a ladder, remove debris, walk through doors, cut through a wall, close valves and connect a hose. NEIL, the Never Ending Image Learner, is released at Carnegie Mellon University to constantly compare and analyze relationships between different images.</td>
</tr>
<tr>
<td>2015</td>
<td>An open letter to ban development and use of autonomous weapons signed by Hawking, Musk, Wozniak and 3,000 researchers in AI and robotics. Google DeepMind's AlphaGo (version: Fan) defeated 3 time European Go champion 2 dan professional Fan Hui by 5 games to 0.</td>
</tr>
<tr>
<td>2016</td>
<td>Google DeepMind's AlphaGo (version: Lee) defeated Lee Sedol 4–1. Lee Sedol is a 9 dan professional Korean Go champion who won 27 major tournaments from 2002 to 2016. Before the match with AlphaGo, Lee Sedol was confident in predicting an easy 5–0 or 4–1 victory.</td>
</tr>
<tr>
<td>2017</td>
<td>Asilomar Conference on Beneficial AI was held, to discuss AI ethics and how to bring about beneficial AI while avoiding the existential risk from artificial general intelligence. Deepstack is the first published algorithm to beat human players in imperfect information games, as shown with statistical significance on heads-up no-limit poker. Soon after, the poker AI Libratus by different research group individually defeated each of its 4 human opponents—among the best players in the world—at an exceptionally high aggregated winrate, over a statistically significant sample. In contrast to Chess and Go, Poker is an imperfect information game. Google DeepMind's AlphaGo (version: Master) won 60–0 rounds on two public Go websites including 3 wins against world Go champion Ke Jie. A propositional logic boolean satisfiability problem (SAT) solver proves a long-standing mathematical conjecture on Pythagorean triples over the set of integers. The initial proof, 200TB long, was checked by...</td>
</tr>
</tbody>
</table>
two independent certified automatic proof checkers.

An OpenAI-machined learned bot played at The International 2017 Dota 2 tournament in August 2017. It won during a 1v1 demonstration game against professional Dota 2 player Dendi.

Google DeepMind revealed that AlphaGo Zero—an improved version of AlphaGo—displayed significant performance gains while using far fewer tensor processing units (as compared to AlphaGo Lee; it used same amount of TPU's as AlphaGo Master). Unlike previous versions, which learned the game by observing millions of human moves, AlphaGo Zero learned by playing only against itself. The system then defeated AlphaGo Lee 100 games to zero, and defeated AlphaGo Master 89 to 11. Although unsupervised learning is a step forward, much has yet to be learned about general intelligence. AlphaZero masters chess in 4 hours, defeating the best chess engine, StockFish 8. AlphaZero won 28 out of 100 games, and the remaining 72 games ended in a draw.

2018

- Alibaba language processing AI outscores top humans at a Stanford University reading and comprehension test, scoring 82.44 against 82.304 on a set of 100,000 questions.

- The European Lab for Learning and Intelligent Systems (aka Ellis) proposed as a pan-European competitor to American AI efforts, with the aim of staving off a brain drain of talent, along the lines of CERN after World War II.

- Announcement of Google Duplex, a service to allow an AI assistant to book appointments over the phone. The LA Times judges the AI's voice to be a "nearly flawless" imitation of human-sounding speech.

"The computer programmer is a creator of universes for which he alone is the lawgiver. No playwright, no stage director, no emperor, however powerful, has ever exercised such absolute authority to arrange a stage or field of battle and to command such unswervingly dutiful actors or troops."

— Joseph Weizenbaum
Write a python program to get all links from a webpage

```python
from bs4 import BeautifulSoup
from urllib.request import Request, urlopen

req = Request("http://www.scribd.com/")
html_page = urlopen(req)

soup = BeautifulSoup(html_page, "lxml")

links = []
for link in soup.findAll('a'):
    links.append(link.get('href'))

print(links)
```

Write a python program to Print all Integers that Aren’t Divisible by Either 2 or 3

```python
# input the maximum number to which you want to send
max_num = 20

# starting numbers from 0
n = 1

# run until it reaches maximum number
print("Numbers not divisible by 2 and 3")
while n <= max_num:
    # check if number is divisible by 2 and 3
    if n % 2 != 0 and n % 3 != 0:
        print(n)
    # incrementing the counter
    n = n+1
```
Write a Python program which iterates the integers from 1 to 50. For multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz".

```python
for fizzbuzz in range(51):
    if fizzbuzz % 3 == 0 and fizzbuzz % 5 == 0:
        print("fizzbuzz")
        continue
    elif fizzbuzz % 3 == 0:
        print("fizz")
        continue
    elif fizzbuzz % 5 == 0:
        print("buzz")
        continue
    print(fizzbuzz)
```

Write a Python program to convert seconds into hours, minutes and seconds

```python
# Python Program to Convert seconds into hours, minutes and seconds

def convert(seconds):
    seconds = seconds % (24 * 3600)
    hour = seconds // 3600
    seconds %= 3600
    minutes = seconds // 60
    seconds %= 60

    return "%d:%02d:%02d" % (hour, minutes, seconds)

# Driver program
n = 12345
print(convert(n))
```
<table>
<thead>
<tr>
<th>Concept</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstraction</td>
<td>hides the implementation details and only provides the functionality to the user</td>
</tr>
<tr>
<td>Encapsulation</td>
<td>wraps up the data and code together as a single unit</td>
</tr>
<tr>
<td>Inheritance</td>
<td>where one class acquires the properties of another</td>
</tr>
<tr>
<td>Polymorphism</td>
<td>the ability of a variable, function or object to take multiple forms</td>
</tr>
</tbody>
</table>

2 types of polymorphism:
- Compile time polymorphism
- Run time polymorphism.

2 types of Constructor:
- Default Constructor → can be called with no arguments.
- Parameterized Constructor → can be called with specific number of arguments.

BigInteger Implementation in Java:

Code:
```
import java.math.BigInteger;

public class BigInteger {

    public static void main(String[] args) {
```
BigInteger Distance = new BigInteger("2401000000000000000000");

BigInteger SpeedofLight = new BigInteger("1080000000");

BigInteger Time = Distance.divide(SpeedofLight);

System.out.println("Time taken by light to reach the Andromeda Galaxy is " + Time + " hours");

}
sort(array, array+n);
cout << "Array after sorting using "
"default sort is : ";
for (int i = 0; i < n; ++i)
cout << array[i] << " ";
return 0;
}

**Output on the screen:**

Array after sorting using default sort is: 2 10 10 35 43 62 77 85 93 345

print("{} is an interpreted, high-level and general-purpose programming language\".format("Python"))

Output

Python is an interpreted, high-level and general-purpose programming language

Problem Occurs → Create Exception → Throw Exception → Handle Exception

**Differences between C, C++ and Java:**
<table>
<thead>
<tr>
<th></th>
<th>C</th>
<th>C++</th>
<th>Java</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Programming Paradigm</strong></td>
<td>Procedural language</td>
<td>Object-Oriented Programming (OOP)</td>
<td>Pure Object Oriented Oriented</td>
</tr>
<tr>
<td><strong>Origin</strong></td>
<td>Based on assembly language</td>
<td>Based on C language</td>
<td>Based on C and C++</td>
</tr>
<tr>
<td><strong>Developer</strong></td>
<td>Dennis Ritchie in 1972</td>
<td>Bjarne Stroustrup in 1979</td>
<td>James Gosling in 1991</td>
</tr>
<tr>
<td><strong>Translator</strong></td>
<td>Compiler only</td>
<td>Compiler only</td>
<td>Interpreted language (Compiler + interpreter)</td>
</tr>
<tr>
<td><strong>Platform Dependency</strong></td>
<td>Platform Dependent</td>
<td>Platform Dependent</td>
<td>Platform Independent</td>
</tr>
<tr>
<td><strong>Code execution</strong></td>
<td>Direct</td>
<td>Direct</td>
<td>Executed by JVM (Java Virtual Machine)</td>
</tr>
<tr>
<td><strong>Approach</strong></td>
<td>Top-down approach</td>
<td>Bottom-up approach</td>
<td>Bottom-up approach</td>
</tr>
<tr>
<td><strong>File generation</strong></td>
<td>.exe files</td>
<td>.exe files</td>
<td>.class files</td>
</tr>
<tr>
<td><strong>Pre-processor directives</strong></td>
<td>Support header files (#include, #define)</td>
<td>Supported (#header, #define)</td>
<td>Use Packages (import)</td>
</tr>
<tr>
<td><strong>keywords</strong></td>
<td>Support 32 keywords</td>
<td>Supports 63 keywords</td>
<td>50 defined keywords</td>
</tr>
<tr>
<td><strong>Datatypes (union, structure)</strong></td>
<td>Supported</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Inheritance</strong></td>
<td>No inheritance</td>
<td>Supported</td>
<td>Supported except Multiple inheritance</td>
</tr>
<tr>
<td><strong>Overloading</strong></td>
<td>No overloading</td>
<td>Support Function overloading (Polymorphism)</td>
<td>Operator overloading is not supported</td>
</tr>
<tr>
<td><strong>Pointers</strong></td>
<td>Supported</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Allocation</strong></td>
<td>Use malloc, calloc</td>
<td>Use new, delete</td>
<td>Garbage collector</td>
</tr>
<tr>
<td><strong>Exception Handling</strong></td>
<td>Not supported</td>
<td>Supported</td>
<td>Supported</td>
</tr>
<tr>
<td><strong>Templates</strong></td>
<td>Not supported</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Destructors</strong></td>
<td>No constructor neither destructor</td>
<td>Supported</td>
<td>Not supported</td>
</tr>
<tr>
<td><strong>Multithreading/ Interfaces</strong></td>
<td>Not supported</td>
<td>Not supported</td>
<td>Supported</td>
</tr>
<tr>
<td><strong>Database connectivity</strong></td>
<td>Not supported</td>
<td>Not supported</td>
<td>Supported</td>
</tr>
<tr>
<td><strong>Storage Classes</strong></td>
<td>Supported ( auto, extern )</td>
<td>Supported ( auto, extern )</td>
<td>Not supported</td>
</tr>
</tbody>
</table>
• **Mutable objects** → The objects in which you can change the fields and states after the object is created.

• **Immutable objects** → The objects in which you cannot change the fields and states after the object is created.

Threads created by Java Virtual Machine → **Daemon threads**

Threads created by an application itself → **User threads**

---

**Steps to create Java Database Connectivity Application:**

- Import the packages
- Register the Java Database Connectivity driver
- Open a connection
- Execute a query
- Extract data from the result set
- Clean up the environment

---

Training data → Learn Algorithm → Build Model → Perform → Feedback

---

**Machine Learning**
### Keywords in C Programming

<table>
<thead>
<tr>
<th>auto</th>
<th>break</th>
<th>case</th>
<th>char</th>
</tr>
</thead>
<tbody>
<tr>
<td>const</td>
<td>continue</td>
<td>default</td>
<td>do</td>
</tr>
<tr>
<td>double</td>
<td>else</td>
<td>enum</td>
<td>extern</td>
</tr>
<tr>
<td>float</td>
<td>for</td>
<td>goto</td>
<td>if</td>
</tr>
<tr>
<td>int</td>
<td>long</td>
<td>register</td>
<td>return</td>
</tr>
<tr>
<td>short</td>
<td>signed</td>
<td>sizeof</td>
<td>static</td>
</tr>
<tr>
<td>struct</td>
<td>switch</td>
<td>typedef</td>
<td>union</td>
</tr>
<tr>
<td>unsigned</td>
<td>void</td>
<td>volatile</td>
<td>while</td>
</tr>
</tbody>
</table>

### Keywords in C++ Programming

<table>
<thead>
<tr>
<th>alignas</th>
<th>decltype</th>
<th>namespace</th>
<th>struct</th>
</tr>
</thead>
<tbody>
<tr>
<td>alignof</td>
<td>default</td>
<td>new</td>
<td>switch</td>
</tr>
<tr>
<td>and</td>
<td>delete</td>
<td>noexcept</td>
<td>template</td>
</tr>
<tr>
<td>and_eq</td>
<td>do</td>
<td>not</td>
<td>this</td>
</tr>
<tr>
<td>asm</td>
<td>double</td>
<td>not_eq</td>
<td>thread_local</td>
</tr>
<tr>
<td>auto</td>
<td>dynamic_cast</td>
<td>nullptr</td>
<td>throw</td>
</tr>
<tr>
<td>bitand</td>
<td>else</td>
<td>operator</td>
<td>true</td>
</tr>
<tr>
<td>bitor</td>
<td>enum</td>
<td>or</td>
<td>try</td>
</tr>
<tr>
<td>bool</td>
<td>explicit</td>
<td>or_eq</td>
<td>typedef</td>
</tr>
<tr>
<td>break</td>
<td>export</td>
<td>private</td>
<td>typeid</td>
</tr>
<tr>
<td>case</td>
<td>extern</td>
<td>protected</td>
<td>typename</td>
</tr>
<tr>
<td>catch</td>
<td>false</td>
<td>public</td>
<td>union</td>
</tr>
<tr>
<td>char</td>
<td>float</td>
<td>register</td>
<td>unsigned</td>
</tr>
<tr>
<td>char16_t</td>
<td>for</td>
<td>reinterpret_cast</td>
<td>using</td>
</tr>
<tr>
<td>char32_t</td>
<td>friend</td>
<td>return</td>
<td>virtual</td>
</tr>
<tr>
<td>class</td>
<td>goto</td>
<td>short</td>
<td>void</td>
</tr>
<tr>
<td>compl</td>
<td>if</td>
<td>signed</td>
<td>volatile</td>
</tr>
<tr>
<td>const</td>
<td>inline</td>
<td>sizeof</td>
<td>wchar_t</td>
</tr>
<tr>
<td>constexpr</td>
<td>int</td>
<td>static</td>
<td>while</td>
</tr>
<tr>
<td>const_cast</td>
<td>long</td>
<td>static_assert</td>
<td>xor</td>
</tr>
<tr>
<td>continue</td>
<td>mutable</td>
<td>static_cast</td>
<td>xor_eq</td>
</tr>
</tbody>
</table>
Keywords in Java Programming

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>abstract</strong></td>
<td><strong>assert</strong></td>
<td><strong>boolean</strong></td>
<td><strong>break</strong></td>
<td><strong>byte</strong></td>
</tr>
<tr>
<td><strong>case</strong></td>
<td><strong>catch</strong></td>
<td><strong>char</strong></td>
<td><strong>class</strong></td>
<td><strong>const</strong></td>
</tr>
<tr>
<td><strong>continue</strong></td>
<td><strong>default</strong></td>
<td><strong>do</strong></td>
<td><strong>double</strong></td>
<td><strong>else</strong></td>
</tr>
<tr>
<td><strong>enum</strong></td>
<td><strong>extends</strong></td>
<td><strong>final</strong></td>
<td><strong>finally</strong></td>
<td><strong>float</strong></td>
</tr>
<tr>
<td><strong>for</strong></td>
<td><strong>goto</strong></td>
<td><strong>if</strong></td>
<td><strong>implements</strong></td>
<td><strong>import</strong></td>
</tr>
<tr>
<td><strong>instanceof</strong></td>
<td><strong>int</strong></td>
<td><strong>interface</strong></td>
<td><strong>long</strong></td>
<td><strong>native</strong></td>
</tr>
<tr>
<td><strong>new</strong></td>
<td><strong>package</strong></td>
<td><strong>private</strong></td>
<td><strong>protected</strong></td>
<td><strong>public</strong></td>
</tr>
<tr>
<td><strong>return</strong></td>
<td><strong>short</strong></td>
<td><strong>static</strong></td>
<td><strong>strictfp</strong></td>
<td><strong>super</strong></td>
</tr>
<tr>
<td><strong>switch</strong></td>
<td><strong>synchronized</strong></td>
<td><strong>this</strong></td>
<td><strong>throw</strong></td>
<td><strong>throws</strong></td>
</tr>
<tr>
<td><strong>transient</strong></td>
<td><strong>try</strong></td>
<td><strong>void</strong></td>
<td><strong>volatile</strong></td>
<td><strong>while</strong></td>
</tr>
</tbody>
</table>

Keywords in Python Programming

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>False</strong></td>
<td><strong>await</strong></td>
<td><strong>else</strong></td>
<td><strong>import</strong></td>
<td><strong>pass</strong></td>
</tr>
<tr>
<td><strong>None</strong></td>
<td><strong>break</strong></td>
<td><strong>except</strong></td>
<td><strong>in</strong></td>
<td><strong>raise</strong></td>
</tr>
<tr>
<td><strong>True</strong></td>
<td><strong>class</strong></td>
<td><strong>finally</strong></td>
<td><strong>is</strong></td>
<td><strong>return</strong></td>
</tr>
<tr>
<td><strong>and</strong></td>
<td><strong>continue</strong></td>
<td><strong>for</strong></td>
<td><strong>lambda</strong></td>
<td><strong>try</strong></td>
</tr>
<tr>
<td><strong>as</strong></td>
<td><strong>def</strong></td>
<td><strong>from</strong></td>
<td><strong>nonlocal</strong></td>
<td><strong>while</strong></td>
</tr>
<tr>
<td><strong>assert</strong></td>
<td><strong>del</strong></td>
<td><strong>global</strong></td>
<td><strong>not</strong></td>
<td><strong>with</strong></td>
</tr>
<tr>
<td><strong>async</strong></td>
<td><strong>elif</strong></td>
<td><strong>if</strong></td>
<td><strong>or</strong></td>
<td><strong>yield</strong></td>
</tr>
</tbody>
</table>

C Language Basic Syntax Rules

```c
#include<stdio.h>      // including header files

int main()            // main() function must be there
{
    // Hello World Program
    printf("Hello, world!");      // Single line comment
    return 0;
}                           // Program enclosed within curly braces
```

452
### Keywords in PHP Programming

<table>
<thead>
<tr>
<th>__halt_compiler()</th>
<th>abstract</th>
<th>and</th>
<th>array()</th>
<th>as</th>
</tr>
</thead>
<tbody>
<tr>
<td>break</td>
<td>callable (as of PHP 5.4)</td>
<td>case</td>
<td>catch</td>
<td>class</td>
</tr>
<tr>
<td>clone</td>
<td>const</td>
<td>continue</td>
<td>declare</td>
<td>default</td>
</tr>
<tr>
<td>die()</td>
<td>do</td>
<td>echo</td>
<td>else</td>
<td>elseif</td>
</tr>
<tr>
<td>empty()</td>
<td>enddeclare</td>
<td>endfor</td>
<td>endforeach</td>
<td>endif</td>
</tr>
<tr>
<td>endswitch</td>
<td>endwhile</td>
<td>eval()</td>
<td>exit()</td>
<td>extends</td>
</tr>
<tr>
<td>finally</td>
<td>finally (as of PHP 5.5)</td>
<td>for</td>
<td>foreach</td>
<td>function</td>
</tr>
<tr>
<td>global</td>
<td>goto (as of PHP 5.3)</td>
<td>if</td>
<td>implements</td>
<td>include</td>
</tr>
<tr>
<td>include_once</td>
<td>instanceof</td>
<td>insteadof (as of PHP 5.4)</td>
<td>interface</td>
<td>isset()</td>
</tr>
<tr>
<td>list()</td>
<td>namespace (as of PHP 5.3)</td>
<td>new</td>
<td>or</td>
<td>print</td>
</tr>
<tr>
<td>private</td>
<td>protected</td>
<td>public</td>
<td>require</td>
<td>require_once</td>
</tr>
<tr>
<td>return</td>
<td>static</td>
<td>switch</td>
<td>throw</td>
<td>trait (as of PHP 5.4)</td>
</tr>
<tr>
<td>try</td>
<td>unset()</td>
<td>use</td>
<td>var</td>
<td>while</td>
</tr>
<tr>
<td>xor</td>
<td>yield (as of PHP 5.5)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Keywords in JavaScript Programming

- await
- break
- case
- catch
- class
- const
- continue
- debugger
- default
- delete
- do
- else
- enum
- export
- extends
- false
- finally
- for
- function
- if
- implements
- import
- in
- instanceof
- interface
- let
- new
- null
- package
- private
- protected
- public
- return
- super
- switch
- static
- this
- throw
- try
- True
- `typeof`
- `var`
- `void`
- `while`
- `with`
- `yield`

**Java Program**

text file named `HelloWorld.java`

```
public class HelloWorld {
    public static void main(String [] args) {
        System.out.println("Hello, World!");
    }
}
```

"Just as it is a good practice to make all fields private unless they need greater visibility, it is a good practice to make all fields final unless they need to be mutable."

— Brian Goetz, Java Concurrency in Practice
Tableau → data visualization tool

- Combines data from multiple sources
- Applies logic and mathematics to data to provide insights for making better quick decisions
- Analyze multiple independent datasets to gain the combined insight
- Visualize and create interactive, shareable dashboards.

6 steps involved in Software Testing Life Cycle:

- Requirement Analysis
- Testing strategy is defined
- Testing team prepares the test data for testing
- Set up the test environment
- Test analysts test the data and compare the expected and actual results.
- Note the summary of the bugs removed and errors found during the software development life cycle.

A blockchain is a backlinked, decentralized and distributed-database of encrypted records, called blocks, which are linked and secured using cryptography.
The most common Shells used in Linux are:

- bash (Bourne Again Shell)
- ksh (Korn Shell)
- csh (C Shell)
- zsh (Z Shell)
- fish (Friendly Interactive Shell)

Commands for Checking Memory Usage in Linux:

- `free -m`
- `vmstat`
- `top`
- `htop`

3 kinds of permission in Linux:

- **r** – **Reading permission:** Allows a user to open and read the file
- **w** – **Writing permission:** Allows a user to open and modify the file
- **x** – **Execution permission:** Allows a user to run the file.

The process states in Linux:

- The process is created
- The process is ready to run
- The process is running
- The process is waiting for input from the user
- The process has been stopped
- The process is dead but its information still exists in the process table
01. Hello World:

```bash
# The following Bash script will print the text "Hello World!" as output.

echo "Hello World"
```

02. Echo Command:

```bash
echo "Printing text with newline"
echo -n "Printing text without newline"
echo -e "\nRemoving \t backslash \t characters\n"
```

03. Comments:

```bash
# Multiply two numeric values
((product=5*3))

#Print the result
echo $product
```
04. Multi-line comment:


:
The following script calculates
the cube value of the number, 2.

((x=2*2*2))
echo $x

05. While Loop:

x=true
count=1
while [ $x ]
do
echo $count
if [ $count -eq 6 ];
then
break
fi
((count++))
done

06. For Loop:
# for loop will iterate for 10 times and print all values of the variable, x in single line.
for (( x=10; x>0; x-- ))
do
echo -n "$x 

done
printf "\n"

07. Get User Input:

echo "Enter Your Name"
read name
echo "Welcome $name to Myw3schools.com"

08. If statement:

x=9
if [ $x -lt 10 ];
then
echo "It is a one digit number"
else
echo "It is a two digit number"
fi
09. And Condition if statement:

```bash
echo "Enter username"
read username
echo "Enter password"
read password

if [[ ( $username == "manju" && $password == "123" ) ]]; then
  echo "valid user"
else
  echo "invalid user"
fi
```

10. Or Condition if statement:

```bash
echo "Enter any number"
read x

if [[ ( $x -eq 12 || $x -eq 56 )]]
then
  echo "your guess is correct"
else
  echo "your guess is wrong"
fi
11. Else if and else condition:

```
echo "Enter any number"
read x

if [ $x -eq 4 ];
then
    echo "You won the first place"
elseif [ $x -eq 20 ];
then
    echo "You won the second place"
elseif [ $x -eq 840 ];
then
    echo "You won the third place"
else
    echo "Sorry, try again next time"
fi
```

12. Case Condition:

```
echo "Enter any number"
read x
case $x in
  4)
    echo echo "You won the first place" ;;
  20)
    echo "You won the second place" ;;
  840)
  *)
    echo "Invalid case" ;;
*)
```
echo "You won the third place" ;
*)
echo "Sorry, try again next time" ;
esac

13. Get Arguments from Command Line:

echo "Total arguments : $#"
echo "First Argument = $1"
echo "Second argument = $2"

14. Get arguments from command line with names:

for arg in "$@"
do
index=$(echo $arg | cut -f1 -d=)
val=$(echo $arg | cut -f2 -d=)
case $index in
  X) x=$val;;
  Y) y=$val;;
  *)
esac
done
((result=x+y))
echo "X+Y=$result"

15. Combine two strings in a variable:

x="Red Hat"
y="Linux"
echo "$x$y"
z=$x+$y
z+=" was a widely used Linux distribution until its discontinuation in 2004."
echo $z

16. Get Substring of Strings:

# The value, 6 indicates the starting point from where the substring will start
# and 5 indicates the length of the substring.

Str="Learn Greek in 30 minutes"
subStr=${Str:6:5}
echo $subStr
17. Multiply 2 numbers into a variable:

```bash
echo "Enter any number"
read a
echo "Enter any number"
read b
(( product=a*b ))
echo "The result of multiplication=$product"
```

18. Create a Function:

```bash
function G2()
{
    echo 'Bash is a Unix shell and command language.'
}
G2
```

19. Use Function Parameters:

```bash
Rectangle_Volume() {
    volume=($(1 * 2 * 3))
echo "Volume is : $volume"
}
```
20. Pass Return Value from Script:

```bash
function welcome() {
    str="Hi, $name"
    echo $str
}

echo "Enter your name"
read name

val=$(welcome)
echo "Return value of the function is $val"
```

21. Make directory:

```bash
echo "Enter directory name"
read newdir
mkdir $newdir`
```
22. Make directory by checking existence:

```
echo "Enter directory name"
read ndir
if [ -d "$ndir" ]
then
echo "Directory exist"
else
`mkdir $ndir`
echo "Directory created"
fi
```

23. Read a file:

```
file=’1.txt’
while read line; do
echo $line
done < $file
```

24. Delete a File:

```
echo "Enter filename to remove"
read fn
rm -i $fn
```
25. Append to file:

```bash
echo "Before appending the file"
cat 1.txt

echo "Learning Red Hat Linux">> 1.txt
echo "After appending the file"
cat 1.txt
```

26. Test if File Exists:

```bash
filename=$1
if [ -f "$filename" ]; then
echo "File exists"
else
echo "File does not exist"
fi
```

27. Send Email Example:

```bash
Recipient="admin@example.com"
```
Subject="Greeting"
Message="Welcome to our Website"
`mail -s $Subject $Recipient <<< $Message`

28. Get Parse Current Date:

Year=`date +%Y`
Month=`date +%m`
Day=`date +%d`
Hour=`date +%H`
Minute=`date +%M`
Second=`date +%S`
echo `date`
echo "Current Date is: $Day-$Month-$Year"
echo "Current Time is: $Hour:$Minute:$Second"

29. Wait Command:

echo "Wait command" &
process_id=$!
wait $process_id
echo "Exited with status $?"

30. Sleep Command:
echo "Wait for 5 seconds"
sleep 5
echo "Completed"

Write a program to print PID of the current shell

```bash
for PID in $$
do
echo $PID
done
```

Write a program to print all array elements and their respective indexes

```bash
array=('Bash' 'is' 'Unix shell' 'Command language')
echo ${array[@]}
echo ${!array[@]}
```

Write a program to print the first array element

```bash
array=('Bash' 'is' 'Unix shell' 'Command language')
echo ${array[0]}
```

Write a program to get current date, time, user name and current working directory

```bash
echo "Hello, $LOGNAME"
echo "Today's date is `date`"
echo "Username is `who i am`"
```
Write a program to print a given number, in reverse order such that the input is provided using command Line Argument only.

```bash
if [ $# -ne 1 ]
  then
  echo "Usage: $0 number"
  echo " Reverse of the given number will be printed"
  echo " For eg. $0 0123, 3210 will be printed"
  exit 1
  fi

n=$1
rev=0
sd=0

while [ $n -gt 0 ]
  do
    sd=`expr $n % 10`
    rev=`expr $rev * 10 + $sd`
    n=`expr $n / 10`
  done
```
JVM (Java Virtual Machine) resides under RAM (Random Access Memory – the stuff that boost up your computer to run faster and allows your computer to perform many tasks at the same time) and it comprises

- **CLASS LOADER**: it loads .class file that contains Java byte codes.
- **BYTE CODE VERIFIER**: it verifies byte codes.
- **EXECUTION ENGINE**: it translates java byte codes to machine codes and executes them.

**HTML Element Reference**

**Basic HTML**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;!DOCTYPE&gt;</td>
<td>Defines the document type</td>
</tr>
<tr>
<td>&lt;html&gt;</td>
<td>Defines an HTML document</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>&lt;head&gt;</code></td>
<td>Contains metadata/information for the document</td>
</tr>
<tr>
<td><code>&lt;title&gt;</code></td>
<td>Defines a title for the document</td>
</tr>
<tr>
<td><code>&lt;body&gt;</code></td>
<td>Defines the document's body</td>
</tr>
<tr>
<td><code>&lt;h1&gt;</code> to <code>&lt;h6&gt;</code></td>
<td>Defines HTML headings</td>
</tr>
<tr>
<td><code>&lt;p&gt;</code></td>
<td>Defines a paragraph</td>
</tr>
<tr>
<td><code>&lt;br&gt;</code></td>
<td>Inserts a single line break</td>
</tr>
<tr>
<td><code>&lt;hr&gt;</code></td>
<td>Defines a thematic change in the content</td>
</tr>
<tr>
<td><code>&lt;!--...--&gt;</code></td>
<td>Defines a comment</td>
</tr>
</tbody>
</table>

**Formatting**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;acronym&gt;</code></td>
<td>Not supported in HTML5. Use <code>&lt;abbr&gt;</code> instead. Defines an acronym</td>
</tr>
<tr>
<td><code>&lt;abbr&gt;</code></td>
<td>Defines an abbreviation or an acronym</td>
</tr>
<tr>
<td><code>&lt;address&gt;</code></td>
<td>Defines contact information for the author/owner of a</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>&lt;b&gt;</td>
<td>Defines bold text</td>
</tr>
<tr>
<td>&lt;bdi&gt;</td>
<td>Isolates a part of text that might be formatted in a different direction from other text outside it</td>
</tr>
<tr>
<td>&lt;bdo&gt;</td>
<td>Overrides the current text direction</td>
</tr>
<tr>
<td>&lt;big&gt;</td>
<td>Not supported in HTML5. Use CSS instead. Defines big text</td>
</tr>
<tr>
<td>&lt;blockquote&gt;</td>
<td>Defines a section that is quoted from another source</td>
</tr>
<tr>
<td>&lt;center&gt;</td>
<td>Not supported in HTML5. Use CSS instead. Defines centered text</td>
</tr>
<tr>
<td>&lt;cite&gt;</td>
<td>Defines the title of a work</td>
</tr>
<tr>
<td>&lt;code&gt;</td>
<td>Defines a piece of computer code</td>
</tr>
<tr>
<td>&lt;del&gt;</td>
<td>Defines text that has been deleted from a document</td>
</tr>
<tr>
<td>&lt;dfn&gt;</td>
<td>Specifies a term that is going to be defined within the content</td>
</tr>
<tr>
<td>&lt;em&gt;</td>
<td>Defines emphasized text</td>
</tr>
<tr>
<td>&lt;font&gt;</td>
<td>Not supported in HTML5. Use CSS instead.</td>
</tr>
<tr>
<td>Tag</td>
<td>Definition</td>
</tr>
<tr>
<td>-----</td>
<td>------------</td>
</tr>
<tr>
<td>&lt;i&gt;</td>
<td>Defines a part of text in an alternate voice or mood</td>
</tr>
<tr>
<td>&lt;ins&gt;</td>
<td>Defines a text that has been inserted into a document</td>
</tr>
<tr>
<td>&lt;kbd&gt;</td>
<td>Defines keyboard input</td>
</tr>
<tr>
<td>&lt;mark&gt;</td>
<td>Defines marked/highlighted text</td>
</tr>
<tr>
<td>&lt;meter&gt;</td>
<td>Defines a scalar measurement within a known range (a gauge)</td>
</tr>
<tr>
<td>&lt;pre&gt;</td>
<td>Defines preformatted text</td>
</tr>
<tr>
<td>&lt;progress&gt;</td>
<td>Represents the progress of a task</td>
</tr>
<tr>
<td>&lt;q&gt;</td>
<td>Defines a short quotation</td>
</tr>
<tr>
<td>&lt;rp&gt;</td>
<td>Defines what to show in browsers that do not support ruby annotations</td>
</tr>
<tr>
<td>&lt;rt&gt;</td>
<td>Defines an explanation/pronunciation of characters (for East Asian typography)</td>
</tr>
<tr>
<td>&lt;ruby&gt;</td>
<td>Defines a ruby annotation (for East Asian typography)</td>
</tr>
<tr>
<td>&lt;s&gt;</td>
<td>Defines text that is no longer correct</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>&lt;samp&gt;</td>
<td>Defines sample output from a computer program</td>
</tr>
<tr>
<td>&lt;small&gt;</td>
<td>Defines smaller text</td>
</tr>
<tr>
<td>&lt;strike&gt;</td>
<td>Not supported in HTML5. Use &lt;del&gt; or &lt;s&gt; instead. Defines strikethrough text</td>
</tr>
<tr>
<td>&lt;strong&gt;</td>
<td>Defines important text</td>
</tr>
<tr>
<td>&lt;sub&gt;</td>
<td>Defines subscripted text</td>
</tr>
<tr>
<td>&lt;sup&gt;</td>
<td>Defines superscripted text</td>
</tr>
<tr>
<td>&lt;template&gt;</td>
<td>Defines a container for content that should be hidden</td>
</tr>
<tr>
<td></td>
<td>when the page loads</td>
</tr>
<tr>
<td>&lt;time&gt;</td>
<td>Defines a specific time (or datetime)</td>
</tr>
<tr>
<td>&lt;tt&gt;</td>
<td>Not supported in HTML5. Use CSS instead. Defines teletype text</td>
</tr>
<tr>
<td>&lt;u&gt;</td>
<td>Defines some text that is unarticulated and styled differently from normal text</td>
</tr>
<tr>
<td>&lt;var&gt;</td>
<td>Defines a variable</td>
</tr>
<tr>
<td>&lt;wbr&gt;</td>
<td>Defines a possible line-break</td>
</tr>
</tbody>
</table>
01. Hello World:

# The following python script will print the text "Hello World!" as output.

print("Hello World!")

02. Join two strings:

# Two words "Programming" and "Languages" are joined and "ProgrammingLanguages" is printed as output.

x = "Programming"
y = "Languages"
z = x + y
print(z)

03. Format floating point in the string:
# Use of String Formatting
\[ x = 462.75897 \]
print("{:5.2f}".format(x))

# Use of String Interpolation
\[ y = 462.75897 \]
print("%5.2f" % y)

---

04. Raise a number to a power:

```python
import math
# Assign values to a and n
a = 4
n = 3

# Method 1
b = a ** n
print("%d to the power %d is %d" % (a, n, b))

# Method 2
b = pow(a, n)
print("%d to the power %d is %d" % (a, n, b))

# Method 3
b = math.pow(a, n)
print("%d to the power %d is %5.2f" % (a, n, b))
```

---

05. Working with Boolean types:
# Boolean value

```python
x = True
print(x)
```

```python
# Number to Boolean
x = 10
print(bool(x))
```

```python
x = -5
print(bool(x))
```

```python
x = 0
print(bool(x))
```

# Boolean from comparison operator

```python
x = 6
y = 3
print(x < y)
```

## 06. If else statement:

```python
# Assign a numeric value
x = 35

# Check the is more than 35 or not
if (x >= 35):
    print("You have passed")
else:
    print("You have not passed")
```
07. Using AND and OR operators:

# Take practical marks
x = float(input("Enter the Practical marks: "))
# Take theory marks
y = float(input("Enter the Theory marks: "))

# Check the passing condition using AND and OR operator
if (x >= 25 and y >= 45) or (x + y) >=70:
    print("\nYou have passed")
else:
    print("\nYou have failed")

08. Switch case statement:

# Switcher for implementing switch case options
def employee_details(ID):
    switcher = {
        "5006": "Employee Name: John",
        "5008": "Employee Name: Ram",
        "5010": "Employee Name: Mohamend",
    }

    '''The first argument will be returned if the match found and
    employee ID does not exist will be returned if no match found'''
    return switcher.get(ID, "employee ID does not exist")

# Take the employee ID
ID = input("Enter the employee ID: ")
# Print the output
print(employee_details(ID))

09. While Loop:

# Initialize counter
counter = 1
# Iterate the loop 9 times
while counter < 10:
    # Print the counter value
    print("%d" % counter)
    # Increment the counter
    counter = counter + 1

11. Use of command-line argument:

# Import sys module
import sys

# Total number of arguments
print('Total arguments:', len(sys.argv))

print("Argument values are:")
# Iterate command-line arguments using for loop
for i in sys.argv:
    print(i)
12. Run one Python script from another:

```python
# Hello.py

print("Hello World!")
# Executes the hello.py file in the interpreter

import os
os.system('hello.py')
```

13. Use of regex:

```python
# Import re module
import re

# Take any string data
string = input("Enter a string value: ")
# Define the searching pattern
pattern = '^[A-Z]'

# match the pattern with input value
found = re.match(pattern, string)

# Print message based on the return value
if found:
    print("The input value is started with the capital letter")
else:
```
print("You have to type string start with the capital letter")

14. Use of getpass:

# import getpass module
import getpass

# Take password from the user
passwd = getpass.getpass('Password:')

# Check the password
if passwd == "python":
    print("You are verified")
else:
    print("You are not verified")

15. Use of date format:

from datetime import date

# Read the current date
current_date = date.today()

# Print the formatted date
print("Today is :%d-%d-%d" % (current_date.day, current_date.month, current_date.year))

# Set the custom date
custom_date = date(2026, 12, 26)
print("The date is:", custom_date)

16. Add and remove the item from a list:

# Declare a fruit list
fruits = ["Mango", "Orange", "Guava", "Banana"]

# Insert an item in the 2nd position
fruits.insert(1, "Apple")

# Displaying list after inserting
print("The fruit list after insert:")
print(fruits)

# Remove an item
fruits.remove("Banana")

# Print the list after delete
print("The fruit list after delete:")
print(fruits)

17. List comprehension:

# Create a list of characters using list comprehension
char_list = [ char for char in "Python" ]
print(char_list)
# Define a tuple of websites
websites = ("google.com","yahoo.com", "history.com", "quora.com")

# Create a list from tuple using list comprehension
site_list = [ site for site in websites ]
print(site_list)

18. Slice data:

# Assign string value
text = "Python Programming Language"

# Slice using one parameter
sliceObj = slice(5)
print(text[sliceObj])

# Slice using two parameter
sliceObj = slice(6,12)
print(text[sliceObj])

# Slice using three parameter
sliceObj = slice(6,25,5)
print(text[sliceObj])

19. Add and search data in the dictionary:
# Define a dictionary
customers = {'05453': 'Ram', '04457': 'Krishna',
'02834': 'Vishnu', '01655': 'Shiva', '07559': 'David'}

# Append a new data
customers['06934'] = 'Salomon'

print("The customer names are:")
# Print the values of the dictionary
for customer in customers:
    print(customers[customer])

# Take customer ID as input to search
name = input("Enter customer ID:")

# Search the ID in the dictionary
for customer in customers:
    if customer == name:
        print(customers[customer])
        break

20. Add and search data in the set:

# Define the number set
numbers = {13, 10, 56, 18, 12, 44, 87}

# Add a new data
numbers.add(63)
# Print the set values
print(numbers)
message = "Number is not found"

# Take a number value for search
search_number = int(input("Enter a number:"))

# Search the number in the set
for val in numbers:
    if val == search_number:
        message = "Number is found"
        break

print(message)

21. Count items in the list:

# Define the string
string = 'Python Go JavaScript HTML CSS MYSQL Python'

# Define the search string
search = 'Python'

# Store the count value
count = string.count(search)

# Print the formatted output
print("%s appears %d times" % (search, count))

22. Define and call a function:
# Define addition function
def addition(number1, number2):
    result = number1 + number2
    print("Addition result:", result)

# Define area function with return statement
def area(radius):
    result = 3.14 * radius * radius
    return result

# Call addition function
addition(5, 3)
# Call area function
print("Area of the circle is", area(2))

---

23. Use of throw and catch exception:

# Try block
try:
    # Take a number
    number = int(input("Enter a number: "))
    if number % 2 == 0:
        print("Number is even")
    else:
        print("Number is odd")

# Exception block
except (ValueError):
    # Print error message
    print("Enter a numeric value")
24. Read and Write File:

```python
#Assign the filename
filename = "names.txt"
# Open file for writing
fileHandler = open(filename, "w")

# Add some text
fileHandler.write("Ram\n")
fileHandler.write("John\n")
fileHandler.write("David\n")

# Close the file
fileHandler.close()

# Open file for reading
fileHandler = open(filename, "r")

# Read a file line by line
for line in fileHandler:
    print(line)

# Close the file
fileHandler.close()
```

25. List files in a directory:
import os

# Set the directory path
path = '/Users/Manju/'

# Read the content of the file
files = os.listdir(path)

# Print the content of the directory
for file in files:
    print(file)

# Import pickle module
import pickle

# Declare the object to store data
dataObject = []

# Iterate the for loop for 5 times
for num in range(10, 15):
    dataObject.append(num)

# Open a file for writing data
file_handler = open('numbers', 'wb')
# Dump the data of the object into the file
pickle.dump(dataObject, file_handler)
# Close the file handler
file_handler.close()
file_handler = open('numbers', 'rb')
# Load the data from the file after deserialization
dataObject = pickle.load(file_handler)
# Iterate the loop to print the data
for val in dataObject:
    print(val)
# close the file handler
file_handler.close()

27. Define class and method:

# Define the class
class Employee:
    name = "Johnson"
    # Define the method
    def details(self):
        print("Post: Associate")
        print("Department: QC")
        print("Salary: $6000")

# Create the employee object
emp = Employee()
# Print the class variable
print("Name:", emp.name)
# Call the class method
emp.details()

28. Use of range function:
# range() with one parameter
for val in range(7):
    print(val, end=' ')
print('
')

# range() with two parameter
for val in range(7, 16):
    print(val, end=' ')
print('
')

# range() with three parameter
for val in range(0, 4, 1):
    print(val, end=' ')

29. Use of map function:

# Define the function to calculate power
def cal_power(n):
    return x ** n

# Take the value of x
x = int(input("Enter the value of x:"))
# Define a tuple of numbers
numbers = [2, 6, 3]

# Calculate the x to the power n using map()
result = map(cal_power, numbers)
print(list(result))
30. Use of filter function:

# Define a list of participant
participant = ['Ram', 'John', 'David', 'Krishna', 'Prasad']
# Define the function to filters selected candidates
def SelectedPerson(participant):
    selected = ['John', 'Rahman', 'Ram']
    if(participant in selected):
        return True

selectedList = filter(SelectedPerson, participant)
print('The selected candidates are:
for candidate in selectedList:
    print(candidate)

Code:

mul_value = 0

def mul_numbers( num1, num2 ):
    mul_value = num1 * num2;
    print ("Local Value: ", mul_value)
    return mul_value

mul_numbers( 58, 77 )
print ("Global Value: ", mul_value)
Write a program to open and identify the given image file

# Imports PIL module
from PIL import Image

# open method used to open different extension image file
im = Image.open(r"C:\Users\Manju\1.png")

# This method will show image in any image viewer
im.show()

Write a program to display hostname and IP address

# Imports socket library
import socket

# Function to display hostname and IP address
def get_Host_name_IP():
    try:
        host_name = socket.gethostname()
        host_ip = socket.gethostbyname(host_name)
        print("Hostname : ",host_name)
        print("IP : ",host_ip)
except:
    print("Unable to get Hostname and IP")

# Driver code
get_Host_name_IP() #Function call

Code:

# install PyPDF2 using the command: pip install PyPDF2

import PyPDF2

pdf = open('1.pdf', 'rb')
read_pdf = PyPDF2.PdfFileReader(pdf)

print("Number of pages in pdf: ", read_pdf.numPages)
pdf.close()

Output on the screen:

Number of pages in pdf: 13
### Forms and Input

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;form&gt;</td>
<td>Defines an HTML form for user input</td>
</tr>
<tr>
<td>&lt;input&gt;</td>
<td>Defines an input control</td>
</tr>
<tr>
<td>&lt;textarea&gt;</td>
<td>Defines a multiline input control (text area)</td>
</tr>
<tr>
<td>&lt;button&gt;</td>
<td>Defines a clickable button</td>
</tr>
<tr>
<td>&lt;select&gt;</td>
<td>Defines a drop-down list</td>
</tr>
<tr>
<td>&lt;optgroup&gt;</td>
<td>Defines a group of related options in a drop-down list</td>
</tr>
<tr>
<td>&lt;option&gt;</td>
<td>Defines an option in a drop-down list</td>
</tr>
<tr>
<td>&lt;label&gt;</td>
<td>Defines a label for an &lt;input&gt; element</td>
</tr>
<tr>
<td>&lt;fieldset&gt;</td>
<td>Groups related elements in a form</td>
</tr>
<tr>
<td>&lt;legend&gt;</td>
<td>Defines a caption for a &lt;fieldset&gt; element</td>
</tr>
<tr>
<td>&lt;datalist&gt;</td>
<td>Specifies a list of pre-defined options for input controls</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>&lt;output&gt;</td>
<td>Defines the result of a calculation</td>
</tr>
</tbody>
</table>

**Frames**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;frame&gt;</td>
<td>Not supported in HTML5. Defines a window (a frame) in a frameset</td>
</tr>
<tr>
<td>&lt;frameset&gt;</td>
<td>Not supported in HTML5. Defines a set of frames</td>
</tr>
<tr>
<td>&lt;noframes&gt;</td>
<td>Not supported in HTML5. Defines an alternate content for users that do not support frames</td>
</tr>
<tr>
<td>&lt;iframe&gt;</td>
<td>Defines an inline frame</td>
</tr>
</tbody>
</table>

**Images**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;img&gt;</td>
<td>Defines an image</td>
</tr>
<tr>
<td>&lt;map&gt;</td>
<td>Defines a client-side image map</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>&lt;area&gt;</td>
<td>Defines an area inside an image map</td>
</tr>
<tr>
<td>&lt;canvas&gt;</td>
<td>Used to draw graphics, on the fly, via scripting (usually JavaScript)</td>
</tr>
<tr>
<td>&lt;figcaption&gt;</td>
<td>Defines a caption for a &lt;figure&gt; element</td>
</tr>
<tr>
<td>&lt;figure&gt;</td>
<td>Specifies self-contained content</td>
</tr>
<tr>
<td>&lt;picture&gt;</td>
<td>Defines a container for multiple image resources</td>
</tr>
<tr>
<td>&lt;svg&gt;</td>
<td>Defines a container for SVG graphics</td>
</tr>
</tbody>
</table>

**Audio / Video**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;audio&gt;</td>
<td>Defines sound content</td>
</tr>
<tr>
<td>&lt;source&gt;</td>
<td>Defines multiple media resources for media elements (&lt;video&gt;, &lt;audio&gt; and &lt;picture&gt;)</td>
</tr>
<tr>
<td>&lt;track&gt;</td>
<td>Defines text tracks for media elements (&lt;video&gt; and &lt;audio&gt;)</td>
</tr>
<tr>
<td>&lt;video&gt;</td>
<td>Defines a video or movie</td>
</tr>
</tbody>
</table>
### Links

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;a&gt;</code></td>
<td>Defines a hyperlink</td>
</tr>
<tr>
<td><code>&lt;link&gt;</code></td>
<td>Defines the relationship between a document and an external resource (most used to link to style sheets)</td>
</tr>
<tr>
<td><code>&lt;nav&gt;</code></td>
<td>Defines navigation links</td>
</tr>
</tbody>
</table>

### Lists

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;ul&gt;</code></td>
<td>Defines an unordered list</td>
</tr>
<tr>
<td><code>&lt;ol&gt;</code></td>
<td>Defines an ordered list</td>
</tr>
<tr>
<td><code>&lt;li&gt;</code></td>
<td>Defines a list item</td>
</tr>
<tr>
<td><code>&lt;dir&gt;</code></td>
<td>Not supported in HTML5. Use <code>&lt;ul&gt;</code> instead.</td>
</tr>
<tr>
<td><code>&lt;dl&gt;</code></td>
<td>Defines a description list</td>
</tr>
<tr>
<td><code>&lt;dt&gt;</code></td>
<td>Defines a term/name in a description list</td>
</tr>
</tbody>
</table>
<dd> Defines a description of a term/name in a description list

**Tables**

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;table&gt;</td>
<td>Defines a table</td>
</tr>
<tr>
<td>&lt;caption&gt;</td>
<td>Defines a table caption</td>
</tr>
<tr>
<td>&lt;th&gt;</td>
<td>Defines a header cell in a table</td>
</tr>
<tr>
<td>&lt;tr&gt;</td>
<td>Defines a row in a table</td>
</tr>
<tr>
<td>&lt;td&gt;</td>
<td>Defines a cell in a table</td>
</tr>
<tr>
<td>&lt;thead&gt;</td>
<td>Groups the header content in a table</td>
</tr>
<tr>
<td>&lt;tbody&gt;</td>
<td>Groups the body content in a table</td>
</tr>
<tr>
<td>&lt;tfoot&gt;</td>
<td>Groups the footer content in a table</td>
</tr>
<tr>
<td>&lt;col&gt;</td>
<td>Specifies column properties for each column within a &lt;colgroup&gt; element</td>
</tr>
<tr>
<td>&lt;colgroup&gt;</td>
<td>Specifies a group of one or more columns in a table for formatting</td>
</tr>
</tbody>
</table>
from selenium import webdriver
from selenium.webdriver.common.keys import Keys

driver = webdriver.Chrome('chromedriver.exe')
driver.get("https://www.python.org")
print(driver.title)
search_bar = driver.find_element_by_name("q")
search_bar.clear()
search_bar.send_keys("getting started with python")
search_bar.send_keys(Keys.RETURN)
print(driver.current_url)
driver.close()

from selenium import webdriver

driver = webdriver.Chrome('chromedriver.exe')
driver.get('http://www.google.com/');
time.sleep(5)
search_box = driver.find_element_by_name('q')
search_box.send_keys('ChromeDriver')
search_box.submit()
time.sleep(5)
driver.quit()
import time

from selenium import webdriver
from selenium.webdriver.chrome.service import Service

service = Service('chromedriver.exe')
service.start()
driver = webdriver.Remote(service.service_url)
driver.get('http://www.google.com/');
time.sleep(5)
driver.quit()

import os
from selenium import webdriver

# get the path of ChromeDriverServer
dir = os.path.dirname(__file__)
chrome_driver_path = "chromedriver.exe"

# create a new Chrome session
driver = webdriver.Chrome(chrome_driver_path)
driver.implicitly_wait(30)
driver.maximize_window()

# Navigate to the application home page
driver.get("http://www.google.com")

# get the search textbox
search_field = driver.find_element_by_name("q")
from selenium import webdriver
import time

# set webdriver path here it may vary
browser = webdriver.Chrome(executable_path = "chromedriver.exe")

# enter search keyword and submit
search_field.send_keys("Selenium WebDriver Interview questions")
search_field.submit()

# get the list of elements which are displayed after the search
# currently on result page using find_elements_by_class_name method
lists = driver.find_elements_by_class_name("r")

# get the number of elements found
print ("Found " + str(len(lists)) + " searches:")

# iterate through each element and print the text that is
# name of the search

i=0
for listitem in lists:
    print (listitem.get_attribute("innerHTML"))
    i=i+1
    if(i>10):
        break

# close the browser window
driver.quit()
website_URL = "https://www.google.co.in/"

browser.get(website_URL)

# After how many seconds you want to refresh the webpage
# Few website count view if you stay there
# for a particular time
# you have to figure that out
refreshrate = int(15)

# This would keep running until you stop the compiler.
while True:
    time.sleep(refreshrate)
    browser.refresh()

from selenium import webdriver
# sleep() function is required because
# selenium needs a page to be fully loaded first
# otherwise errors may occur
from time import sleep
# Usage sleep(x) Where x is time in seconds and
# may vary according to your connection

# I have made class so that extra methods can be added later on
# if required
class instagramBot:
    def __init__(self, username, password):
        # these lines will help if someone faces issues like
        # chrome closes after execution
self.opts = webdriver.ChromeOptions()
self.opts.add_experimental_option("detach", True)
self.driver = webdriver.Chrome(options=self.opts)

# Username and password
self.username = username
self.password = password

# Opens Instagram login page
self.driver.get("https://instagram.com")
sleep(2)  # 1 Second Wait

# Automatically enters your username and
# password to instagram's username field
self.driver.find_element_by_xpath('//input[@name = 'username']').send_keys(self.username)
self.driver.find_element_by_xpath('//input[@name = 'password']').send_keys(self.password)

# Clicks on Log In Button
self.driver.find_element_by_xpath("//div[contains(text(), 'Log In')]").click()
sleep(2)

# Bonus: Automatically clicks on 'Not Now'
# when Instagram asks to show notifications
self.driver.find_element_by_xpath("//button[contains(text(), 'Not Now')]").click()

# Testing Your Code
instagramBot('Sample Username','Sample Password')
### Styles and Semantics

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;style&gt;</code></td>
<td>Defines style information for a document</td>
</tr>
<tr>
<td><code>&lt;div&gt;</code></td>
<td>Defines a section in a document</td>
</tr>
<tr>
<td><code>&lt;span&gt;</code></td>
<td>Defines a section in a document</td>
</tr>
<tr>
<td><code>&lt;header&gt;</code></td>
<td>Defines a header for a document or section</td>
</tr>
<tr>
<td><code>&lt;footer&gt;</code></td>
<td>Defines a footer for a document or section</td>
</tr>
<tr>
<td><code>&lt;main&gt;</code></td>
<td>Specifies the main content of a document</td>
</tr>
<tr>
<td><code>&lt;section&gt;</code></td>
<td>Defines a section in a document</td>
</tr>
<tr>
<td><code>&lt;article&gt;</code></td>
<td>Defines an article</td>
</tr>
<tr>
<td><code>&lt;aside&gt;</code></td>
<td>Defines content aside from the page content</td>
</tr>
<tr>
<td><code>&lt;details&gt;</code></td>
<td>Defines additional details that the user can view or hide</td>
</tr>
<tr>
<td><code>&lt;dialog&gt;</code></td>
<td>Defines a dialog box or window</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>&lt;summary&gt;</td>
<td>Defines a visible heading for a &lt;details&gt; element</td>
</tr>
<tr>
<td>&lt;data&gt;</td>
<td>Adds a machine-readable translation of a given content</td>
</tr>
</tbody>
</table>

### Meta Info

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;head&gt;</td>
<td>Defines information about the document</td>
</tr>
<tr>
<td>&lt;meta&gt;</td>
<td>Defines metadata about an HTML document</td>
</tr>
<tr>
<td>&lt;base&gt;</td>
<td>Specifies the base URL/target for all relative URLs in a document</td>
</tr>
<tr>
<td>&lt;basefont&gt;</td>
<td>Not supported in HTML5. Use CSS instead. Specifies a default color, size, and font for all text in a document</td>
</tr>
</tbody>
</table>

### Programming

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;script&gt;</td>
<td>Defines a client-side script</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>&lt;noscript&gt;</code></td>
<td>Defines an alternate content for users that do not support client-side scripts</td>
</tr>
<tr>
<td><code>&lt;applet&gt;</code></td>
<td>Not supported in HTML5. Use <code>&lt;embed&gt;</code> or <code>&lt;object&gt;</code> instead. Defines an embedded applet</td>
</tr>
<tr>
<td><code>&lt;embed&gt;</code></td>
<td>Defines a container for an external (non-HTML) application</td>
</tr>
<tr>
<td><code>&lt;object&gt;</code></td>
<td>Defines an embedded object</td>
</tr>
<tr>
<td><code>&lt;param&gt;</code></td>
<td>Defines a parameter for an object</td>
</tr>
</tbody>
</table>

**CSS Reference**

**CSS Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>align-content</strong></td>
<td>Specifies the alignment between the lines inside a flexible container when the items do not use all available space</td>
</tr>
<tr>
<td><strong>align-items</strong></td>
<td>Specifies the alignment for items inside a flexible container</td>
</tr>
<tr>
<td><strong>align-self</strong></td>
<td>Specifies the alignment for selected items inside a flexible container</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>all</strong></td>
<td>Resets all properties (except unicode-bidi and direction)</td>
</tr>
<tr>
<td><strong>animation</strong></td>
<td>A shorthand property for all the animation-* properties</td>
</tr>
<tr>
<td><strong>animation-delay</strong></td>
<td>Specifies a delay for the start of an animation</td>
</tr>
<tr>
<td><strong>animation-direction</strong></td>
<td>Specifies whether an animation should be played forwards, backwards or in alternate cycles</td>
</tr>
<tr>
<td><strong>animation-duration</strong></td>
<td>Specifies how long an animation should take to complete one cycle</td>
</tr>
<tr>
<td><strong>animation-fill-mode</strong></td>
<td>Specifies a style for the element when the animation is not playing (before it starts, after it ends, or both)</td>
</tr>
<tr>
<td><strong>animation-iteration-count</strong></td>
<td>Specifies the number of times an animation should be played</td>
</tr>
<tr>
<td><strong>animation-name</strong></td>
<td>Specifies a name for the @keyframes animation</td>
</tr>
<tr>
<td><strong>animation-play-state</strong></td>
<td>Specifies whether the animation is running or paused</td>
</tr>
<tr>
<td><strong>animation-timing-function</strong></td>
<td>Specifies the speed curve of an animation</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>backface-visibility</td>
<td>Defines whether or not the back face of an element should be visible when facing the user</td>
</tr>
<tr>
<td>background</td>
<td>A shorthand property for all the background-* properties</td>
</tr>
<tr>
<td>background-attachment</td>
<td>Sets whether a background image scrolls with the rest of the page, or is fixed</td>
</tr>
<tr>
<td>background-blend-mode</td>
<td>Specifies the blending mode of each background layer (color/image)</td>
</tr>
<tr>
<td>background-clip</td>
<td>Defines how far the background (color or image) should extend within an element</td>
</tr>
<tr>
<td>background-color</td>
<td>Specifies the background color of an element</td>
</tr>
<tr>
<td>background-image</td>
<td>Specifies one or more background images for an element</td>
</tr>
<tr>
<td>background-origin</td>
<td>Specifies the origin position of a background image</td>
</tr>
<tr>
<td>background-position</td>
<td>Specifies the position of a background image</td>
</tr>
<tr>
<td>background-repeat</td>
<td>Sets if/how a background image will be repeated</td>
</tr>
<tr>
<td>background-size</td>
<td>Specifies the size of the background images</td>
</tr>
<tr>
<td>border</td>
<td>A shorthand property for border-width, border-</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>border-bottom</code></td>
<td>A shorthand property for <code>border-bottom-width</code>, <code>border-bottom-style</code> and <code>border-bottom-color</code></td>
</tr>
<tr>
<td><code>border-bottom-color</code></td>
<td>Sets the color of the bottom border</td>
</tr>
<tr>
<td><code>border-bottom-left-radius</code></td>
<td>Defines the radius of the border of the bottom-left corner</td>
</tr>
<tr>
<td><code>border-bottom-right-radius</code></td>
<td>Defines the radius of the border of the bottom-right corner</td>
</tr>
<tr>
<td><code>border-bottom-style</code></td>
<td>Sets the style of the bottom border</td>
</tr>
<tr>
<td><code>border-bottom-width</code></td>
<td>Sets the width of the bottom border</td>
</tr>
<tr>
<td><code>border-collapse</code></td>
<td>Sets whether table borders should collapse into a single border or be separated</td>
</tr>
<tr>
<td><code>border-color</code></td>
<td>Sets the color of the four borders</td>
</tr>
<tr>
<td><code>border-image</code></td>
<td>A shorthand property for all the <code>border-image-*</code> properties</td>
</tr>
<tr>
<td><code>border-image-outset</code></td>
<td>Specifies the amount by which the border image area extends beyond the border box</td>
</tr>
<tr>
<td><code>border-image-repeat</code></td>
<td>Specifies whether the border image should be repeated, rounded or stretched</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td><code>border-image-slice</code></td>
<td>Specifies how to slice the border image</td>
</tr>
<tr>
<td><code>border-image-source</code></td>
<td>Specifies the path to the image to be used as a border</td>
</tr>
<tr>
<td><code>border-image-width</code></td>
<td>Specifies the width of the border image</td>
</tr>
<tr>
<td><code>border-left</code></td>
<td>A shorthand property for all the <code>border-left-*</code> properties</td>
</tr>
<tr>
<td><code>border-left-color</code></td>
<td>Sets the color of the left border</td>
</tr>
<tr>
<td><code>border-left-style</code></td>
<td>Sets the style of the left border</td>
</tr>
<tr>
<td><code>border-left-width</code></td>
<td>Sets the width of the left border</td>
</tr>
<tr>
<td><code>border-radius</code></td>
<td>A shorthand property for the four <code>border-*radius</code> properties</td>
</tr>
<tr>
<td><code>border-right</code></td>
<td>A shorthand property for all the <code>border-right-*</code> properties</td>
</tr>
<tr>
<td><code>border-right-color</code></td>
<td>Sets the color of the right border</td>
</tr>
<tr>
<td><code>border-right-style</code></td>
<td>Sets the style of the right border</td>
</tr>
<tr>
<td><code>border-right-width</code></td>
<td>Sets the width of the right border</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>border-spacing</code></td>
<td>Sets the distance between the borders of adjacent cells</td>
</tr>
<tr>
<td><code>border-style</code></td>
<td>Sets the style of the four borders</td>
</tr>
<tr>
<td><code>border-top</code></td>
<td>A shorthand property for <code>border-top-width</code>, <code>border-top-style</code> and <code>border-top-color</code></td>
</tr>
<tr>
<td><code>border-top-color</code></td>
<td>Sets the color of the top border</td>
</tr>
<tr>
<td><code>border-top-left-radius</code></td>
<td>Defines the radius of the border of the top-left corner</td>
</tr>
<tr>
<td><code>border-top-right-radius</code></td>
<td>Defines the radius of the border of the top-right corner</td>
</tr>
<tr>
<td><code>border-top-style</code></td>
<td>Sets the style of the top border</td>
</tr>
<tr>
<td><code>border-top-width</code></td>
<td>Sets the width of the top border</td>
</tr>
<tr>
<td><code>border-width</code></td>
<td>Sets the width of the four borders</td>
</tr>
<tr>
<td><code>bottom</code></td>
<td>Sets the elements position, from the bottom of its parent element</td>
</tr>
<tr>
<td><code>box-decoration-break</code></td>
<td>Sets the behavior of the background and border of an element at page-break, or, for in-line elements, at line-break.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>box-shadow</td>
<td>Attaches one or more shadows to an element</td>
</tr>
<tr>
<td>box-sizing</td>
<td>Defines how the width and height of an element are calculated: should they include padding and borders, or not</td>
</tr>
<tr>
<td>break-after</td>
<td>Specifies whether or not a page-, column-, or region-break should occur after the specified element</td>
</tr>
<tr>
<td>break-before</td>
<td>Specifies whether or not a page-, column-, or region-break should occur before the specified element</td>
</tr>
<tr>
<td>break-inside</td>
<td>Specifies whether or not a page-, column-, or region-break should occur inside the specified element</td>
</tr>
<tr>
<td>caption-side</td>
<td>Specifies the placement of a table caption</td>
</tr>
<tr>
<td>caret-color</td>
<td>Specifies the color of the cursor (caret) in inputs, textareas, or any element that is editable</td>
</tr>
<tr>
<td>@charset</td>
<td>Specifies the character encoding used in the style sheet</td>
</tr>
<tr>
<td>clear</td>
<td>Specifies on which sides of an element floating elements are not allowed to float</td>
</tr>
<tr>
<td><strong>clip</strong></td>
<td>Clips an absolutely positioned element</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------------------------</td>
</tr>
<tr>
<td><strong>color</strong></td>
<td>Sets the color of text</td>
</tr>
<tr>
<td><strong>column-count</strong></td>
<td>Specifies the number of columns an element should be divided into</td>
</tr>
<tr>
<td><strong>column-fill</strong></td>
<td>Specifies how to fill columns, balanced or not</td>
</tr>
<tr>
<td><strong>column-gap</strong></td>
<td>Specifies the gap between the columns</td>
</tr>
<tr>
<td><strong>column-rule</strong></td>
<td>A shorthand property for all the column-rule-* properties</td>
</tr>
<tr>
<td><strong>column-rule-color</strong></td>
<td>Specifies the color of the rule between columns</td>
</tr>
<tr>
<td><strong>column-rule-style</strong></td>
<td>Specifies the style of the rule between columns</td>
</tr>
<tr>
<td><strong>column-rule-width</strong></td>
<td>Specifies the width of the rule between columns</td>
</tr>
<tr>
<td><strong>column-span</strong></td>
<td>Specifies how many columns an element should span across</td>
</tr>
<tr>
<td><strong>column-width</strong></td>
<td>Specifies the column width</td>
</tr>
<tr>
<td><strong>columns</strong></td>
<td>A shorthand property for column-width and column-count</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>content</td>
<td>Used with the :before and :after pseudo-elements, to insert generated content</td>
</tr>
<tr>
<td>counter-increment</td>
<td>Increases or decreases the value of one or more CSS counters</td>
</tr>
<tr>
<td>counter-reset</td>
<td>Creates or resets one or more CSS counters</td>
</tr>
<tr>
<td>cursor</td>
<td>Specifies the mouse cursor to be displayed when pointing over an element</td>
</tr>
<tr>
<td>direction</td>
<td>Specifies the text direction/writing direction</td>
</tr>
<tr>
<td>display</td>
<td>Specifies how a certain HTML element should be displayed</td>
</tr>
<tr>
<td>empty-cells</td>
<td>Specifies whether or not to display borders and background on empty cells in a table</td>
</tr>
<tr>
<td>filter</td>
<td>Defines effects (e.g. blurring or color shifting) on an element before the element is displayed</td>
</tr>
<tr>
<td>flex</td>
<td>A shorthand property for the flex-grow, flex-shrink, and the flex-basis properties</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>flex-basis</td>
<td>Specifies the initial length of a flexible item</td>
</tr>
<tr>
<td>flex-direction</td>
<td>Specifies the direction of the flexible items</td>
</tr>
<tr>
<td>flex-flow</td>
<td>A shorthand property for the flex-direction and the flex-wrap properties</td>
</tr>
<tr>
<td>flex-grow</td>
<td>Specifies how much the item will grow relative to the rest</td>
</tr>
<tr>
<td>flex-shrink</td>
<td>Specifies how the item will shrink relative to the rest</td>
</tr>
<tr>
<td>flex-wrap</td>
<td>Specifies whether the flexible items should wrap or not</td>
</tr>
<tr>
<td>float</td>
<td>Specifies whether or not a box should float</td>
</tr>
<tr>
<td>font</td>
<td>A shorthand property for the font-style, font-variant, font-weight, font-size/line-height, and the font-family properties</td>
</tr>
<tr>
<td>@font-face</td>
<td>A rule that allows websites to download and use fonts other than the &quot;web-safe&quot; fonts</td>
</tr>
<tr>
<td>font-family</td>
<td>Specifies the font family for text</td>
</tr>
<tr>
<td>font-feature-settings</td>
<td>Allows control over advanced typographic features in OpenType fonts</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>@font-feature-values</code></td>
<td>Allows authors to use a common name in font-variant-alternate for feature activated differently in OpenType</td>
</tr>
<tr>
<td><code>font-kerning</code></td>
<td>Controls the usage of the kerning information (how letters are spaced)</td>
</tr>
<tr>
<td><code>font-language-override</code></td>
<td>Controls the usage of language-specific glyphs in a typeface</td>
</tr>
<tr>
<td><code>font-size</code></td>
<td>Specifies the font size of text</td>
</tr>
<tr>
<td><code>font-size-adjust</code></td>
<td>Preserves the readability of text when font fallback occurs</td>
</tr>
<tr>
<td><code>font-stretch</code></td>
<td>Selects a normal, condensed, or expanded face from a font family</td>
</tr>
<tr>
<td><code>font-style</code></td>
<td>Specifies the font style for text</td>
</tr>
<tr>
<td><code>font-synthesis</code></td>
<td>Controls which missing typefaces (bold or italic) may be synthesized by the browser</td>
</tr>
<tr>
<td><code>font-variant</code></td>
<td>Specifies whether or not a text should be displayed in a small-caps font</td>
</tr>
<tr>
<td><code>font-variant-alternates</code></td>
<td>Controls the usage of alternate glyphs associated to alternative names defined in @font-feature-values</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>font-variant-caps</code></td>
<td>Controls the usage of alternate glyphs for capital letters</td>
</tr>
<tr>
<td><code>font-variant-east-asian</code></td>
<td>Controls the usage of alternate glyphs for East Asian scripts (e.g. Japanese and Chinese)</td>
</tr>
<tr>
<td><code>font-variant-ligatures</code></td>
<td>Controls which ligatures and contextual forms are used in textual content of the elements it applies to</td>
</tr>
<tr>
<td><code>font-variant-numeric</code></td>
<td>Controls the usage of alternate glyphs for numbers, fractions, and ordinal markers</td>
</tr>
<tr>
<td><code>font-variant-position</code></td>
<td>Controls the usage of alternate glyphs of smaller size positioned as superscript or subscript regarding the baseline of the font</td>
</tr>
<tr>
<td><code>font-weight</code></td>
<td>Specifies the weight of a font</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>grid</code></td>
<td>A shorthand property for the <code>grid-template-rows</code>, <code>grid-template-columns</code>, <code>grid-template-areas</code>, <code>grid-auto-rows</code>, <code>grid-auto-columns</code>, and the <code>grid-auto-flow</code> properties</td>
</tr>
<tr>
<td><code>grid-area</code></td>
<td>Either specifies a name for the grid item, or this property is a shorthand property for the <code>grid-row-start</code>, <code>grid-column-start</code>, <code>grid-row-end</code>, and <code>grid-</code></td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>grid-auto-columns</code></td>
<td>Specifies a default column size</td>
</tr>
<tr>
<td><code>grid-auto-flow</code></td>
<td>Specifies how auto-placed items are inserted in the grid</td>
</tr>
<tr>
<td><code>grid-auto-rows</code></td>
<td>Specifies a default row size</td>
</tr>
<tr>
<td><code>grid-column</code></td>
<td>A shorthand property for the <code>grid-column-start</code> and the <code>grid-column-end</code> properties</td>
</tr>
<tr>
<td><code>grid-column-end</code></td>
<td>Specifies where to end the grid item</td>
</tr>
<tr>
<td><code>grid-column-gap</code></td>
<td>Specifies the size of the gap between columns</td>
</tr>
<tr>
<td><code>grid-column-start</code></td>
<td>Specifies where to start the grid item</td>
</tr>
<tr>
<td><code>grid-gap</code></td>
<td>A shorthand property for the <code>grid-row-gap</code> and <code>grid-column-gap</code> properties</td>
</tr>
<tr>
<td><code>grid-row</code></td>
<td>A shorthand property for the <code>grid-row-start</code> and the <code>grid-row-end</code> properties</td>
</tr>
<tr>
<td><code>grid-row-end</code></td>
<td>Specifies where to end the grid item</td>
</tr>
<tr>
<td><code>grid-row-gap</code></td>
<td>Specifies the size of the gap between rows</td>
</tr>
<tr>
<td><code>grid-row-start</code></td>
<td>Specifies where to start the grid item</td>
</tr>
<tr>
<td><strong>property</strong></td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>grid-template</strong></td>
<td>A shorthand property for the <code>grid-template-rows</code>, <code>grid-template-columns</code> and <code>grid-areas</code> properties</td>
</tr>
<tr>
<td><strong>grid-template-areas</strong></td>
<td>Specifies how to display columns and rows, using named grid items</td>
</tr>
<tr>
<td><strong>grid-template-columns</strong></td>
<td>Specifies the size of the columns, and how many columns in a grid layout</td>
</tr>
<tr>
<td><strong>grid-template-rows</strong></td>
<td>Specifies the size of the rows in a grid layout</td>
</tr>
<tr>
<td><strong>hanging-punctuation</strong></td>
<td>Specifies whether a punctuation character may be placed outside the line box</td>
</tr>
<tr>
<td><strong>height</strong></td>
<td>Sets the height of an element</td>
</tr>
<tr>
<td><strong>hyphens</strong></td>
<td>Sets how to split words to improve the layout of paragraphs</td>
</tr>
<tr>
<td><strong>image-rendering</strong></td>
<td>Gives a hint to the browser about what aspects of an image are most important to preserve when the image is scaled</td>
</tr>
<tr>
<td><strong>@import</strong></td>
<td>Allows you to import a style sheet into another style sheet</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>isolation</td>
<td>Defines whether an element must create a new stacking content</td>
</tr>
<tr>
<td>justify-content</td>
<td>Specifies the alignment between the items inside a flexible container when the items do not use all available space</td>
</tr>
<tr>
<td>@keyframes</td>
<td>Specifies the animation code</td>
</tr>
<tr>
<td>left</td>
<td>Specifies the left position of a positioned element</td>
</tr>
<tr>
<td>letter-spacing</td>
<td>Increases or decreases the space between characters in a text</td>
</tr>
<tr>
<td>line-break</td>
<td>Specifies how/if to break lines</td>
</tr>
<tr>
<td>line-height</td>
<td>Sets the line height</td>
</tr>
<tr>
<td>list-style</td>
<td>Sets all the properties for a list in one declaration</td>
</tr>
<tr>
<td>list-style-image</td>
<td>Specifies an image as the list-item marker</td>
</tr>
<tr>
<td>list-style-position</td>
<td>Specifies the position of the list-item markers (bullet points)</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>list-style-type</td>
<td>Specifies the type of list-item marker</td>
</tr>
<tr>
<td>margin</td>
<td>Sets all the margin properties in one declaration</td>
</tr>
<tr>
<td>margin-bottom</td>
<td>Sets the bottom margin of an element</td>
</tr>
<tr>
<td>margin-left</td>
<td>Sets the left margin of an element</td>
</tr>
<tr>
<td>margin-right</td>
<td>Sets the right margin of an element</td>
</tr>
<tr>
<td>margin-top</td>
<td>Sets the top margin of an element</td>
</tr>
<tr>
<td>max-height</td>
<td>Sets the maximum height of an element</td>
</tr>
<tr>
<td>max-width</td>
<td>Sets the maximum width of an element</td>
</tr>
<tr>
<td>@media</td>
<td>Sets the style rules for different media types/devices/sizes</td>
</tr>
<tr>
<td>min-height</td>
<td>Sets the minimum height of an element</td>
</tr>
<tr>
<td>min-width</td>
<td>Sets the minimum width of an element</td>
</tr>
<tr>
<td>mix-blend-mode</td>
<td>Specifies how an element's content should blend with its direct parent background</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>object-fit</td>
<td>Specifies how the contents of a replaced element should be fitted to the box established by its used height and width</td>
</tr>
<tr>
<td>object-position</td>
<td>Specifies the alignment of the replaced element inside its box</td>
</tr>
<tr>
<td>opacity</td>
<td>Sets the opacity level for an element</td>
</tr>
<tr>
<td>order</td>
<td>Sets the order of the flexible item, relative to the rest</td>
</tr>
<tr>
<td>orphans</td>
<td>Sets the minimum number of lines that must be left at the bottom of a page when a page break occurs inside an element</td>
</tr>
<tr>
<td>outline</td>
<td>A shorthand property for the outline-width, outline-style, and the outline-color properties</td>
</tr>
<tr>
<td>outline-color</td>
<td>Sets the color of an outline</td>
</tr>
<tr>
<td>outline-offset</td>
<td>Offsets an outline, and draws it beyond the border edge</td>
</tr>
<tr>
<td>outline-style</td>
<td>Sets the style of an outline</td>
</tr>
<tr>
<td>outline-width</td>
<td>Sets the width of an outline</td>
</tr>
<tr>
<td>overflow</td>
<td>Specifies what happens if content overflows an element's box</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>overflow-wrap</td>
<td>Specifies whether or not the browser may break lines within words in order to prevent overflow (when a string is too long to fit its containing box)</td>
</tr>
<tr>
<td>overflow-x</td>
<td>Specifies whether or not to clip the left/right edges of the content, if it overflows the element's content area</td>
</tr>
<tr>
<td>overflow-y</td>
<td>Specifies whether or not to clip the top/bottom edges of the content, if it overflows the element's content area</td>
</tr>
<tr>
<td>padding</td>
<td>A shorthand property for all the padding-* properties</td>
</tr>
<tr>
<td>padding-bottom</td>
<td>Sets the bottom padding of an element</td>
</tr>
<tr>
<td>padding-left</td>
<td>Sets the left padding of an element</td>
</tr>
<tr>
<td>padding-right</td>
<td>Sets the right padding of an element</td>
</tr>
<tr>
<td>padding-top</td>
<td>Sets the top padding of an element</td>
</tr>
<tr>
<td>page-break-after</td>
<td>Sets the page-break behavior after an element</td>
</tr>
<tr>
<td>page-break-before</td>
<td>Sets the page-break behavior before an element</td>
</tr>
<tr>
<td>page-break-inside</td>
<td>Sets the page-break behavior inside an element</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>perspective</td>
<td>Gives a 3D-positioned element some perspective</td>
</tr>
<tr>
<td>perspective-origin</td>
<td>Defines at which position the user is looking at the 3D-positioned element</td>
</tr>
<tr>
<td>pointer-events</td>
<td>Defines whether or not an element reacts to pointer events</td>
</tr>
<tr>
<td>position</td>
<td>Specifies the type of positioning method used for an element (static, relative, absolute or fixed)</td>
</tr>
<tr>
<td>quotes</td>
<td>Sets the type of quotation marks for embedded quotations</td>
</tr>
<tr>
<td>resize</td>
<td>Defines if (and how) an element is resizable by the user</td>
</tr>
<tr>
<td>right</td>
<td>Specifies the right position of a positioned element</td>
</tr>
<tr>
<td>scroll-behavior</td>
<td>Specifies whether to smoothly animate the scroll position in a scrollable box, instead of a straight jump</td>
</tr>
<tr>
<td>tab-size</td>
<td>Specifies the width of a tab character</td>
</tr>
<tr>
<td><strong>Property</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>table-layout</code></td>
<td>Defines the algorithm used to lay out table cells, rows, and columns</td>
</tr>
<tr>
<td><code>text-align</code></td>
<td>Specifies the horizontal alignment of text</td>
</tr>
<tr>
<td><code>text-align-last</code></td>
<td>Describes how the last line of a block or a line right before a forced line break is aligned when text-align is &quot;justify&quot;</td>
</tr>
<tr>
<td><code>text-combine-upright</code></td>
<td>Specifies the combination of multiple characters into the space of a single character</td>
</tr>
<tr>
<td><code>text-decoration</code></td>
<td>Specifies the decoration added to text</td>
</tr>
<tr>
<td><code>text-decoration-color</code></td>
<td>Specifies the color of the text-decoration</td>
</tr>
<tr>
<td><code>text-decoration-line</code></td>
<td>Specifies the type of line in a text-decoration</td>
</tr>
<tr>
<td><code>text-decoration-style</code></td>
<td>Specifies the style of the line in a text decoration</td>
</tr>
<tr>
<td><code>text-indent</code></td>
<td>Specifies the indentation of the first line in a text-block</td>
</tr>
<tr>
<td><code>text-justify</code></td>
<td>Specifies the justification method used when text-align is &quot;justify&quot;</td>
</tr>
<tr>
<td><code>text-orientation</code></td>
<td>Defines the orientation of the text in a line</td>
</tr>
<tr>
<td><code>text-overflow</code></td>
<td>Specifies what should happen when text overflows the</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>text-shadow</code></td>
<td>Adds shadow to text</td>
</tr>
<tr>
<td><code>text-transform</code></td>
<td>Controls the capitalization of text</td>
</tr>
<tr>
<td><code>text-underline-position</code></td>
<td>Specifies the position of the underline which is set using the text-decoration property</td>
</tr>
<tr>
<td><code>top</code></td>
<td>Specifies the top position of a positioned element</td>
</tr>
<tr>
<td><code>transform</code></td>
<td>Applies a 2D or 3D transformation to an element</td>
</tr>
<tr>
<td><code>transform-origin</code></td>
<td>Allows you to change the position on transformed elements</td>
</tr>
<tr>
<td><code>transform-style</code></td>
<td>Specifies how nested elements are rendered in 3D space</td>
</tr>
<tr>
<td><code>transition</code></td>
<td>A shorthand property for all the <code>transition-*</code> properties</td>
</tr>
<tr>
<td><code>transition-delay</code></td>
<td>Specifies when the transition effect will start</td>
</tr>
<tr>
<td><code>transition-duration</code></td>
<td>Specifies how many seconds or milliseconds a transition effect takes to complete</td>
</tr>
<tr>
<td><code>transition-property</code></td>
<td>Specifies the name of the CSS property the transition effect is for</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>transition-timing-function</td>
<td>Specifies the speed curve of the transition effect</td>
</tr>
<tr>
<td>unicode-bidi</td>
<td>Used together with the direction property to set or return whether the text should be overridden to support multiple languages in the same document</td>
</tr>
<tr>
<td>user-select</td>
<td>Specifies whether the text of an element can be selected</td>
</tr>
<tr>
<td>vertical-align</td>
<td>Sets the vertical alignment of an element</td>
</tr>
<tr>
<td>visibility</td>
<td>Specifies whether or not an element is visible</td>
</tr>
<tr>
<td>white-space</td>
<td>Specifies how white-space inside an element is handled</td>
</tr>
<tr>
<td>widows</td>
<td>Sets the minimum number of lines that must be left at the top of a page when a page break occurs inside an element</td>
</tr>
<tr>
<td>width</td>
<td>Sets the width of an element</td>
</tr>
<tr>
<td>word-break</td>
<td>Specifies how words should break when reaching the end of a line</td>
</tr>
<tr>
<td><strong>word-spacing</strong></td>
<td>Increases or decreases the space between words in a text</td>
</tr>
<tr>
<td>------------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td><strong>word-wrap</strong></td>
<td>Allows long, unbreakable words to be broken and wrap to the next line</td>
</tr>
<tr>
<td><strong>writing-mode</strong></td>
<td>Specifies whether lines of text are laid out horizontally or vertically</td>
</tr>
<tr>
<td><strong>z-index</strong></td>
<td>Sets the stack order of a positioned element</td>
</tr>
</tbody>
</table>

### JavaScript Reference

<table>
<thead>
<tr>
<th><strong>Property/Method</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>abs()</td>
<td>Returns the absolute value of $x$</td>
</tr>
<tr>
<td>accessKey</td>
<td>Sets or returns the accesskey attribute of an element</td>
</tr>
<tr>
<td>acos()</td>
<td>Returns the arccosine of $x$, in radians</td>
</tr>
<tr>
<td>acosh()</td>
<td>Returns the hyperbolic arccosine of $x$</td>
</tr>
<tr>
<td>activeElement</td>
<td>Returns the currently focused element in the</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>document</code></td>
<td></td>
</tr>
<tr>
<td><code>addEventListener()</code></td>
<td>Attaches an event handler to the document</td>
</tr>
<tr>
<td><code>adoptNode()</code></td>
<td>Adopts a node from another document</td>
</tr>
<tr>
<td><code>alert()</code></td>
<td>Displays an alert box with a message and an OK button</td>
</tr>
<tr>
<td><code>altKey</code></td>
<td>Returns whether the &quot;ALT&quot; key was pressed when the mouse event was triggered</td>
</tr>
<tr>
<td><code>anchors</code></td>
<td>Returns a collection of all <code>&lt;a&gt;</code> elements in the document that have a name attribute</td>
</tr>
<tr>
<td><code>animationName</code></td>
<td>Returns the name of the animation</td>
</tr>
<tr>
<td><code>appCodeName</code></td>
<td>Returns the code name of the browser</td>
</tr>
<tr>
<td><code>appendChild()</code></td>
<td>Adds a new child node, to an element, as the last child node</td>
</tr>
<tr>
<td><code>applets</code></td>
<td>Returns a collection of all <code>&lt;applet&gt;</code> elements in the document</td>
</tr>
<tr>
<td><code>appName</code></td>
<td>Returns the name of the browser</td>
</tr>
<tr>
<td><code>appVersion</code></td>
<td>Returns the version information of the browser</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>asin()</td>
<td>Returns the arcsine of x, in radians</td>
</tr>
<tr>
<td>asinh()</td>
<td>Returns the hyperbolic arcsine of x</td>
</tr>
<tr>
<td>assert()</td>
<td>Writes an error message to the console if the assertion is false</td>
</tr>
<tr>
<td>assign()</td>
<td>Loads a new document</td>
</tr>
<tr>
<td>atan()</td>
<td>Returns the arctangent of x as a numeric value between $-\pi/2$ and $\pi/2$ radians</td>
</tr>
<tr>
<td>atan2()</td>
<td>Returns the arctangent of the quotient of its arguments</td>
</tr>
<tr>
<td>atanh()</td>
<td>Returns the hyperbolic arctangent of x</td>
</tr>
<tr>
<td>atob()</td>
<td>Decodes a base-64 encoded string</td>
</tr>
<tr>
<td>attributes</td>
<td>Returns a NamedNodeMap of an element's attributes</td>
</tr>
<tr>
<td>availHeight</td>
<td>Returns the height of the screen (excluding the Windows Taskbar)</td>
</tr>
<tr>
<td>availWidth</td>
<td>Returns the width of the screen (excluding the Windows Taskbar)</td>
</tr>
<tr>
<td>back()</td>
<td>Loads the previous URL in the history list</td>
</tr>
</tbody>
</table>
DevOps isn't any single person's job. It's everyone's job.

Christophe Capel

Ideate, define, and describe features and capabilities of the application

Plan

Workflow

Collaboration

Application

Security and Compliance

Continuous improvement

Develop

Operate

Deliver

Write, test, review and integrate code

Maintain, monitor and troubleshoot application in production environment

Deploy application into production environment in a consistent and reliable way

Write, test, review and integrate code

Maintain, monitor and troubleshoot application in production environment

Deploy application into production environment in a consistent and reliable way

Write, test, review and integrate code

Maintain, monitor and troubleshoot application in production environment

Deploy application into production environment in a consistent and reliable way

Write, test, review and integrate code
Development (Software engineering) + Quality assurance + operations + Security = DevOps

DevSecOps

- Communication
- Collaboration
- Integration

Dev

(Software releases and updates)

Ops

(Reliability, performance and scaling)

Sec

(Confidentiality, availability and integrity)

Agile

Focuses on processes highlighting change while accelerating delivery

CI/CD

Focuses on software-defined life cycles highlighting tools that emphasize automation

DevOps

Focuses on culture highlighting roles that emphasize responsiveness
**DevOps Flow**

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Git</strong></td>
<td>Version Control System tool</td>
</tr>
<tr>
<td><strong>Jenkins</strong></td>
<td>Continuous Integration tool</td>
</tr>
<tr>
<td><strong>Selenium</strong></td>
<td>Continuous Testing tool</td>
</tr>
<tr>
<td><strong>Puppet, Chef, Ansible</strong></td>
<td>Configuration Management and Deployment tools</td>
</tr>
<tr>
<td><strong>Nagios</strong></td>
<td>Continuous Monitoring tool</td>
</tr>
<tr>
<td><strong>Docker</strong></td>
<td>Containerization tool</td>
</tr>
</tbody>
</table>
DevOps

Agile
Development start  Development end  Release  Production support
Development  Operations  Testing

Cloud Service:
On-premise

You manage
Applications
  Data
  Runtime
  Middleware
  OS
  Virtualization
  Servers
  Storage
  Networking

You manage

Platform as a service

You manage
Applications
  Data
  Runtime
  Middleware
  OS
  Virtualization
  Servers
  Storage
  Networking

Others manage
Applications
  Data
  Runtime
  Middleware
  OS
  Virtualization
  Servers
  Storage
  Networking

Others manage

Software as a service

You manage
Applications
  Data
  Runtime
  Middleware
  OS
  Virtualization
  Servers
  Storage
  Networking

Others manage
Applications
  Data
  Runtime
  Middleware
  OS
  Virtualization
  Servers
  Storage
  Networking
Stages of a typical DevOps delivery pipeline

1. Understand your users and frame a hypothesis
2. Craft an experiment
3. Test your hypothesis with users
4. Learn, adopt and correct
# Basic Hello World program written in Groovy

class MyClass {
    static void main(String[] args) {
        println('Hello World');
    }
}

---

6 Build Phases in Maven:

Validate → Compile → Test → Package → Install → Deploy

Command:

ifstat

Description:

Prints network interface statistics
Processes

Parent processes
Processes that create other processes during runtime

Child processes
Processes that are created by other processes during run-time

Command:

pidof bash

Description:
Display the process IDs of a specific running program (Bash)

# Display the process ID of the current shell

echo $$

# Display the parent process ID of the current shell

echo $PPID
### Command:

```
ps aux | awk '{print $6/1024 " MBtt" $11}r | sort -n
```

### Description:

Display a list of most memory consuming processes

<table>
<thead>
<tr>
<th>Github Fetch</th>
<th>Github Pull</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fetches the required information only to local repository</td>
<td>Fetches the required information not only to local repository but also to the workspace that you are currently working in</td>
</tr>
</tbody>
</table>

\[
\text{Github Pull} = \text{Github fetch} + \text{Merge}
\]

- **Github Push**: Combination of fetch and merge the content
- **Github Pull**: Fetch the content

<table>
<thead>
<tr>
<th>Docker registry</th>
<th>Docker repository</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service for hosting and distributing docker images</td>
<td>Collection of related Docker images</td>
</tr>
</tbody>
</table>
Selenium supports 2 types of testing:

- **Regression Testing** → retesting a product around an area where a bug was fixed.
- **Functional Testing** → testing of software features (functional points) individually.

**Command:**

```
ps aux | awk '{print $6/1024 " MBtt" $11}' | sort -n
```

**Description:**
Display a list of most memory consuming processes

**Command:**

```
ps aux
```

**Description:**
Display all processes and their status and resource usage

---

**Diagram:**

```
Docker file ➔ Docker image ➔ Docker hub
```

**Containers:**

- Container
- Container

**Docker container**
<table>
<thead>
<tr>
<th>baseURI</th>
<th>Returns the absolute base URI of a document</th>
</tr>
</thead>
<tbody>
<tr>
<td>blur()</td>
<td>Removes focus from an element</td>
</tr>
<tr>
<td>body</td>
<td>Sets or returns the document's body (the &lt;body&gt; element)</td>
</tr>
<tr>
<td>break</td>
<td>Exits a switch or a loop</td>
</tr>
<tr>
<td>btoa()</td>
<td>Encodes a string in base-64</td>
</tr>
<tr>
<td>bubbles</td>
<td>Returns whether or not a specific event is a bubbling event</td>
</tr>
<tr>
<td>button</td>
<td>Returns which mouse button was pressed when the mouse event was triggered</td>
</tr>
<tr>
<td>buttons</td>
<td>Returns which mouse buttons were pressed when the mouse event was triggered</td>
</tr>
<tr>
<td>cancelable</td>
<td>Returns whether or not an event can have its default action prevented</td>
</tr>
<tr>
<td>cbrt()</td>
<td>Returns the cubic root of x</td>
</tr>
<tr>
<td>cell()</td>
<td>Returns x, rounded upwards to the nearest integer</td>
</tr>
<tr>
<td>changeTouches</td>
<td>Returns a list of all the touch objects whose state changed between the previous touch and</td>
</tr>
<tr>
<td><strong>characterSet</strong></td>
<td>Returns the character encoding for the document</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td><strong>charAt()</strong></td>
<td>Returns the character at the specified index (position)</td>
</tr>
<tr>
<td><strong>charCodeAt()</strong></td>
<td>Returns the Unicode character code of the key that triggered the onkeypress event</td>
</tr>
<tr>
<td><strong>charCodeAt()</strong></td>
<td>Returns the Unicode of the character at the specified index</td>
</tr>
<tr>
<td><strong>charset</strong></td>
<td>Deprecated. Use characterSet instead. Returns the character encoding for the document</td>
</tr>
<tr>
<td><strong>childElementCount</strong></td>
<td>Returns the number of child elements an element has</td>
</tr>
<tr>
<td><strong>childNodes</strong></td>
<td>Returns a collection of an element's child nodes (including text and comment nodes)</td>
</tr>
<tr>
<td><strong>children</strong></td>
<td>Returns a collection of an element's child element (excluding text and comment nodes)</td>
</tr>
<tr>
<td><strong>classList</strong></td>
<td>Returns the class name(s) of an element</td>
</tr>
<tr>
<td><strong>class</strong></td>
<td>Declares a class</td>
</tr>
<tr>
<td>className</td>
<td>Sets or returns the value of the class attribute of an element</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>clear()</td>
<td>Clears the console</td>
</tr>
<tr>
<td>clearInterval()</td>
<td>Clears a timer set with setInterval()</td>
</tr>
<tr>
<td>clearTimeout()</td>
<td>Clears a timer set with setTimeout()</td>
</tr>
<tr>
<td>clearWatch()</td>
<td>Unregister location/error monitoring handlers previously installed using Geolocation.watchPosition()</td>
</tr>
<tr>
<td>click()</td>
<td>Simulates a mouse-click on an element</td>
</tr>
<tr>
<td>clientHeight</td>
<td>Returns the height of an element, including padding</td>
</tr>
<tr>
<td>clientLeft</td>
<td>Returns the width of the left border of an element</td>
</tr>
<tr>
<td>clientTop</td>
<td>Returns the width of the top border of an element</td>
</tr>
<tr>
<td>clientWidth</td>
<td>Returns the width of an element, including padding</td>
</tr>
<tr>
<td>.clientX</td>
<td>Returns the horizontal coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered</td>
</tr>
<tr>
<td><strong>clientY</strong></td>
<td>Returns the vertical coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered</td>
</tr>
<tr>
<td><strong>clipboardData</strong></td>
<td>Returns an object containing the data affected by the clipboard operation</td>
</tr>
<tr>
<td><strong>closed</strong></td>
<td>Returns a Boolean value indicating whether a window has been closed or not</td>
</tr>
<tr>
<td><strong>close()</strong></td>
<td>Closes the output stream previously opened with <code>document.open()</code></td>
</tr>
<tr>
<td><strong>code</strong></td>
<td>Returns the code of the key that triggered the event</td>
</tr>
<tr>
<td><strong>colorDepth</strong></td>
<td>Returns the bit depth of the color palette for displaying images</td>
</tr>
<tr>
<td><strong>compareDocumentPosition()</strong></td>
<td>Compares the document position of two elements</td>
</tr>
<tr>
<td><strong>compile()</strong></td>
<td>Deprecated in version 1.5. Compiles a regular expression</td>
</tr>
<tr>
<td><strong>composed</strong></td>
<td>Returns whether the event is composed or not</td>
</tr>
<tr>
<td><strong>concat()</strong></td>
<td>Joins two or more arrays, and returns a copy of the joined arrays</td>
</tr>
<tr>
<td><strong>confirm()</strong></td>
<td>Displays a dialog box with a message and an OK button</td>
</tr>
<tr>
<td>const</td>
<td>Declares a variable with a constant value</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>constructor()</td>
<td>Creates and initialize objects created within a class</td>
</tr>
<tr>
<td>constructor</td>
<td>Returns the function that created the Array object's prototype</td>
</tr>
<tr>
<td>contains()</td>
<td>Returns true if a node is a descendant of a node, otherwise false</td>
</tr>
<tr>
<td>contentEditable</td>
<td>Sets or returns whether the content of an element is editable or not</td>
</tr>
<tr>
<td>continue</td>
<td>Breaks one iteration (in the loop) if a specified condition occurs, and continues with the next iteration in the loop</td>
</tr>
<tr>
<td>console</td>
<td>Returns a reference to the Console object, which provides methods for logging information to the browser's console (See Console object)</td>
</tr>
<tr>
<td>cookie</td>
<td>Returns all name/value pairs of cookies in the document</td>
</tr>
<tr>
<td>cookieEnabled</td>
<td>Determines whether cookies are enabled in the browser</td>
</tr>
<tr>
<td>coordinates</td>
<td>Returns the position and altitude of the device on Earth</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>copyWithin()</td>
<td>Copies array elements within the array, to and from specified positions</td>
</tr>
<tr>
<td>cos(x)</td>
<td>Returns the cosine of x (x is in radians)</td>
</tr>
<tr>
<td>cosh(x)</td>
<td>Returns the hyperbolic cosine of x</td>
</tr>
<tr>
<td>count()</td>
<td>Logs the number of times that this particular call to count() has been called</td>
</tr>
<tr>
<td>createAttribute()</td>
<td>Creates an attribute node</td>
</tr>
<tr>
<td>createComment()</td>
<td>Creates a Comment node with the specified text</td>
</tr>
<tr>
<td>createDocumentFragment()</td>
<td>Creates an empty DocumentFragment node</td>
</tr>
<tr>
<td>createElement()</td>
<td>Creates an Element node</td>
</tr>
<tr>
<td>createEvent()</td>
<td>Creates a new event</td>
</tr>
<tr>
<td>createTextNode()</td>
<td>Creates a Text node</td>
</tr>
<tr>
<td>ctrlKey</td>
<td>Returns whether the &quot;CTRL&quot; key was pressed when the mouse event was triggered</td>
</tr>
<tr>
<td>currentTarget</td>
<td>Returns the element whose event listeners triggered the event</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>data</td>
<td>Returns the inserted characters</td>
</tr>
<tr>
<td>dataTransfer</td>
<td>Returns an object containing the data being dragged/dropped, or inserted/deleted</td>
</tr>
<tr>
<td>debugger</td>
<td>Stops the execution of JavaScript, and calls (if available) the debugging function</td>
</tr>
<tr>
<td>decodeURI()</td>
<td>Decodes a URI</td>
</tr>
<tr>
<td>decodeURIComponent()</td>
<td>Decodes a URI component</td>
</tr>
<tr>
<td>defaultPrevented</td>
<td>Returns whether or not the preventDefault() method was called for the event</td>
</tr>
<tr>
<td>defaultStatus</td>
<td>Sets or returns the default text in the statusbar of a window</td>
</tr>
<tr>
<td>defaultView</td>
<td>Returns the window object associated with a document, or null if none is available.</td>
</tr>
<tr>
<td>delete</td>
<td>Deletes a property from an object</td>
</tr>
<tr>
<td>deltaX</td>
<td>Returns the horizontal scroll amount of a mouse wheel (x-axis)</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>deltaY</td>
<td>Returns the vertical scroll amount of a mouse wheel (y-axis)</td>
</tr>
<tr>
<td>deltaZ</td>
<td>Returns the scroll amount of a mouse wheel for the z-axis</td>
</tr>
<tr>
<td>deltaMode</td>
<td>Returns a number that represents the unit of measurements for delta values (pixels, lines or pages)</td>
</tr>
<tr>
<td>designMode</td>
<td>Controls whether the entire document should be editable or not.</td>
</tr>
<tr>
<td>detail</td>
<td>Returns a number that indicates how many times the mouse was clicked</td>
</tr>
<tr>
<td>do ... while</td>
<td>Executes a block of statements and repeats the block while a condition is true</td>
</tr>
<tr>
<td>doctype</td>
<td>Returns the Document Type Declaration associated with the document</td>
</tr>
<tr>
<td>document</td>
<td>Returns the Document object for the window (See Document object)</td>
</tr>
<tr>
<td>documentElement</td>
<td>Returns the Document Element of the document (the &lt;html&gt; element)</td>
</tr>
<tr>
<td>documentMode</td>
<td>Returns the mode used by the browser to render the document</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>documentURI</td>
<td>Sets or returns the location of the document</td>
</tr>
<tr>
<td>domain</td>
<td>Returns the domain name of the server that loaded the document</td>
</tr>
<tr>
<td>domConfig</td>
<td>Obsolete. Returns the DOM configuration of the document</td>
</tr>
<tr>
<td>elapsedTime</td>
<td>Returns the number of seconds an animation has been running</td>
</tr>
<tr>
<td>elapsedTime</td>
<td>Returns the number of seconds a transition has been running</td>
</tr>
<tr>
<td>embeds</td>
<td>Returns a collection of all <code>&lt;embed&gt;</code> elements in the document</td>
</tr>
<tr>
<td>encodeURI()</td>
<td>Encodes a URI</td>
</tr>
<tr>
<td>encodeURICOMPONENT()</td>
<td>Encodes a URI component</td>
</tr>
<tr>
<td>E</td>
<td>Returns Euler's number (approx. 2.718)</td>
</tr>
<tr>
<td>endsWith()</td>
<td>Checks whether a string ends with specified string/characters</td>
</tr>
<tr>
<td>entries()</td>
<td>Returns a key/value pair Array Iteration Object</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>error()</td>
<td>Outputs an error message to the console</td>
</tr>
<tr>
<td>escape()</td>
<td>Deprecated in version 1.5. Use <code>encodeURI()</code> or <code>encodeURIComponent()</code> instead</td>
</tr>
<tr>
<td>eval()</td>
<td>Evaluates a string and executes it as if it was script code</td>
</tr>
<tr>
<td>eventPhase</td>
<td>Returns which phase of the event flow is currently being evaluated</td>
</tr>
<tr>
<td>every()</td>
<td>Checks if every element in an array pass a test</td>
</tr>
<tr>
<td>exec()</td>
<td>Tests for a match in a string. Returns the first match</td>
</tr>
<tr>
<td>execCommand()</td>
<td>Invokes the specified clipboard operation on the element currently having focus.</td>
</tr>
<tr>
<td>exitFullscreen()</td>
<td>Cancels an element in fullscreen mode</td>
</tr>
<tr>
<td>exp(x)</td>
<td>Returns the value of $E^x$</td>
</tr>
<tr>
<td>export</td>
<td>Export functions so they can be used for imports in external modules, and other scripts</td>
</tr>
<tr>
<td>extends</td>
<td>Extends a class (inherit)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>dir</td>
<td>Sets or returns the value of the dir attribute of an element</td>
</tr>
<tr>
<td>fill()</td>
<td>Fill the elements in an array with a static value</td>
</tr>
<tr>
<td>filter()</td>
<td>Creates a new array with every element in an array that pass a test</td>
</tr>
<tr>
<td>find()</td>
<td>Returns the value of the first element in an array that pass a test</td>
</tr>
<tr>
<td>findIndex()</td>
<td>Returns the index of the first element in an array that pass a test</td>
</tr>
<tr>
<td>floor()</td>
<td>Returns x, rounded downwards to the nearest integer</td>
</tr>
<tr>
<td>focus()</td>
<td>Gives focus to an element</td>
</tr>
<tr>
<td>for</td>
<td>Marks a block of statements to be executed as long as a condition is true</td>
</tr>
<tr>
<td>for ... in</td>
<td>Marks a block of statements to be executed for each element of an object (or array)</td>
</tr>
<tr>
<td>forEach()</td>
<td>Calls a function for each array element</td>
</tr>
<tr>
<td>forms</td>
<td>Returns a collection of all &lt;form&gt; elements in the document</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------</td>
</tr>
<tr>
<td>forward()</td>
<td>Loads the next URL in the history list</td>
</tr>
<tr>
<td>frameElement</td>
<td>Returns the <code>&lt;iframe&gt;</code> element in which the current window is inserted</td>
</tr>
<tr>
<td>frames</td>
<td>Returns all <code>&lt;iframe&gt;</code> elements in the current window</td>
</tr>
<tr>
<td>from()</td>
<td>Creates an array from an object</td>
</tr>
<tr>
<td>fromCharCode()</td>
<td>Converts Unicode values to characters</td>
</tr>
<tr>
<td>fullscreenElement</td>
<td>Returns the current element that is displayed in fullscreen mode</td>
</tr>
<tr>
<td>fullscreenEnabled()</td>
<td>Returns a Boolean value indicating whether the document can be viewed in fullscreen mode</td>
</tr>
<tr>
<td>function</td>
<td>Declares a function</td>
</tr>
<tr>
<td>geolocation</td>
<td>Returns a Geolocation object that can be used to locate the user's position</td>
</tr>
<tr>
<td>getDate()</td>
<td>Returns the day of the month (from 1-31)</td>
</tr>
<tr>
<td>getDay()</td>
<td>Returns the day of the week (from 0-6)</td>
</tr>
<tr>
<td>getAttribute()</td>
<td>Returns the specified attribute value of an element node</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>getAttributeNode()</code></td>
<td>Returns the specified attribute node</td>
</tr>
<tr>
<td><code>getBoundingClientRect()</code></td>
<td>Returns the size of an element and its position relative to the viewport</td>
</tr>
<tr>
<td><code>getComputedStyle()</code></td>
<td>Gets the current computed CSS styles applied to an element</td>
</tr>
<tr>
<td><code>getCurrentPosition()</code></td>
<td>Returns the current position of the device</td>
</tr>
<tr>
<td><code>getElementById()</code></td>
<td>Returns the element that has the ID attribute with the specified value</td>
</tr>
<tr>
<td><code>getElementsByClassName()</code></td>
<td>Returns a NodeList containing all elements with the specified class name</td>
</tr>
<tr>
<td><code>getElementsByClassName()</code></td>
<td>Returns a NodeList containing all elements with the specified class name</td>
</tr>
<tr>
<td><code>getElementsByClassName()</code></td>
<td>Returns a NodeList containing all elements with a specified name</td>
</tr>
<tr>
<td><code>getElementsByClassName()</code></td>
<td>Returns a NodeList containing all elements with the specified tag name</td>
</tr>
<tr>
<td><code>getFullYear()</code></td>
<td>Returns the year</td>
</tr>
<tr>
<td><code>getHours()</code></td>
<td>Returns the hour (from 0-23)</td>
</tr>
<tr>
<td><code>getItem()</code></td>
<td>Returns the value of the specified key name</td>
</tr>
<tr>
<td><code>getMilliseconds()</code></td>
<td>Returns the milliseconds (from 0-999)</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>getMinutes()</td>
<td>Returns the minutes (from 0-59)</td>
</tr>
<tr>
<td>getModifierState()</td>
<td>Returns an array containing target ranges that will be affected by the insertion/deletion</td>
</tr>
<tr>
<td>getMonth()</td>
<td>Returns the month (from 0-11)</td>
</tr>
<tr>
<td>getNamedItem()</td>
<td>Returns a specified attribute node from a NamedNodeMap</td>
</tr>
<tr>
<td>getSeconds()</td>
<td>Returns the seconds (from 0-59)</td>
</tr>
<tr>
<td>getSelection()</td>
<td>Returns a Selection object representing the range of text selected by the user</td>
</tr>
<tr>
<td>getTime()</td>
<td>Returns the number of milliseconds since midnight Jan 1 1970, and a specified date</td>
</tr>
<tr>
<td>getTimezoneOffset()</td>
<td>Returns the time difference between UTC time and local time, in minutes</td>
</tr>
<tr>
<td>getUTCDate()</td>
<td>Returns the day of the month, according to universal time (from 1-31)</td>
</tr>
<tr>
<td>getUTCDay()</td>
<td>Returns the day of the week, according to universal time (from 0-6)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>getUTCFullYear()</td>
<td>Returns the year, according to universal time</td>
</tr>
<tr>
<td>getUTCHours()</td>
<td>Returns the hour, according to universal time (from 0-23)</td>
</tr>
<tr>
<td>getUTCMilliseconds()</td>
<td>Returns the milliseconds, according to universal time (from 0-999)</td>
</tr>
<tr>
<td>getUTCMinutes()</td>
<td>Returns the minutes, according to universal time (from 0-59)</td>
</tr>
<tr>
<td>getUTCMonth()</td>
<td>Returns the month, according to universal time (from 0-11)</td>
</tr>
<tr>
<td>getUTCSeconds()</td>
<td>Returns the seconds, according to universal time (from 0-59)</td>
</tr>
<tr>
<td>getYear()</td>
<td>Deprecated. Use the getFullYear() method instead</td>
</tr>
<tr>
<td>global</td>
<td>Checks whether the &quot;g&quot; modifier is set</td>
</tr>
<tr>
<td>go()</td>
<td>Loads a specific URL from the history list</td>
</tr>
<tr>
<td>group()</td>
<td>Creates a new inline group in the console. This indents following console</td>
</tr>
<tr>
<td></td>
<td>messages by an additional level, until console.groupEnd() is called</td>
</tr>
<tr>
<td>groupCollapsed()</td>
<td>Creates a new inline group in the console.</td>
</tr>
</tbody>
</table>
However, the new group is created collapsed. The user will need to use the disclosure button to expand it

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>groupEnd()</td>
<td>Exits the current inline group in the console</td>
</tr>
<tr>
<td>hasAttribute()</td>
<td>Returns true if an element has the specified attribute, otherwise false</td>
</tr>
<tr>
<td>hasAttributes()</td>
<td>Returns true if an element has any attributes, otherwise false</td>
</tr>
<tr>
<td>hasChildNodes()</td>
<td>Returns true if an element has any child nodes, otherwise false</td>
</tr>
<tr>
<td>hasFocus()</td>
<td>Returns a Boolean value indicating whether the document has focus</td>
</tr>
<tr>
<td>hash</td>
<td>Sets or returns the anchor part (#) of a URL</td>
</tr>
<tr>
<td>head</td>
<td>Returns the &lt;head&gt; element of the document</td>
</tr>
<tr>
<td>height</td>
<td>Returns the total height of the screen</td>
</tr>
<tr>
<td>history</td>
<td>Returns the History object for the window (See History object)</td>
</tr>
<tr>
<td>host</td>
<td>Sets or returns the hostname and port number of a URL</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>hostname</td>
<td>Sets or returns the hostname of a URL</td>
</tr>
<tr>
<td>href</td>
<td>Sets or returns the entire URL</td>
</tr>
<tr>
<td>id</td>
<td>Sets or returns the value of the id attribute of an element</td>
</tr>
<tr>
<td>if ... else ... else if</td>
<td>Marks a block of statements to be executed depending on a condition</td>
</tr>
<tr>
<td>ignoreCase</td>
<td>Checks whether the &quot;i&quot; modifier is set</td>
</tr>
<tr>
<td>images</td>
<td>Returns a collection of all &lt;img&gt; elements in the document</td>
</tr>
<tr>
<td>implementation</td>
<td>Returns the DOMImplementation object that handles this document</td>
</tr>
<tr>
<td>import</td>
<td>Import functions exported from an external module, and another script</td>
</tr>
<tr>
<td>importNode()</td>
<td>Imports a node from another document</td>
</tr>
<tr>
<td>in</td>
<td>Returns true if the specified property is in the specified object, otherwise false</td>
</tr>
<tr>
<td>includes()</td>
<td>Check if an array contains the specified element</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>indexOf()</code></td>
<td>Search the array for an element and returns its position</td>
</tr>
<tr>
<td><strong>Infinity</strong></td>
<td>A numeric value that represents positive/negative infinity</td>
</tr>
<tr>
<td><code>info()</code></td>
<td>Outputs an informational message to the console</td>
</tr>
<tr>
<td><code>innerHeight</code></td>
<td>Returns the height of the window's content area (viewport) including scrollbars</td>
</tr>
<tr>
<td><code>innerHTML</code></td>
<td>Sets or returns the content of an element</td>
</tr>
<tr>
<td><code>innerText</code></td>
<td>Sets or returns the text content of a node and its descendants</td>
</tr>
<tr>
<td><code>innerWidth</code></td>
<td>Returns the width of a window's content area (viewport) including scrollbars</td>
</tr>
<tr>
<td><code>inputEncoding</code></td>
<td>Returns the encoding, character set, used for the document</td>
</tr>
<tr>
<td><code>inputType</code></td>
<td>Returns the type of the change (i.e. &quot;inserting&quot; or &quot;deleting&quot;)</td>
</tr>
<tr>
<td><code>insertAdjacentElement()</code></td>
<td>Inserts a HTML element at the specified position relative to the current element</td>
</tr>
<tr>
<td><code>insertAdjacentHTML()</code></td>
<td>Inserts a HTML formatted text at the specified position</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>insertAdjacentText()</td>
<td>Inserts text into the specified position relative to the current element</td>
</tr>
<tr>
<td>insertBefore()</td>
<td>Inserts a new child node before a specified, existing, child node</td>
</tr>
<tr>
<td>instanceof</td>
<td>Returns true if the specified object is an instance of the specified object</td>
</tr>
<tr>
<td>isArray()</td>
<td>Checks whether an object is an array</td>
</tr>
<tr>
<td>isComposing</td>
<td>Returns whether the state of the event is composing or not</td>
</tr>
<tr>
<td>isContentEditable</td>
<td>Returns true if the content of an element is editable, otherwise false</td>
</tr>
<tr>
<td>isDefaultNamespace()</td>
<td>Returns true if a specified namespaceURI is the default, otherwise false</td>
</tr>
<tr>
<td>isEqualNode()</td>
<td>Checks if two elements are equal</td>
</tr>
<tr>
<td>isFinite()</td>
<td>Determines whether a value is a finite, legal number</td>
</tr>
<tr>
<td>isId</td>
<td>Returns true if the attribute is of type Id, otherwise it returns false</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>isInteger()</code></td>
<td>Checks whether a value is an integer</td>
</tr>
<tr>
<td><code>isNaN()</code></td>
<td>Determines whether a value is an illegal number</td>
</tr>
<tr>
<td><code>isSafeInteger()</code></td>
<td>Checks whether a value is a safe integer</td>
</tr>
<tr>
<td><code>isSameNode()</code></td>
<td>Checks if two elements are the same node</td>
</tr>
<tr>
<td><code>isSupported()</code></td>
<td>Returns true if a specified feature is supported on the element</td>
</tr>
<tr>
<td><code>isTrusted</code></td>
<td>Returns whether or not an event is trusted</td>
</tr>
<tr>
<td><code>item()</code></td>
<td>Returns the attribute node at a specified index in a NamedNodeMap</td>
</tr>
<tr>
<td><code>join()</code></td>
<td>Joins all elements of an array into a string</td>
</tr>
<tr>
<td><code>key</code></td>
<td>Returns the key value of the key represented by the event</td>
</tr>
<tr>
<td><code>key()</code></td>
<td>Returns the name of the nth key in the storage</td>
</tr>
<tr>
<td><code>keyCode</code></td>
<td>Returns the Unicode character code of the key that triggered the onkeypress event, or the Unicode key code of the key that triggered the onkeydown or onkeyup event</td>
</tr>
<tr>
<td>keys()</td>
<td>Returns a Array Iteration Object, containing the keys of the original array</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>lang</td>
<td>Sets or returns the value of the lang attribute of an element</td>
</tr>
<tr>
<td>language</td>
<td>Returns the language of the browser</td>
</tr>
<tr>
<td>lastChild</td>
<td>Returns the last child node of an element</td>
</tr>
<tr>
<td>lastElementChild</td>
<td>Returns the last child element of an element</td>
</tr>
<tr>
<td>lastIndex</td>
<td>Specifies the index at which to start the next match</td>
</tr>
<tr>
<td>lastIndexOf()</td>
<td>Search the array for an element, starting at the end, and returns its position</td>
</tr>
<tr>
<td>lastModified</td>
<td>Returns the date and time the document was last modified</td>
</tr>
<tr>
<td>length</td>
<td>Sets or returns the number of elements in an array</td>
</tr>
<tr>
<td>lengthComputable</td>
<td>Returns whether the length of the progress can be computable or not</td>
</tr>
<tr>
<td>let</td>
<td>Declares a variable inside brackets {} scope</td>
</tr>
<tr>
<td>links</td>
<td>Returns a collection of all <code>&lt;a&gt;</code> and <code>&lt;area&gt;</code> elements in the document that have a <code>href</code> attribute</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LN2</td>
<td>Returns the natural logarithm of 2 (approx. 0.693)</td>
</tr>
<tr>
<td>LN10</td>
<td>Returns the natural logarithm of 10 (approx. 2.302)</td>
</tr>
<tr>
<td>loaded</td>
<td>Returns how much work has been loaded</td>
</tr>
<tr>
<td>localeCompare()</td>
<td>Compares two strings in the current locale</td>
</tr>
<tr>
<td>localStorage</td>
<td>Allows to save key/value pairs in a web browser. Stores the data with no expiration date</td>
</tr>
<tr>
<td>location</td>
<td>Returns the location of a key on the keyboard or device</td>
</tr>
<tr>
<td>location</td>
<td>Returns the Location object for the window (See Location object)</td>
</tr>
<tr>
<td>log()</td>
<td>Returns the natural logarithm (base E) of x</td>
</tr>
<tr>
<td>LOG2E</td>
<td>Returns the base-2 logarithm of E (approx. 1.442)</td>
</tr>
<tr>
<td>LOG10E</td>
<td>Returns the base-10 logarithm of E (approx.</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>map()</code></td>
<td>Creates a new array with the result of calling a function for each array element</td>
</tr>
<tr>
<td><code>match()</code></td>
<td>Searches a string for a match against a regular expression, and returns the matches</td>
</tr>
<tr>
<td><code>matchMedia()</code></td>
<td>Returns a MediaQueryList object representing the specified CSS media query string</td>
</tr>
<tr>
<td><code>max()</code></td>
<td>Returns the number with the highest value</td>
</tr>
<tr>
<td><code>MAX_VALUE</code></td>
<td>Returns the largest number possible in JavaScript</td>
</tr>
<tr>
<td><code>message</code></td>
<td>Sets or returns an error message (a string)</td>
</tr>
<tr>
<td><code>metaKey</code></td>
<td>Returns whether the &quot;META&quot; key was pressed when an event was triggered</td>
</tr>
<tr>
<td><code>min()</code></td>
<td>Returns the number with the lowest value</td>
</tr>
<tr>
<td><code>multiline</code></td>
<td>Checks whether the &quot;m&quot; modifier is set</td>
</tr>
<tr>
<td><code>MIN_VALUE</code></td>
<td>Returns the smallest number possible in JavaScript</td>
</tr>
<tr>
<td><code>moveBy()</code></td>
<td>Moves a window relative to its current position</td>
</tr>
<tr>
<td>function</td>
<td>description</td>
</tr>
<tr>
<td>----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>moveTo()</td>
<td>Moves a window to the specified position</td>
</tr>
<tr>
<td>MovementX</td>
<td>Returns the horizontal coordinate of the mouse pointer relative to the position of the last mousemove event</td>
</tr>
<tr>
<td>MovementY</td>
<td>Returns the vertical coordinate of the mouse pointer relative to the position of the last mousemove event</td>
</tr>
<tr>
<td>name</td>
<td>Sets or returns an error name</td>
</tr>
<tr>
<td>namedItem()</td>
<td>Returns the element with the specified ID, or name, in an HTMLCollection</td>
</tr>
<tr>
<td>namespaceURI</td>
<td>Returns the namespace URI of an element</td>
</tr>
<tr>
<td>NaN</td>
<td>&quot;Not-a-Number&quot; value</td>
</tr>
<tr>
<td>navigator</td>
<td>Returns the Navigator object for the window (See Navigator object)</td>
</tr>
<tr>
<td>NEGATIVE_INFINITY</td>
<td>Represents negative infinity (returned on overflow)</td>
</tr>
<tr>
<td>new</td>
<td>Creates an instance of a constructor</td>
</tr>
<tr>
<td>newURL</td>
<td>Returns the URL of the document, after the hash has been changed</td>
</tr>
<tr>
<td>newValue</td>
<td>Returns the new value of the changed storage item</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>nextSibling</td>
<td>Returns the next node at the same node tree level</td>
</tr>
<tr>
<td>nextElementSibling</td>
<td>Returns the next element at the same node tree level</td>
</tr>
<tr>
<td>nodeName</td>
<td>Returns the name of a node</td>
</tr>
<tr>
<td>nodeType</td>
<td>Returns the node type of a node</td>
</tr>
<tr>
<td>nodeValue</td>
<td>Sets or returns the value of a node</td>
</tr>
<tr>
<td>normalize()</td>
<td>Removes empty Text nodes, and joins adjacent nodes</td>
</tr>
<tr>
<td>normalizeDocument()</td>
<td>Removes empty Text nodes, and joins adjacent nodes</td>
</tr>
<tr>
<td>now()</td>
<td>Returns the number of milliseconds since midnight Jan 1, 1970</td>
</tr>
<tr>
<td>Number()</td>
<td>Converts an object's value to a number</td>
</tr>
<tr>
<td>offsetHeight</td>
<td>Returns the height of an element, including padding, border and scrollbar</td>
</tr>
<tr>
<td>offsetWidth</td>
<td>Returns the width of an element, including padding, border and scrollbar</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>offsetLeft</td>
<td>Returns the horizontal offset position of an element</td>
</tr>
<tr>
<td>offsetParent</td>
<td>Returns the offset container of an element</td>
</tr>
<tr>
<td>offsetTop</td>
<td>Returns the vertical offset position of an element</td>
</tr>
<tr>
<td>offsetX</td>
<td>Returns the horizontal coordinate of the mouse pointer relative to the position of the edge of the target element</td>
</tr>
<tr>
<td>offsetY</td>
<td>Returns the vertical coordinate of the mouse pointer relative to the position of the edge of the target element</td>
</tr>
<tr>
<td>oldURL</td>
<td>Returns the URL of the document, before the hash was changed</td>
</tr>
<tr>
<td>oldValue</td>
<td>Returns the old value of the changed storage item</td>
</tr>
<tr>
<td>onabort</td>
<td>The event occurs when the loading of a media is aborted</td>
</tr>
<tr>
<td>onafterprint</td>
<td>The event occurs when a page has started printing, or if the print dialogue box has</td>
</tr>
<tr>
<td>Event Name</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onanimationend</td>
<td>The event occurs when a CSS animation has completed</td>
</tr>
<tr>
<td>onanimationiteration</td>
<td>The event occurs when a CSS animation is repeated</td>
</tr>
<tr>
<td>onanimationstart</td>
<td>The event occurs when a CSS animation has started</td>
</tr>
<tr>
<td>onbeforeprint</td>
<td>The event occurs when a page is about to be printed</td>
</tr>
<tr>
<td>onbeforeunload</td>
<td>The event occurs before the document is about to be unloaded</td>
</tr>
<tr>
<td>onblur</td>
<td>The event occurs when an element loses focus</td>
</tr>
<tr>
<td>oncanplay</td>
<td>The event occurs when the browser can start playing the media (when it has buffered enough to begin)</td>
</tr>
<tr>
<td>oncanplaythrough</td>
<td>The event occurs when the browser can play through the media without stopping for buffering</td>
</tr>
<tr>
<td>onchange</td>
<td>The event occurs when the content of a form element, the selection, or the checked state have changed (for <code>&lt;input&gt;</code>, <code>&lt;select&gt;</code>, and <code>&lt;textarea&gt;</code>)</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onclick</td>
<td>The event occurs when the user clicks on an element</td>
</tr>
<tr>
<td>oncontextmenu</td>
<td>The event occurs when the user right-clicks on an element to open a context menu</td>
</tr>
<tr>
<td>oncopy</td>
<td>The event occurs when the user copies the content of an element</td>
</tr>
<tr>
<td>oncut</td>
<td>The event occurs when the user cuts the content of an element</td>
</tr>
<tr>
<td>ondbiclick</td>
<td>The event occurs when the user double-clicks on an element</td>
</tr>
<tr>
<td>ondrag</td>
<td>The event occurs when an element is being dragged</td>
</tr>
<tr>
<td>ondragend</td>
<td>The event occurs when the user has finished dragging an element</td>
</tr>
<tr>
<td>ondragenter</td>
<td>The event occurs when the dragged element enters the drop target</td>
</tr>
<tr>
<td>ondragleave</td>
<td>The event occurs when the dragged element leaves the drop target</td>
</tr>
<tr>
<td>ondragover</td>
<td>The event occurs when the dragged element is over the drop target</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ondragstart</td>
<td>The event occurs when the user starts to drag an element</td>
</tr>
<tr>
<td>ondrop</td>
<td>The event occurs when the dragged element is dropped on the drop target</td>
</tr>
<tr>
<td>ondurationchange</td>
<td>The event occurs when the duration of the media is changed</td>
</tr>
<tr>
<td>onemptied</td>
<td>The event occurs when something bad happens and the media file is suddenly unavailable (like unexpectedly disconnects)</td>
</tr>
<tr>
<td>onended</td>
<td>The event occurs when the media has reach the end (useful for messages like &quot;thanks for listening&quot;)</td>
</tr>
<tr>
<td>onerror</td>
<td>The event occurs when an error occurs while loading an external file</td>
</tr>
<tr>
<td>onfocus</td>
<td>The event occurs when an element gets focus</td>
</tr>
<tr>
<td>onfocusin</td>
<td>The event occurs when an element is about to get focus</td>
</tr>
<tr>
<td>onfocusout</td>
<td>The event occurs when an element is about to lose focus</td>
</tr>
<tr>
<td>onfullscreenchange</td>
<td>The event occurs when an element is displayed in fullscreen mode</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onfullscreenerror</td>
<td>The event occurs when an element cannot be displayed in fullscreen mode</td>
</tr>
<tr>
<td>onhashchange</td>
<td>The event occurs when there has been changes to the anchor part of a URL</td>
</tr>
<tr>
<td>oninput</td>
<td>The event occurs when an element gets user input</td>
</tr>
<tr>
<td>oninvalid</td>
<td>The event occurs when an element is invalid</td>
</tr>
<tr>
<td>onkeydown</td>
<td>The event occurs when the user is pressing a key</td>
</tr>
<tr>
<td>onkeypress</td>
<td>The event occurs when the user presses a key</td>
</tr>
<tr>
<td>onkeyup</td>
<td>The event occurs when the user releases a key</td>
</tr>
<tr>
<td>online</td>
<td>Determines whether the browser is online</td>
</tr>
<tr>
<td>onload</td>
<td>The event occurs when an object has loaded</td>
</tr>
<tr>
<td>onloadeddata</td>
<td>The event occurs when media data is loaded</td>
</tr>
<tr>
<td>onloadedmetadata</td>
<td>The event occurs when meta data (like dimensions and duration) are loaded</td>
</tr>
<tr>
<td>onloadstart</td>
<td>The event occurs when the browser starts looking for the specified media</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onmessage</td>
<td>The event occurs when a message is received through the event source</td>
</tr>
<tr>
<td>onmousedown</td>
<td>The event occurs when the user presses a mouse button over an element</td>
</tr>
<tr>
<td>onmouseenter</td>
<td>The event occurs when the pointer is moved onto an element</td>
</tr>
<tr>
<td>onmouseleave</td>
<td>The event occurs when the pointer is moved out of an element</td>
</tr>
<tr>
<td>onmousemove</td>
<td>The event occurs when the pointer is moving while it is over an element</td>
</tr>
<tr>
<td>onmouseover</td>
<td>The event occurs when the pointer is moved onto an element, or onto one of its children</td>
</tr>
<tr>
<td>onmouseout</td>
<td>The event occurs when a user moves the mouse pointer out of an element, or out of one of its children</td>
</tr>
<tr>
<td>onmouseup</td>
<td>The event occurs when a user releases a mouse button over an element</td>
</tr>
<tr>
<td>onmousewheel</td>
<td>Deprecated. Use the wheel event instead</td>
</tr>
<tr>
<td>onoffline</td>
<td>The event occurs when the browser starts to work offline</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>ononline</code></td>
<td>The event occurs when the browser starts to work online</td>
</tr>
<tr>
<td><code>onopen</code></td>
<td>The event occurs when a connection with the event source is opened</td>
</tr>
<tr>
<td><code>onpagehide</code></td>
<td>The event occurs when the user navigates away from a webpage</td>
</tr>
<tr>
<td><code>onpageshow</code></td>
<td>The event occurs when the user navigates to a webpage</td>
</tr>
<tr>
<td><code>onpaste</code></td>
<td>The event occurs when the user pastes some content in an element</td>
</tr>
<tr>
<td><code>onpause</code></td>
<td>The event occurs when the media is paused either by the user or programmatically</td>
</tr>
<tr>
<td><code>onplay</code></td>
<td>The event occurs when the media has been started or is no longer paused</td>
</tr>
<tr>
<td><code>onplaying</code></td>
<td>The event occurs when the media is playing after having been paused or stopped for buffering</td>
</tr>
<tr>
<td><code>onpopstate</code></td>
<td>The event occurs when the window's history changes</td>
</tr>
<tr>
<td><code>onprogress</code></td>
<td>The event occurs when the browser is in the process of getting the media data (downloading</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onratechange</td>
<td>The event occurs when the playing speed of the media is changed</td>
</tr>
<tr>
<td>onresize</td>
<td>The event occurs when the document view is resized</td>
</tr>
<tr>
<td>onreset</td>
<td>The event occurs when a form is reset</td>
</tr>
<tr>
<td>onscroll</td>
<td>The event occurs when an element's scrollbar is being scrolled</td>
</tr>
<tr>
<td>onsearch</td>
<td>The event occurs when the user writes something in a search field (for <code>&lt;input=&quot;search&quot;&gt;</code>)</td>
</tr>
<tr>
<td>onseeked</td>
<td>The event occurs when the user is finished moving/skipping to a new position in the media</td>
</tr>
<tr>
<td>onseeking</td>
<td>The event occurs when the user starts moving/skipping to a new position in the media</td>
</tr>
<tr>
<td>onselect</td>
<td>The event occurs after the user selects some text (for <code>&lt;input&gt;</code> and <code>&lt;textarea&gt;</code>)</td>
</tr>
<tr>
<td>onshow</td>
<td>The event occurs when a <code>&lt;menu&gt;</code> element is shown as a context menu</td>
</tr>
<tr>
<td>onstalled</td>
<td>The event occurs when the browser is trying to</td>
</tr>
<tr>
<td>Event Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onstorage</td>
<td>The event occurs when a Web Storage area is updated</td>
</tr>
<tr>
<td>onsubmit</td>
<td>The event occurs when a form is submitted</td>
</tr>
<tr>
<td>onsuspend</td>
<td>The event occurs when the browser is intentionally not getting media data</td>
</tr>
<tr>
<td>ontimeupdate</td>
<td>The event occurs when the playing position has changed (like when the user fast forwards to a different point in the media)</td>
</tr>
<tr>
<td>ontoggle</td>
<td>The event occurs when the user opens or closes the <code>&lt;details&gt;</code> element</td>
</tr>
<tr>
<td>ontouchcancel</td>
<td>The event occurs when the touch is interrupted</td>
</tr>
<tr>
<td>ontouchend</td>
<td>The event occurs when a finger is removed from a touch screen</td>
</tr>
<tr>
<td>ontouchmove</td>
<td>The event occurs when a finger is dragged across the screen</td>
</tr>
<tr>
<td>ontouchstart</td>
<td>The event occurs when a finger is placed on a touch screen</td>
</tr>
<tr>
<td>ontransitionend</td>
<td>The event occurs when a CSS transition has completed</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>onunload</td>
<td>The event occurs once a page has unloaded (for &lt;body&gt;)</td>
</tr>
<tr>
<td>onvolumechange</td>
<td>The event occurs when the volume of the media has changed (includes setting the volume to &quot;mute&quot;)</td>
</tr>
<tr>
<td>onwaiting</td>
<td>The event occurs when the media has paused but is expected to resume (like when the media pauses to buffer more data)</td>
</tr>
<tr>
<td>onwheel</td>
<td>The event occurs when the mouse wheel rolls up or down over an element</td>
</tr>
<tr>
<td>open()</td>
<td>Opens an HTML output stream to collect output from document.write()</td>
</tr>
<tr>
<td>opener</td>
<td>Returns a reference to the window that created the window</td>
</tr>
<tr>
<td>origin</td>
<td>Returns the protocol, hostname and port number of a URL</td>
</tr>
<tr>
<td>outerHeight</td>
<td>Returns the height of the browser window, including toolbars/scrollbars</td>
</tr>
<tr>
<td>outerHTML</td>
<td>Sets or returns the outer content of an element</td>
</tr>
<tr>
<td>outerText</td>
<td>Sets or returns the text outer content of a</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>outerWidth</td>
<td>Returns the width of the browser window, including toolbars/scrollbars</td>
</tr>
<tr>
<td>ownerDocument</td>
<td>Returns the root element (document object) for an element</td>
</tr>
<tr>
<td>pageX</td>
<td>Returns the horizontal coordinate of the mouse pointer, relative to the document, when the mouse event was triggered</td>
</tr>
<tr>
<td>pageXOffset</td>
<td>Returns the pixels the current document has been scrolled (horizontally) from the upper left corner of the window</td>
</tr>
<tr>
<td>pageY</td>
<td>Returns the vertical coordinate of the mouse pointer, relative to the document, when the mouse event was triggered</td>
</tr>
<tr>
<td>pageYOffset</td>
<td>Returns the pixels the current document has been scrolled (vertically) from the upper left corner of the window</td>
</tr>
<tr>
<td>parent</td>
<td>Returns the parent window of the current window</td>
</tr>
<tr>
<td>parentNode</td>
<td>Returns the parent node of an element</td>
</tr>
<tr>
<td>parentElement</td>
<td>Returns the parent element node of an element</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>parse()</td>
<td>Parses a date string and returns the number of milliseconds since January 1, 1970</td>
</tr>
<tr>
<td>parseFloat()</td>
<td>Parses a string and returns a floating point number</td>
</tr>
<tr>
<td>parseInt()</td>
<td>Parses a string and returns an integer</td>
</tr>
<tr>
<td>pathname</td>
<td>Sets or returns the path name of a URL</td>
</tr>
<tr>
<td>persisted</td>
<td>Returns whether the webpage was cached by the browser</td>
</tr>
<tr>
<td>PI</td>
<td>Returns PI (approx. 3.14)</td>
</tr>
<tr>
<td>pixelDepth</td>
<td>Returns the color resolution (in bits per pixel) of the screen</td>
</tr>
<tr>
<td>platform</td>
<td>Returns for which platform the browser is compiled</td>
</tr>
<tr>
<td>pop()</td>
<td>Removes the last element of an array, and returns that element</td>
</tr>
<tr>
<td>port</td>
<td>Sets or returns the port number of a URL</td>
</tr>
<tr>
<td>position</td>
<td>Returns the position of the concerned device at a given time</td>
</tr>
<tr>
<td>positionError</td>
<td>Returns the reason of an error occurring when using the geolocating device</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------------------</td>
</tr>
<tr>
<td>positionOptions</td>
<td>Describes an object containing option properties to pass as a parameter of Geolocation.getCurrentPosition() and Geolocation.watchPosition()</td>
</tr>
<tr>
<td>POSITIVE_INFINITY</td>
<td>Represents infinity (returned on overflow)</td>
</tr>
<tr>
<td>preventDefault()</td>
<td>Cancels the event if it is cancelable, meaning that the default action that belongs to the event will not occur</td>
</tr>
<tr>
<td>print()</td>
<td>Prints the content of the current window</td>
</tr>
<tr>
<td>product</td>
<td>Returns the engine name of the browser</td>
</tr>
<tr>
<td>propertyName</td>
<td>Returns the name of the CSS property associated with the animation or transition</td>
</tr>
<tr>
<td>protocol</td>
<td>Sets or returns the protocol of a URL</td>
</tr>
<tr>
<td>prototype</td>
<td>Allows you to add properties and methods to an Array object</td>
</tr>
<tr>
<td>pseudoElement</td>
<td>Returns the name of the pseudo-element of the animation or transition</td>
</tr>
<tr>
<td>push()</td>
<td>Adds new elements to the end of an array, and</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>querySelector()</td>
<td>Returns the first element that matches a specified CSS selector(s) in the document</td>
</tr>
<tr>
<td>querySelectorAll()</td>
<td>Returns a static NodeList containing all elements that matches a specified CSS selector(s) in the document</td>
</tr>
<tr>
<td>random()</td>
<td>Returns a random number between 0 and 1</td>
</tr>
<tr>
<td>readyState</td>
<td>Returns the (loading) status of the document</td>
</tr>
<tr>
<td>reduce()</td>
<td>Reduce the values of an array to a single value (going left-to-right)</td>
</tr>
<tr>
<td>reduceRight()</td>
<td>Reduce the values of an array to a single value (going right-to-left)</td>
</tr>
<tr>
<td>referrer</td>
<td>Returns the URL of the document that loaded the current document</td>
</tr>
<tr>
<td>region</td>
<td></td>
</tr>
<tr>
<td>reload()</td>
<td>Reloads the current document</td>
</tr>
<tr>
<td>remove()</td>
<td>Removes the element from the DOM</td>
</tr>
<tr>
<td>removeAttribute()</td>
<td>Removes a specified attribute from an element</td>
</tr>
<tr>
<td>Method Name</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>removeAttributeNode()</code></td>
<td>Removes a specified attribute node, and returns the removed node.</td>
</tr>
<tr>
<td><code>removeChild()</code></td>
<td>Removes a child node from an element.</td>
</tr>
<tr>
<td><code>removeEventListener()</code></td>
<td>Removes an event handler that has been attached with the addEventListener() method.</td>
</tr>
<tr>
<td><code>removeItem()</code></td>
<td>Removes that key from the storage.</td>
</tr>
<tr>
<td><code>repeat</code></td>
<td>Returns whether a key is being hold down repeatedly, or not.</td>
</tr>
<tr>
<td><code>repeat()</code></td>
<td>Returns a new string with a specified number of copies of an existing string.</td>
</tr>
<tr>
<td><code>replace()</code></td>
<td>Searches a string for a specified value, or a regular expression, and returns a new string where the specified values are replaced</td>
</tr>
<tr>
<td><code>replaceChild()</code></td>
<td>Replaces a child node in an element.</td>
</tr>
<tr>
<td><code>requestAnimationFrame()</code></td>
<td>Requests the browser to call a function to update an animation before the next repaint.</td>
</tr>
<tr>
<td><code>requestFullscreen()</code></td>
<td>Shows an element in full screen mode.</td>
</tr>
<tr>
<td><code>resizeBy()</code></td>
<td>Resizes the window by the specified pixels.</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>resizeTo()</td>
<td>Resizes the window to the specified width and height</td>
</tr>
<tr>
<td>return</td>
<td>Stops the execution of a function and returns a value from that function</td>
</tr>
<tr>
<td>reverse()</td>
<td>Reverses the order of the elements in an array</td>
</tr>
<tr>
<td>round()</td>
<td>Rounds x to the nearest integer</td>
</tr>
<tr>
<td>pow()</td>
<td>Returns the value of x to the power of y</td>
</tr>
<tr>
<td>previousSibling</td>
<td>Returns the previous node at the same node tree level</td>
</tr>
<tr>
<td>previousElementSibling</td>
<td>Returns the previous element at the same node tree level</td>
</tr>
<tr>
<td>prompt()</td>
<td>Displays a dialog box that prompts the visitor for input</td>
</tr>
<tr>
<td>prototype</td>
<td>Allows you to add properties and methods to an object</td>
</tr>
<tr>
<td>relatedTarget</td>
<td>Returns the element related to the element that triggered the mouse event</td>
</tr>
<tr>
<td>removeEventListener()</td>
<td>Removes an event handler from the document (that has been attached with the addEventListener() method)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>removeNamedItem()</td>
<td>Removes a specified attribute node</td>
</tr>
<tr>
<td>renameNode()</td>
<td>Renames the specified node</td>
</tr>
<tr>
<td>screen</td>
<td>Returns the Screen object for the window (See Screen object)</td>
</tr>
<tr>
<td>screenLeft</td>
<td>Returns the horizontal coordinate of the window relative to the screen</td>
</tr>
<tr>
<td>screenTop</td>
<td>Returns the vertical coordinate of the window relative to the screen</td>
</tr>
<tr>
<td>screenX</td>
<td>Returns the horizontal coordinate of the window/mouse pointer relative to the screen</td>
</tr>
<tr>
<td>screenY</td>
<td>Returns the vertical coordinate of the window/mouse pointer relative to the screen</td>
</tr>
<tr>
<td>scripts</td>
<td>Returns a collection of &lt;script&gt; elements in the document</td>
</tr>
<tr>
<td>scroll()</td>
<td>Deprecated. This method has been replaced by the scrollTo() method.</td>
</tr>
<tr>
<td>scrollBy()</td>
<td>Scrolls the document by the specified number of pixels</td>
</tr>
<tr>
<td>scrollHeight</td>
<td>Returns the entire height of an element, including padding</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>scrollIntoView()</code></td>
<td>Scrolls the specified element into the visible area of the browser window</td>
</tr>
<tr>
<td><code>scrollTop</code></td>
<td>Sets or returns the number of pixels an element's content is scrolled vertically</td>
</tr>
<tr>
<td><code>scrollTo()</code></td>
<td>Scrolls the document to the specified coordinates</td>
</tr>
<tr>
<td><code>scrollWidth</code></td>
<td>Returns the entire width of an element, including padding</td>
</tr>
<tr>
<td><code>scrollX</code></td>
<td>An alias of <code>pageXOffset</code></td>
</tr>
<tr>
<td><code>scrollY</code></td>
<td>An alias of <code>pageYOffset</code></td>
</tr>
<tr>
<td><code>search</code></td>
<td>Sets or returns the querystring part of a URL</td>
</tr>
<tr>
<td><code>search()</code></td>
<td>Searches a string for a specified value, or regular expression, and returns the position of the match</td>
</tr>
<tr>
<td><code>self</code></td>
<td>Returns the current window</td>
</tr>
<tr>
<td><code>sessionStorage</code></td>
<td>Allows to save key/value pairs in a web browser. Stores the data for one session</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>setAttribute()</code></td>
<td>Sets or changes the specified attribute, to the specified value</td>
</tr>
<tr>
<td><code>setAttributeNode()</code></td>
<td>Sets or changes the specified attribute node</td>
</tr>
<tr>
<td><code>setDate()</code></td>
<td>Sets the day of the month of a date object</td>
</tr>
<tr>
<td><code>setFullYear()</code></td>
<td>Sets the year of a date object</td>
</tr>
<tr>
<td><code>setHours()</code></td>
<td>Sets the hour of a date object</td>
</tr>
<tr>
<td><code>setInterval()</code></td>
<td>Calls a function or evaluates an expression at specified intervals (in milliseconds)</td>
</tr>
<tr>
<td><code>setItem()</code></td>
<td>Adds that key to the storage, or update that key's value if it already exists</td>
</tr>
<tr>
<td><code>setMilliseconds()</code></td>
<td>Sets the milliseconds of a date object</td>
</tr>
<tr>
<td><code>setMinutes()</code></td>
<td>Set the minutes of a date object</td>
</tr>
<tr>
<td><code>setMonth()</code></td>
<td>Sets the month of a date object</td>
</tr>
<tr>
<td><code>setNamedItem()</code></td>
<td>Sets the specified attribute node (by name)</td>
</tr>
<tr>
<td><code>setSeconds()</code></td>
<td>Sets the seconds of a date object</td>
</tr>
<tr>
<td><code>setTime()</code></td>
<td>Sets a date to a specified number of</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>setTimeout()</code></td>
<td>Calls a function or evaluates an expression after a specified number of milliseconds</td>
</tr>
<tr>
<td><code>setUTCDate()</code></td>
<td>Sets the day of the month of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCFullYear()</code></td>
<td>Sets the year of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCHours()</code></td>
<td>Sets the hour of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCMilliseconds()</code></td>
<td>Sets the milliseconds of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCMinutes()</code></td>
<td>Sets the minutes of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCMonth()</code></td>
<td>Sets the month of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setUTCSeconds()</code></td>
<td>Sets the seconds of a date object, according to universal time</td>
</tr>
<tr>
<td><code>setYear()</code></td>
<td>Deprecated. Use the setFullYear() method instead</td>
</tr>
<tr>
<td><code>shift()</code></td>
<td>Removes the first element of an array, and</td>
</tr>
</tbody>
</table>

milliseconds after/before January 1, 1970
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>shiftKey</td>
<td>Returns whether the &quot;SHIFT&quot; key was pressed when an event was triggered.</td>
</tr>
<tr>
<td>sin()</td>
<td>Returns the sine of x (x is in radians).</td>
</tr>
<tr>
<td>sinh()</td>
<td>Returns the hyperbolic sine of x.</td>
</tr>
<tr>
<td>slice()</td>
<td>Selects a part of an array, and returns the new array.</td>
</tr>
<tr>
<td>some()</td>
<td>Checks if any of the elements in an array pass a test.</td>
</tr>
<tr>
<td>sort()</td>
<td>Sorts the elements of an array.</td>
</tr>
<tr>
<td>source</td>
<td>Returns the text of the RegExp pattern.</td>
</tr>
<tr>
<td>specified</td>
<td>Returns true if the attribute has been specified, otherwise it returns false</td>
</tr>
<tr>
<td>splice()</td>
<td>Adds/Removes elements from an array.</td>
</tr>
<tr>
<td>split()</td>
<td>Splits a string into an array of substrings.</td>
</tr>
<tr>
<td>static</td>
<td>Defines a static method for a class.</td>
</tr>
<tr>
<td>startsWith()</td>
<td>Checks whether a string begins with specified</td>
</tr>
<tr>
<td>Function/Method</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>state</td>
<td>Returns an object containing a copy of the history entries</td>
</tr>
<tr>
<td>String()</td>
<td>Converts an object's value to a string</td>
</tr>
<tr>
<td>stop()</td>
<td>Stops the window from loading</td>
</tr>
<tr>
<td>stopImmediatePropagation()</td>
<td>Prevents other listeners of the same event from being called</td>
</tr>
<tr>
<td>stopPropagation()</td>
<td>Prevents further propagation of an event during event flow</td>
</tr>
<tr>
<td>stringify()</td>
<td>Convert a JavaScript object to a JSON string</td>
</tr>
<tr>
<td>sqrt()</td>
<td>Returns the square root of x</td>
</tr>
<tr>
<td>SQRT1_2</td>
<td>Returns the square root of 1/2 (approx. 0.707)</td>
</tr>
<tr>
<td>SQRT2</td>
<td>Returns the square root of 2 (approx. 1.414)</td>
</tr>
<tr>
<td>status</td>
<td>Sets or returns the text in the statusbar of a window</td>
</tr>
<tr>
<td>strictErrorChecking</td>
<td>Sets or returns whether error-checking is enforced or not</td>
</tr>
<tr>
<td>storageArea</td>
<td>Returns an object representing the affected storage object</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------</td>
</tr>
<tr>
<td>style</td>
<td>Sets or returns the value of the style attribute of an element</td>
</tr>
<tr>
<td>substr()</td>
<td>Extracts the characters from a string, beginning at a specified start position, and through the specified number of character</td>
</tr>
<tr>
<td>substring()</td>
<td>Extracts the characters from a string, between two specified indices</td>
</tr>
<tr>
<td>super</td>
<td>Refers to the parent class</td>
</tr>
<tr>
<td>switch</td>
<td>Marks a block of statements to be executed depending on different cases</td>
</tr>
<tr>
<td>table()</td>
<td>Displays tabular data as a table</td>
</tr>
<tr>
<td>tabIndex</td>
<td>Sets or returns the value of the tabindex attribute of an element</td>
</tr>
<tr>
<td>tagName</td>
<td>Returns the tag name of an element</td>
</tr>
<tr>
<td>tan()</td>
<td>Returns the tangent of an angle</td>
</tr>
<tr>
<td>tanh()</td>
<td>Returns the hyperbolic tangent of a number</td>
</tr>
<tr>
<td><strong>target</strong></td>
<td>Returns the element that triggered the event</td>
</tr>
<tr>
<td>------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td><strong>targetTouches</strong></td>
<td>Returns a list of all the touch objects that are in contact with the surface and where the touchstart event occurred on the same target element as the current target element</td>
</tr>
<tr>
<td><strong>test()</strong></td>
<td>Tests for a match in a string. Returns true or false</td>
</tr>
<tr>
<td><strong>textContent</strong></td>
<td>Sets or returns the textual content of a node and its descendants</td>
</tr>
<tr>
<td><strong>this</strong></td>
<td>Refers to the object it belongs to</td>
</tr>
<tr>
<td><strong>throw</strong></td>
<td>Throws (generates) an error</td>
</tr>
<tr>
<td><strong>time()</strong></td>
<td>Starts a timer (can track how long an operation takes)</td>
</tr>
<tr>
<td><strong>timeEnd()</strong></td>
<td>Stops a timer that was previously started by console.time()</td>
</tr>
<tr>
<td><strong>timeStamp</strong></td>
<td>Returns the time (in milliseconds relative to the epoch) at which the event was created</td>
</tr>
<tr>
<td><strong>title</strong></td>
<td>Sets or returns the title of the document</td>
</tr>
<tr>
<td><strong>toDateString()</strong></td>
<td>Converts the date portion of a Date object into a readable string</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>toGMTString()</td>
<td>Deprecated. Use the toUTCString() method instead</td>
</tr>
<tr>
<td>toExponential()</td>
<td>Converts a number into an exponential notation</td>
</tr>
<tr>
<td>toFixed(x)</td>
<td>Formats a number with x numbers of digits after the decimal point</td>
</tr>
<tr>
<td>toJSON()</td>
<td>Returns the date as a string, formatted as a JSON date</td>
</tr>
<tr>
<td>toISOString()</td>
<td>Returns the date as a string, using the ISO standard</td>
</tr>
<tr>
<td>toLocaleDateString()</td>
<td>Returns the date portion of a Date object as a string, using locale conventions</td>
</tr>
<tr>
<td>toLocaleLowerCase()</td>
<td>Converts a string to lowercase letters, according to the host's locale</td>
</tr>
<tr>
<td>toLocaleString()</td>
<td>Converts a Date object to a string, using locale conventions</td>
</tr>
<tr>
<td>toLocaleTimeString()</td>
<td>Returns the time portion of a Date object as a string, using locale conventions</td>
</tr>
<tr>
<td>toLocaleUpperCase()</td>
<td>Converts a string to uppercase letters, according to the host's locale</td>
</tr>
<tr>
<td>toLowerCase()</td>
<td>Converts a string to lowercase letters</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>top</td>
<td>Returns the topmost browser window</td>
</tr>
<tr>
<td>toPrecision(x)</td>
<td>Formats a number to x length</td>
</tr>
<tr>
<td>toString()</td>
<td>Converts an array to a string, and returns the result</td>
</tr>
<tr>
<td>total</td>
<td>Returns the total amount of work that will be loaded</td>
</tr>
<tr>
<td>toTimeString()</td>
<td>Converts the time portion of a Date object to a string</td>
</tr>
<tr>
<td>touches</td>
<td>Returns a list of all the touch objects that are currently in contact with the surface</td>
</tr>
<tr>
<td>toUpperCase()</td>
<td>Converts a string to uppercase letters</td>
</tr>
<tr>
<td>toUTCString()</td>
<td>Converts a Date object to a string, according to universal time</td>
</tr>
<tr>
<td>trace()</td>
<td>Outputs a stack trace to the console</td>
</tr>
<tr>
<td>transitionend</td>
<td>The event occurs when a CSS transition has completed</td>
</tr>
<tr>
<td>trim()</td>
<td>Removes whitespace from both ends of a string</td>
</tr>
<tr>
<td>trunc()</td>
<td>Returns the integer part of a number (x)</td>
</tr>
<tr>
<td>try ... catch ... finally</td>
<td>Marks the block of statements to be executed when an error occurs in a try block, and implements error handling</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>type</td>
<td>Returns the name of the event</td>
</tr>
<tr>
<td>typeof</td>
<td>Returns the type of a variable, object, function or expression</td>
</tr>
<tr>
<td>undefined</td>
<td>Indicates that a variable has not been assigned a value</td>
</tr>
<tr>
<td>unescape()</td>
<td>Deprecated in version 1.5. Use decodeURI() or decodeURIComponent() instead</td>
</tr>
<tr>
<td>unshift()</td>
<td>Adds new elements to the beginning of an array, and returns the new length</td>
</tr>
<tr>
<td>url</td>
<td>Returns the URL of the changed item's document</td>
</tr>
<tr>
<td>URL</td>
<td>Returns the full URL of the HTML document</td>
</tr>
<tr>
<td>userAgent</td>
<td>Returns the user-agent header sent by the browser to the server</td>
</tr>
<tr>
<td>UTC()</td>
<td>Returns the number of milliseconds in a date since midnight of January 1, 1970, according to UTC time</td>
</tr>
<tr>
<td>value</td>
<td>Sets or returns the value of the attribute</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>valueOf()</td>
<td>Returns the primitive value of an array</td>
</tr>
<tr>
<td>var</td>
<td>Declares a variable</td>
</tr>
<tr>
<td>warn()</td>
<td>Outputs a warning message to the console</td>
</tr>
<tr>
<td>watchPosition()</td>
<td>Returns a watch ID value that then can be used to unregister the handler by passing it to the Geolocation.clearWatch() method</td>
</tr>
<tr>
<td>which</td>
<td>Returns which mouse button was pressed when the mouse event was triggered</td>
</tr>
<tr>
<td>width</td>
<td>Returns the total width of the screen</td>
</tr>
<tr>
<td>view</td>
<td>Returns a reference to the Window object where the event occurred</td>
</tr>
<tr>
<td>void</td>
<td>Evaluates an expression and returns undefined</td>
</tr>
<tr>
<td>while</td>
<td>Marks a block of statements to be executed while a condition is true</td>
</tr>
<tr>
<td>write()</td>
<td>Writes HTML expressions or JavaScript code to a document</td>
</tr>
<tr>
<td>writeln()</td>
<td>Same as write(), but adds a newline character</td>
</tr>
<tr>
<td>SQL Statement</td>
<td>Syntax</td>
</tr>
<tr>
<td>---------------</td>
<td>--------</td>
</tr>
</tbody>
</table>
| **AND / OR**  | SELECT column_name(s)  
                FROM table_name  
                WHERE condition  
                AND|OR condition |
| **ALTER TABLE** | ALTER TABLE table_name  
                    ADD column_name datatype  
                    or  
                    ALTER TABLE table_name  
                    DROP COLUMN column_name |
| **AS (alias)** | SELECT column_name AS column_alias  
                    FROM table_name  
                    or  
                    SELECT column_name  
                    FROM table_name AS table_alias |
| **BETWEEN**    | SELECT column_name(s)  
                    FROM table_name  
                    WHERE column_name  
                    BETWEEN value1 AND value2 |
<table>
<thead>
<tr>
<th>Command</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>CREATE DATABASE</td>
<td><code>CREATE DATABASE database_name</code></td>
</tr>
<tr>
<td>CREATE TABLE</td>
<td><code>CREATE TABLE table_name</code></td>
</tr>
<tr>
<td></td>
<td>(</td>
</tr>
<tr>
<td></td>
<td>column_name1 data_type,</td>
</tr>
<tr>
<td></td>
<td>column_name2 data_type,</td>
</tr>
<tr>
<td></td>
<td>column_name3 data_type,</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>)</td>
</tr>
<tr>
<td>CREATE INDEX</td>
<td><code>CREATE INDEX index_name</code></td>
</tr>
<tr>
<td></td>
<td>ON table_name (column_name)</td>
</tr>
<tr>
<td></td>
<td>(Note: Deletes the entire table!!)</td>
</tr>
<tr>
<td>CREATE VIEW</td>
<td><code>CREATE VIEW view_name AS</code></td>
</tr>
<tr>
<td></td>
<td>SELECT column_name(s)</td>
</tr>
<tr>
<td></td>
<td>FROM table_name WHERE condition</td>
</tr>
<tr>
<td>DELETE</td>
<td><code>DELETE FROM table_name</code></td>
</tr>
<tr>
<td></td>
<td>WHERE some_column=some_value</td>
</tr>
<tr>
<td></td>
<td>(Note: Deletes the entire table!!)</td>
</tr>
<tr>
<td>DROP DATABASE</td>
<td><code>DROP DATABASE database_name</code></td>
</tr>
<tr>
<td>DROP INDEX</td>
<td><code>DROP INDEX table_name.index_name (SQL Server)</code></td>
</tr>
<tr>
<td></td>
<td><code>DROP INDEX index_name ON table_name (MS Access)</code></td>
</tr>
<tr>
<td></td>
<td><code>DROP INDEX index_name (DB2/Oracle)</code></td>
</tr>
<tr>
<td></td>
<td><code>ALTER TABLE table_name</code></td>
</tr>
<tr>
<td></td>
<td><code>DROP INDEX index_name (MySQL)</code></td>
</tr>
<tr>
<td>DROP TABLE</td>
<td><code>DROP TABLE table_name</code></td>
</tr>
</tbody>
</table>
EXISTS
IF EXISTS (SELECT * FROM table_name WHERE id = ?)
BEGIN
  --do what needs to be done if exists
END
ELSE
BEGIN
  --do what needs to be done if not
END

GROUP BY
SELECT column_name,
aggregate_function(column_name)
FROM table_name
WHERE column_name operator value
GROUP BY column_name

HAVING
SELECT column_name,
aggregate_function(column_name)
FROM table_name
WHERE column_name operator value
GROUP BY column_name
HAVING aggregate_function(column_name) operator value

IN
SELECT column_name(s)
FROM table_name
WHERE column_name
IN (value1,value2,..)

INSERT INTO
INSERT INTO table_name
VALUES (value1, value2, value3,....)
or
INSERT INTO table_name
(column1, column2, column3,...)
VALUES (value1, value2, value3,....)

INNER JOIN
SELECT column_name(s)
FROM table_name1
INNER JOIN table_name2
ON table_name1.column_name=table_name2.column_name

LEFT JOIN
SELECT column_name(s)
<table>
<thead>
<tr>
<th>SQL Clause</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM table_name1</td>
<td><code>FROM table_name1</code></td>
</tr>
<tr>
<td>LEFT JOIN table_name2</td>
<td><code>LEFT JOIN table_name2</code></td>
</tr>
<tr>
<td>ON table_name1.column_name=table_name2.column_name</td>
<td><code>ON table_name1.column_name=table_name2.column_name</code></td>
</tr>
<tr>
<td>RIGHT JOIN</td>
<td><code>RIGHT JOIN</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name1</td>
<td><code>FROM table_name1</code></td>
</tr>
<tr>
<td>RIGHT JOIN table_name2</td>
<td><code>RIGHT JOIN table_name2</code></td>
</tr>
<tr>
<td>ON table_name1.column_name=table_name2.column_name</td>
<td><code>ON table_name1.column_name=table_name2.column_name</code></td>
</tr>
<tr>
<td>FULL JOIN</td>
<td><code>FULL JOIN</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name1</td>
<td><code>FROM table_name1</code></td>
</tr>
<tr>
<td>FULL JOIN table_name2</td>
<td><code>FULL JOIN table_name2</code></td>
</tr>
<tr>
<td>ON table_name1.column_name=table_name2.column_name</td>
<td><code>ON table_name1.column_name=table_name2.column_name</code></td>
</tr>
<tr>
<td>LIKE</td>
<td><code>LIKE</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>WHERE column_name LIKE pattern</td>
<td><code>WHERE column_name LIKE pattern</code></td>
</tr>
<tr>
<td>ORDER BY</td>
<td><code>ORDER BY</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>ORDER BY column_name [ASC</td>
<td>DESC]</td>
</tr>
<tr>
<td>SELECT</td>
<td><code>SELECT</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>SELECT *</td>
<td><code>SELECT *</code></td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>SELECT DISTINCT</td>
<td><code>SELECT DISTINCT</code></td>
</tr>
<tr>
<td>column_name(s)</td>
<td><code>column_name(s)</code></td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>SELECT INTO</td>
<td><code>SELECT INTO</code></td>
</tr>
<tr>
<td>SELECT *</td>
<td><code>SELECT *</code></td>
</tr>
<tr>
<td>INTO new_table_name [IN externaldatabase]</td>
<td><code>INTO new_table_name [IN externaldatabase]</code></td>
</tr>
<tr>
<td>FROM old_table_name</td>
<td><code>FROM old_table_name</code></td>
</tr>
<tr>
<td>or</td>
<td><code>or</code></td>
</tr>
<tr>
<td>SELECT column_name(s)</td>
<td><code>SELECT column_name(s)</code></td>
</tr>
<tr>
<td>INTO new_table_name [IN externaldatabase]</td>
<td><code>INTO new_table_name [IN externaldatabase]</code></td>
</tr>
<tr>
<td>FROM old_table_name</td>
<td><code>FROM old_table_name</code></td>
</tr>
<tr>
<td>SELECT TOP</td>
<td><code>SELECT TOP</code></td>
</tr>
<tr>
<td>number</td>
<td>percent column_name(s)</td>
</tr>
<tr>
<td>FROM table_name</td>
<td><code>FROM table_name</code></td>
</tr>
<tr>
<td>TRUNCATE TABLE</td>
<td><code>TRUNCATE TABLE</code></td>
</tr>
<tr>
<td>table_name</td>
<td><code>table_name</code></td>
</tr>
</tbody>
</table>
UNION
SELECT column_name(s) FROM table_name1
UNION
SELECT column_name(s) FROM table_name2

UNION ALL
SELECT column_name(s) FROM table_name1
UNION ALL
SELECT column_name(s) FROM table_name2

UPDATE
UPDATE table_name
SET column1=value, column2=value,...
WHERE some_column=some_value

WHERE
SELECT column_name(s)
FROM table_name
WHERE column_name operator value

Python Reference

Python Built in Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs()</td>
<td>Returns the absolute value of a number</td>
</tr>
<tr>
<td>all()</td>
<td>Returns True if all items in an iterable object are true</td>
</tr>
<tr>
<td>any()</td>
<td>Returns True if any item in an iterable object is true</td>
</tr>
<tr>
<td>ascii()</td>
<td>Returns a readable version of an object. Replaces none-ascii characters with escape character</td>
</tr>
<tr>
<td>bin()</td>
<td>Returns the binary version of a number</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>bool()</td>
<td>Returns the boolean value of the specified object</td>
</tr>
<tr>
<td>bytearray()</td>
<td>Returns an array of bytes</td>
</tr>
<tr>
<td>bytes()</td>
<td>Returns a bytes object</td>
</tr>
<tr>
<td>callable()</td>
<td>Returns True if the specified object is callable, otherwise False</td>
</tr>
<tr>
<td>chr()</td>
<td>Returns a character from the specified Unicode code.</td>
</tr>
<tr>
<td>classmethod()</td>
<td>Converts a method into a class method</td>
</tr>
<tr>
<td>compile()</td>
<td>Returns the specified source as an object, ready to be executed</td>
</tr>
<tr>
<td>complex()</td>
<td>Returns a complex number</td>
</tr>
<tr>
<td>delattr()</td>
<td>Deletes the specified attribute (property or method) from the specified object</td>
</tr>
<tr>
<td>dict()</td>
<td>Returns a dictionary (Array)</td>
</tr>
<tr>
<td>dir()</td>
<td>Returns a list of the specified object's properties and methods</td>
</tr>
<tr>
<td>divmod()</td>
<td>Returns the quotient and the remainder when argument1 is divided by argument2</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>enumerate()</td>
<td>Takes a collection (e.g. a tuple) and returns it as an enumerate object</td>
</tr>
<tr>
<td>eval()</td>
<td>Evaluates and executes an expression</td>
</tr>
<tr>
<td>exec()</td>
<td>Executes the specified code (or object)</td>
</tr>
<tr>
<td>filter()</td>
<td>Use a filter function to exclude items in an iterable object</td>
</tr>
<tr>
<td>float()</td>
<td>Returns a floating point number</td>
</tr>
<tr>
<td>format()</td>
<td>Formats a specified value</td>
</tr>
<tr>
<td>frozenset()</td>
<td>Returns a frozenset object</td>
</tr>
<tr>
<td>getattr()</td>
<td>Returns the value of the specified attribute (property or method)</td>
</tr>
<tr>
<td>globals()</td>
<td>Returns the current global symbol table as a dictionary</td>
</tr>
<tr>
<td>hasattr()</td>
<td>Returns True if the specified object has the specified attribute (property/method)</td>
</tr>
<tr>
<td>hash()</td>
<td>Returns the hash value of a specified object</td>
</tr>
<tr>
<td>help()</td>
<td>Executes the built-in help system</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>hex()</td>
<td>Converts a number into a hexadecimal value</td>
</tr>
<tr>
<td>id()</td>
<td>Returns the id of an object</td>
</tr>
<tr>
<td>input()</td>
<td>Allowing user input</td>
</tr>
<tr>
<td>int()</td>
<td>Returns an integer number</td>
</tr>
<tr>
<td>isinstance()</td>
<td>Returns True if a specified object is an instance of a specified object</td>
</tr>
<tr>
<td>issubclass()</td>
<td>Returns True if a specified class is a subclass of a specified object</td>
</tr>
<tr>
<td>iter()</td>
<td>Returns an iterator object</td>
</tr>
<tr>
<td>len()</td>
<td>Returns the length of an object</td>
</tr>
<tr>
<td>list()</td>
<td>Returns a list</td>
</tr>
<tr>
<td>locals()</td>
<td>Returns an updated dictionary of the current local symbol table</td>
</tr>
<tr>
<td>map()</td>
<td>Returns the specified iterator with the specified function applied to each item</td>
</tr>
<tr>
<td>max()</td>
<td>Returns the largest item in an iterable</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>memoryview()</td>
<td>Returns a memory view object</td>
</tr>
<tr>
<td>min()</td>
<td>Returns the smallest item in an iterable</td>
</tr>
<tr>
<td>next()</td>
<td>Returns the next item in an iterable</td>
</tr>
<tr>
<td>object()</td>
<td>Returns a new object</td>
</tr>
<tr>
<td>oct()</td>
<td>Converts a number into an octal</td>
</tr>
<tr>
<td>open()</td>
<td>Opens a file and returns a file object</td>
</tr>
<tr>
<td>ord()</td>
<td>Convert an integer representing the Unicode of the specified character</td>
</tr>
<tr>
<td>pow()</td>
<td>Returns the value of x to the power of y</td>
</tr>
<tr>
<td>print()</td>
<td>Prints to the standard output device</td>
</tr>
<tr>
<td>property()</td>
<td>Gets, sets, deletes a property</td>
</tr>
<tr>
<td>range()</td>
<td>Returns a sequence of numbers, starting from 0 and increments by 1 (by default)</td>
</tr>
<tr>
<td>repr()</td>
<td>Returns a readable version of an object</td>
</tr>
<tr>
<td>reversed()</td>
<td>Returns a reversed iterator</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>round()</code></td>
<td>Rounds a number</td>
</tr>
<tr>
<td><code>set()</code></td>
<td>Returns a new set object</td>
</tr>
<tr>
<td><code>setattr()</code></td>
<td>Sets an attribute (property/method) of an object</td>
</tr>
<tr>
<td><code>slice()</code></td>
<td>Returns a slice object</td>
</tr>
<tr>
<td><code>sorted()</code></td>
<td>Returns a sorted list</td>
</tr>
<tr>
<td><code>@staticmethod()</code></td>
<td>Converts a method into a static method</td>
</tr>
<tr>
<td><code>str()</code></td>
<td>Returns a string object</td>
</tr>
<tr>
<td><code>sum()</code></td>
<td>Sums the items of an iterator</td>
</tr>
<tr>
<td><code>super()</code></td>
<td>Returns an object that represents the parent class</td>
</tr>
<tr>
<td><code>tuple()</code></td>
<td>Returns a tuple</td>
</tr>
<tr>
<td><code>type()</code></td>
<td>Returns the type of an object</td>
</tr>
<tr>
<td><code>vars()</code></td>
<td>Returns the <code>__dict__</code> property of an object</td>
</tr>
<tr>
<td><code>zip()</code></td>
<td>Returns an iterator, from two or more iterators</td>
</tr>
</tbody>
</table>

**Python String Methods**
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>capitalize()</td>
<td>Converts the first character to upper case</td>
</tr>
<tr>
<td>casefold()</td>
<td>Converts string into lower case</td>
</tr>
<tr>
<td>center()</td>
<td>Returns a centered string</td>
</tr>
<tr>
<td>count()</td>
<td>Returns the number of times a specified value occurs in a string</td>
</tr>
<tr>
<td>encode()</td>
<td>Returns an encoded version of the string</td>
</tr>
<tr>
<td>endswith()</td>
<td>Returns true if the string ends with the specified value</td>
</tr>
<tr>
<td>expandtabs()</td>
<td>Sets the tab size of the string</td>
</tr>
<tr>
<td>find()</td>
<td>Searches the string for a specified value and returns the position of where it was found</td>
</tr>
<tr>
<td>format()</td>
<td>Formats specified values in a string</td>
</tr>
<tr>
<td>format_map()</td>
<td>Formats specified values in a string</td>
</tr>
<tr>
<td>index()</td>
<td>Searches the string for a specified value and returns the position of where it was found</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>isalnum()</td>
<td>Returns True if all characters in the string are alphanumeric</td>
</tr>
<tr>
<td>isalpha()</td>
<td>Returns True if all characters in the string are in the alphabet</td>
</tr>
<tr>
<td>isdecimal()</td>
<td>Returns True if all characters in the string are decimals</td>
</tr>
<tr>
<td>isdigit()</td>
<td>Returns True if all characters in the string are digits</td>
</tr>
<tr>
<td>isidentifier()</td>
<td>Returns True if the string is an identifier</td>
</tr>
<tr>
<td>islower()</td>
<td>Returns True if all characters in the string are lower case</td>
</tr>
<tr>
<td>isnumeric()</td>
<td>Returns True if all characters in the string are numeric</td>
</tr>
<tr>
<td>isprintable()</td>
<td>Returns True if all characters in the string are printable</td>
</tr>
<tr>
<td>isspace()</td>
<td>Returns True if all characters in the string are whitespaces</td>
</tr>
<tr>
<td>istitle()</td>
<td>Returns True if the string follows the rules of a title</td>
</tr>
<tr>
<td>isupper()</td>
<td>Returns True if all characters in the string are</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>upper()</code></td>
<td>Converts a string into upper case</td>
</tr>
<tr>
<td><code>join()</code></td>
<td>Joins the elements of an iterable to the end of the string</td>
</tr>
<tr>
<td><code>ljust()</code></td>
<td>Returns a left justified version of the string</td>
</tr>
<tr>
<td><code>lower()</code></td>
<td>Converts a string into lower case</td>
</tr>
<tr>
<td><code>lstrip()</code></td>
<td>Returns a left trim version of the string</td>
</tr>
<tr>
<td><code>maketrans()</code></td>
<td>Returns a translation table to be used in translations</td>
</tr>
<tr>
<td><code>partition()</code></td>
<td>Returns a tuple where the string is parted into three parts</td>
</tr>
<tr>
<td><code>replace()</code></td>
<td>Returns a string where a specified value is replaced with a specified value</td>
</tr>
<tr>
<td><code>rfind()</code></td>
<td>Searches the string for a specified value and returns the last position of where it was found</td>
</tr>
<tr>
<td><code>rindex()</code></td>
<td>Searches the string for a specified value and returns the last position of where it was found</td>
</tr>
<tr>
<td><code>rjust()</code></td>
<td>Returns a right justified version of the string</td>
</tr>
<tr>
<td><code>rpartition()</code></td>
<td>Returns a tuple where the string is parted into three parts</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>rsplit()</td>
<td>Splits the string at the specified separator, and returns a list</td>
</tr>
<tr>
<td>rstrip()</td>
<td>Returns a right trim version of the string</td>
</tr>
<tr>
<td>split()</td>
<td>Splits the string at the specified separator, and returns a list</td>
</tr>
<tr>
<td>splitlines()</td>
<td>Splits the string at line breaks and returns a list</td>
</tr>
<tr>
<td>startswith()</td>
<td>Returns true if the string starts with the specified value</td>
</tr>
<tr>
<td>strip()</td>
<td>Returns a trimmed version of the string</td>
</tr>
<tr>
<td>swapcase()</td>
<td>Swaps cases, lower case becomes upper case and vice versa</td>
</tr>
<tr>
<td>title()</td>
<td>Converts the first character of each word to upper case</td>
</tr>
<tr>
<td>translate()</td>
<td>Returns a translated string</td>
</tr>
<tr>
<td>upper()</td>
<td>Converts a string into upper case</td>
</tr>
<tr>
<td>zfill()</td>
<td>Fills the string with a specified number of 0 values at the beginning</td>
</tr>
</tbody>
</table>

**Python List/Array Methods**
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>append()</td>
<td>Adds an element at the end of the list</td>
</tr>
<tr>
<td>clear()</td>
<td>Removes all the elements from the list</td>
</tr>
<tr>
<td>copy()</td>
<td>Returns a copy of the list</td>
</tr>
<tr>
<td>count()</td>
<td>Returns the number of elements with the specified value</td>
</tr>
<tr>
<td>extend()</td>
<td>Add the elements of a list (or any iterable), to the end of the current list</td>
</tr>
<tr>
<td>index()</td>
<td>Returns the index of the first element with the specified value</td>
</tr>
<tr>
<td>insert()</td>
<td>Adds an element at the specified position</td>
</tr>
<tr>
<td>pop()</td>
<td>Removes the element at the specified position</td>
</tr>
<tr>
<td>remove()</td>
<td>Removes the first item with the specified value</td>
</tr>
<tr>
<td>reverse()</td>
<td>Reverses the order of the list</td>
</tr>
<tr>
<td>sort()</td>
<td>Sorts the list</td>
</tr>
</tbody>
</table>

**Python Dictionary Methods**
<table>
<thead>
<tr>
<th><strong>Method</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>clear()</td>
<td>Removes all the elements from the dictionary</td>
</tr>
<tr>
<td>copy()</td>
<td>Returns a copy of the dictionary</td>
</tr>
<tr>
<td>fromkeys()</td>
<td>Returns a dictionary with the specified keys and value</td>
</tr>
<tr>
<td>get()</td>
<td>Returns the value of the specified key</td>
</tr>
<tr>
<td>items()</td>
<td>Returns a list containing a tuple for each key-value pair</td>
</tr>
<tr>
<td>keys()</td>
<td>Returns a list containing the dictionary's keys</td>
</tr>
<tr>
<td>pop()</td>
<td>Removes the element with the specified key</td>
</tr>
<tr>
<td>popitem()</td>
<td>Removes the last inserted key-value pair</td>
</tr>
<tr>
<td>setdefault()</td>
<td>Returns the value of the specified key. If the key does not exist: insert the key, with the specified value</td>
</tr>
<tr>
<td>update()</td>
<td>Updates the dictionary with the specified key-value pairs</td>
</tr>
<tr>
<td>values()</td>
<td>Returns a list of all the values in the dictionary</td>
</tr>
</tbody>
</table>
Write a python program to rotate pdf pages

```python
import PyPDF2

pdf = open('1.pdf', 'rb')
rd_pdf = PyPDF2.PdfFileReader(pdf)
wr_pdf = PyPDF2.PdfFileWriter()

for pg_num in range(rd_pdf.numPages):
    pdf_page = rd_pdf.getPage(pg_num)
    pdf_page.rotateClockwise(90)
    wr_pdf.addPage(pdf_page)

pdf_out = open('rotated.pdf', 'wb')
wr_pdf.write(pdf_out)
pdf_out.close()
pdf.close()
```

**Code:**

```python
import textwrap

str1 = "Python is an interpreted high-level programming language for general-purpose programming."

s = textwrap.shorten(str1, width=50)
print(s)
```

**Output on the screen:**

```
Python is an interpreted high-level [...]
```
import re

str_line = "This is python tutorial. Do you enjoy learning python?"

obj = re.match(r'(.*) enjoy (.*) \.', str_line)

if obj:
    print(obj.groups())

Output on the screen:

('This is python tutorial. Do you', 'learning')

---

**Magic 8-ball in Python**

# import modules
import sys
import random

ans1=True

while ans1:
    que=input("Ask the magic 8 ball a question(1-8) press .(dot) if you want to exit")
    print("\n")
anss=random.randint(1,8)

if que== ".":
    sys.exit()

elif anss== 1:
    print("It is certain")

elif anss== 2:
    print("Outlook good")

elif anss== 3:
    print("You may rely on it")

elif anss== 4:
    print("Ask again later")

elif anss==5:
    print("Concentrate and ask again")

elif anss== 6:
    print("Reply hazy, try again")

elif anss== 7:
    print("My reply is no")

elif anss== 8:
    print("My sources say no")
Pig Latin Translator

#Take the users input
words = input("Enter some text to translate to pig Latin: ")
print ("You entered: ", words)

#Now I need to break apart the words into a list
words = words.split(' ')

#Now words is a list, so I can manipulate each one using a loop
for i in words:
    if len(i) >= 3: #I only want to translate words greater than 3 characters
        i = i + "%say" % (i[0])
        i = i[1:] 
        print (i)
    else:
        pass

Rolling the dice

import random
min = 1
max = 6
roll_again = "yes"
while roll_again == "yes" or roll_again == "y":
    print ("Rolling the dices...")
    print ("The values are....")
    print (random.randint(min, max))
    print (random.randint(min, max))

    roll_again = input("Roll the dices again?")
## Python Tuple Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count()</td>
<td>Returns the number of times a specified value occurs in a tuple</td>
</tr>
<tr>
<td>index()</td>
<td>Searches the tuple for a specified value and returns the position of where it was found</td>
</tr>
</tbody>
</table>

## Python Set Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add()</td>
<td>Adds an element to the set</td>
</tr>
<tr>
<td>clear()</td>
<td>Removes all the elements from the set</td>
</tr>
<tr>
<td>copy()</td>
<td>Returns a copy of the set</td>
</tr>
<tr>
<td>difference()</td>
<td>Returns a set containing the difference between two or more sets</td>
</tr>
<tr>
<td>difference_update()</td>
<td>Removes the items in this set that are also included in another, specified set</td>
</tr>
<tr>
<td>discard()</td>
<td>Remove the specified item</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>intersection()</td>
<td>Returns a set, that is the intersection of two other sets</td>
</tr>
<tr>
<td>intersection_update()</td>
<td>Removes the items in this set that are not present in other, specified set(s)</td>
</tr>
<tr>
<td>isdisjoint()</td>
<td>Returns whether two sets have a intersection or not</td>
</tr>
<tr>
<td>issubset()</td>
<td>Returns whether another set contains this set or not</td>
</tr>
<tr>
<td>issuperset()</td>
<td>Returns whether this set contains another set or not</td>
</tr>
<tr>
<td>pop()</td>
<td>Removes an element from the set</td>
</tr>
<tr>
<td>remove()</td>
<td>Removes the specified element</td>
</tr>
<tr>
<td>symmetric_difference()</td>
<td>Returns a set with the symmetric differences of two sets</td>
</tr>
<tr>
<td>symmetric_difference_update()</td>
<td>inserts the symmetric differences from this set and another</td>
</tr>
<tr>
<td>union()</td>
<td>Return a set containing the union of sets</td>
</tr>
<tr>
<td>update()</td>
<td>Update the set with the union of this set and others</td>
</tr>
</tbody>
</table>

**Python File Methods**
<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>close()</td>
<td>Closes the file</td>
</tr>
<tr>
<td>detach()</td>
<td>Returns the separated raw stream from the buffer</td>
</tr>
<tr>
<td>fileno()</td>
<td>Returns a number that represents the stream, from the operating system's perspective</td>
</tr>
<tr>
<td>flush()</td>
<td>Flushes the internal buffer</td>
</tr>
<tr>
<td>isatty()</td>
<td>Returns whether the file stream is interactive or not</td>
</tr>
<tr>
<td>read()</td>
<td>Returns the file content</td>
</tr>
<tr>
<td>readable()</td>
<td>Returns whether the file stream can be read or not</td>
</tr>
<tr>
<td>readline()</td>
<td>Returns one line from the file</td>
</tr>
<tr>
<td>readlines()</td>
<td>Returns a list of lines from the file</td>
</tr>
<tr>
<td>seek()</td>
<td>Change the file position</td>
</tr>
<tr>
<td>seekable()</td>
<td>Returns whether the file allows us to change the file position</td>
</tr>
<tr>
<td>tell()</td>
<td>Returns the current file position</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>truncate()</td>
<td>Resizes the file to a specified size</td>
</tr>
<tr>
<td>writable()</td>
<td>Returns whether the file can be written to or not</td>
</tr>
<tr>
<td>write()</td>
<td>Writes the specified string to the file</td>
</tr>
<tr>
<td>writelines()</td>
<td>Writes a list of strings to the file</td>
</tr>
</tbody>
</table>

**Python Built-in Exceptions**

<table>
<thead>
<tr>
<th>Exception</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArithmeticError</td>
<td>Raised when an error occurs in numeric calculations</td>
</tr>
<tr>
<td>AssertionError</td>
<td>Raised when an assert statement fails</td>
</tr>
<tr>
<td>AttributeError</td>
<td>Raised when attribute reference or assignment fails</td>
</tr>
<tr>
<td>Exception</td>
<td>Base class for all exceptions</td>
</tr>
<tr>
<td>EOFError</td>
<td>Raised when the input() method hits an &quot;end of file&quot; condition (EOF)</td>
</tr>
<tr>
<td>FloatingPointError</td>
<td>Raised when a floating point calculation fails</td>
</tr>
<tr>
<td>GeneratorExit</td>
<td>Raised when a generator is closed (with the close() method)</td>
</tr>
<tr>
<td>ImportError</td>
<td>Raised when an imported module does not exist</td>
</tr>
<tr>
<td>IndentationError</td>
<td>Raised when indentation is not correct</td>
</tr>
<tr>
<td>IndexError</td>
<td>Raised when an index of a sequence does not exist</td>
</tr>
<tr>
<td>KeyError</td>
<td>Raised when a key does not exist in a dictionary</td>
</tr>
<tr>
<td>KeyboardInterrupt</td>
<td>Raised when the user presses Ctrl+c, Ctrl+z or Delete</td>
</tr>
<tr>
<td>LookupError</td>
<td>Raised when errors raised can't be found</td>
</tr>
<tr>
<td>MemoryError</td>
<td>Raised when a program runs out of memory</td>
</tr>
<tr>
<td>NameError</td>
<td>Raised when a variable does not exist</td>
</tr>
<tr>
<td>NotImplementedError</td>
<td>Raised when an abstract method requires an inherited class to override the method</td>
</tr>
<tr>
<td>OSError</td>
<td>Raised when a system related operation causes an</td>
</tr>
<tr>
<td>Error</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>OverflowError</td>
<td>Raised when the result of a numeric calculation is too large</td>
</tr>
<tr>
<td>ReferenceError</td>
<td>Raised when a weak reference object does not exist</td>
</tr>
<tr>
<td>RuntimeError</td>
<td>Raised when an error occurs that do not belong to any specific exceptions</td>
</tr>
<tr>
<td>StopIteration</td>
<td>Raised when the next() method of an iterator has no further values</td>
</tr>
<tr>
<td>SyntaxError</td>
<td>Raised when a syntax error occurs</td>
</tr>
<tr>
<td>TabError</td>
<td>Raised when indentation consists of tabs or spaces</td>
</tr>
<tr>
<td>SystemError</td>
<td>Raised when a system error occurs</td>
</tr>
<tr>
<td>SystemExit</td>
<td>Raised when the sys.exit() function is called</td>
</tr>
<tr>
<td>TypeError</td>
<td>Raised when two different types are combined</td>
</tr>
<tr>
<td>UnboundLocalError</td>
<td>Raised when a local variable is referenced before assignment</td>
</tr>
<tr>
<td>UnicodeError</td>
<td>Raised when a unicode problem occurs</td>
</tr>
<tr>
<td>UnicodeEncodeError</td>
<td>Raised when a unicode encoding problem occurs</td>
</tr>
<tr>
<td>UnicodeDecodeError</td>
<td>Raised when a unicode decoding problem occurs</td>
</tr>
<tr>
<td>UnicodeTranslateError</td>
<td>Raised when a unicode translation problem occurs</td>
</tr>
<tr>
<td>ValueError</td>
<td>Raised when there is a wrong value in a specified data type</td>
</tr>
<tr>
<td>ZeroDivisionError</td>
<td>Raised when the second operator in a division is zero</td>
</tr>
</tbody>
</table>

### Python Keywords

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>and</td>
<td>A logical operator</td>
</tr>
<tr>
<td>as</td>
<td>To create an alias</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>assert</td>
<td>For debugging</td>
</tr>
<tr>
<td>break</td>
<td>To break out of a loop</td>
</tr>
<tr>
<td>class</td>
<td>To define a class</td>
</tr>
<tr>
<td>continue</td>
<td>To continue to the next iteration of a loop</td>
</tr>
<tr>
<td>def</td>
<td>To define a function</td>
</tr>
<tr>
<td>del</td>
<td>To delete an object</td>
</tr>
<tr>
<td>elif</td>
<td>Used in conditional statements, same as else if</td>
</tr>
<tr>
<td>else</td>
<td>Used in conditional statements</td>
</tr>
<tr>
<td>except</td>
<td>Used with exceptions, what to do when an exception occurs</td>
</tr>
<tr>
<td>False</td>
<td>Boolean value, result of comparison operations</td>
</tr>
<tr>
<td>finally</td>
<td>Used with exceptions, a block of code that will be executed no matter if there is an exception or not</td>
</tr>
<tr>
<td>for</td>
<td>To create a for loop</td>
</tr>
<tr>
<td>from</td>
<td>To import specific parts of a module</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>global</code></td>
<td>To declare a global variable</td>
</tr>
<tr>
<td><code>if</code></td>
<td>To make a conditional statement</td>
</tr>
<tr>
<td><code>import</code></td>
<td>To import a module</td>
</tr>
<tr>
<td><code>in</code></td>
<td>To check if a value is present in a list, tuple, etc.</td>
</tr>
<tr>
<td><code>is</code></td>
<td>To test if two variables are equal</td>
</tr>
<tr>
<td><code>lambda</code></td>
<td>To create an anonymous function</td>
</tr>
<tr>
<td><code>None</code></td>
<td>Represents a null value</td>
</tr>
<tr>
<td><code>nonlocal</code></td>
<td>To declare a non-local variable</td>
</tr>
<tr>
<td><code>not</code></td>
<td>A logical operator</td>
</tr>
<tr>
<td><code>or</code></td>
<td>A logical operator</td>
</tr>
<tr>
<td><code>pass</code></td>
<td>A null statement, a statement that will do nothing</td>
</tr>
<tr>
<td><code>raise</code></td>
<td>To raise an exception</td>
</tr>
<tr>
<td><code>return</code></td>
<td>To exit a function and return a value</td>
</tr>
<tr>
<td>True</td>
<td>Boolean value, result of comparison operations</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>try</td>
<td>To make a try...except statement</td>
</tr>
<tr>
<td>while</td>
<td>To create a while loop</td>
</tr>
<tr>
<td>with</td>
<td>Used to simplify exception handling</td>
</tr>
<tr>
<td>yield</td>
<td>To end a function, returns a generator</td>
</tr>
</tbody>
</table>

**PHP Reference**

**PHP Array Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>array()</td>
<td>Creates an array</td>
</tr>
<tr>
<td>array_change_key_case()</td>
<td>Changes all keys in an array to lowercase or uppercase</td>
</tr>
<tr>
<td>array_chunk()</td>
<td>Splits an array into chunks of arrays</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>array_column()</td>
<td>Returns the values from a single column in the input array</td>
</tr>
<tr>
<td>array_combine()</td>
<td>Creates an array by using the elements from one &quot;keys&quot; array and one &quot;values&quot; array</td>
</tr>
<tr>
<td>array_count_values()</td>
<td>Counts all the values of an array</td>
</tr>
<tr>
<td>array_diff()</td>
<td>Compare arrays, and returns the differences (compare values only)</td>
</tr>
<tr>
<td>array_diff_assoc()</td>
<td>Compare arrays, and returns the differences (compare keys and values)</td>
</tr>
<tr>
<td>array_diff_key()</td>
<td>Compare arrays, and returns the differences (compare keys only)</td>
</tr>
<tr>
<td>array_diff_uassoc()</td>
<td>Compare arrays, and returns the differences (compare keys and values, using a user-defined key comparison function)</td>
</tr>
<tr>
<td>array_diff_ukey()</td>
<td>Compare arrays, and returns the differences (compare keys only, using a user-defined key comparison function)</td>
</tr>
<tr>
<td>array_fill()</td>
<td>Fills an array with values</td>
</tr>
<tr>
<td>array_fill_keys()</td>
<td>Fills an array with values, specifying keys</td>
</tr>
<tr>
<td>array_filter()</td>
<td>Filters the values of an array using a callback</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>array_flip()</code></td>
<td>Flips/Exchanges all keys with their associated values in an array</td>
</tr>
<tr>
<td><code>array_intersect()</code></td>
<td>Compare arrays, and returns the matches (compare values only)</td>
</tr>
<tr>
<td><code>array_intersect_assoc()</code></td>
<td>Compare arrays and returns the matches (compare keys and values)</td>
</tr>
<tr>
<td><code>array_intersect_key()</code></td>
<td>Compare arrays, and returns the matches (compare keys only)</td>
</tr>
<tr>
<td><code>array_intersect_uassoc()</code></td>
<td>Compare arrays, and returns the matches (compare keys and values, using a user-defined key comparison function)</td>
</tr>
<tr>
<td><code>array_intersect_ukey()</code></td>
<td>Compare arrays, and returns the matches (compare keys only, using a user-defined key comparison function)</td>
</tr>
<tr>
<td><code>array_key_exists()</code></td>
<td>Checks if the specified key exists in the array</td>
</tr>
<tr>
<td><code>array_keys()</code></td>
<td>Returns all the keys of an array</td>
</tr>
<tr>
<td><code>array_map()</code></td>
<td>Sends each value of an array to a user-made function, which returns new values</td>
</tr>
<tr>
<td><code>array_merge()</code></td>
<td>Merges one or more arrays into one array</td>
</tr>
<tr>
<td>Function Name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>array_merge_recursive()</td>
<td>Merges one or more arrays into one array recursively</td>
</tr>
<tr>
<td>array_multisort()</td>
<td>Sorts multiple or multi-dimensional arrays</td>
</tr>
<tr>
<td>array_pad()</td>
<td>Inserts a specified number of items, with a specified value, to an array</td>
</tr>
<tr>
<td>array_pop()</td>
<td>Deletes the last element of an array</td>
</tr>
<tr>
<td>array_product()</td>
<td>Calculates the product of the values in an array</td>
</tr>
<tr>
<td>array_push()</td>
<td>Inserts one or more elements to the end of an array</td>
</tr>
<tr>
<td>array_rand()</td>
<td>Returns one or more random keys from an array</td>
</tr>
<tr>
<td>array_reduce()</td>
<td>Returns an array as a string, using a user-defined function</td>
</tr>
<tr>
<td>array_replace()</td>
<td>Replaces the values of the first array with the values from following arrays</td>
</tr>
<tr>
<td>array_replace_recursive()</td>
<td>Replaces the values of the first array with the values from following arrays recursively</td>
</tr>
<tr>
<td>array_reverse()</td>
<td>Returns an array in the reverse order</td>
</tr>
<tr>
<td>array_search()</td>
<td>Searches an array for a given value and returns the key</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>array_shift()</td>
<td>Removes the first element from an array, and returns the value of the removed element</td>
</tr>
<tr>
<td>array_slice()</td>
<td>Returns selected parts of an array</td>
</tr>
<tr>
<td>array_splice()</td>
<td>Removes and replaces specified elements of an array</td>
</tr>
<tr>
<td>array_sum()</td>
<td>Returns the sum of the values in an array</td>
</tr>
<tr>
<td>array_udiff()</td>
<td>Compare arrays, and returns the differences (compare values only, using a user-defined key comparison function)</td>
</tr>
<tr>
<td>array_udiff_assoc()</td>
<td>Compare arrays, and returns the differences (compare keys and values, using a built-in function to compare the keys and a user-defined function to compare the values)</td>
</tr>
<tr>
<td>array_udiff_uassoc()</td>
<td>Compare arrays, and returns the differences (compare keys and values, using two user-defined key comparison functions)</td>
</tr>
<tr>
<td>array_uintersect()</td>
<td>Compare arrays, and returns the matches (compare values only, using a user-defined key comparison function)</td>
</tr>
<tr>
<td>array_uintersect_assoc()</td>
<td>Compare arrays, and returns the matches (compare keys and values, using a built-in function to compare the keys and a user-defined function to compare the values)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>array_uintersect_uassoc()</td>
<td>Compare arrays, and returns the matches (compare keys and values, using two user-defined key comparison functions)</td>
</tr>
<tr>
<td>array_unique()</td>
<td>Removes duplicate values from an array</td>
</tr>
<tr>
<td>array_unshift()</td>
<td>Adds one or more elements to the beginning of an array</td>
</tr>
<tr>
<td>array_values()</td>
<td>Returns all the values of an array</td>
</tr>
<tr>
<td>array_walk()</td>
<td>Applies a user function to every member of an array</td>
</tr>
<tr>
<td>array_walk_recursive()</td>
<td>Applies a user function recursively to every member of an array</td>
</tr>
<tr>
<td>arsort()</td>
<td>Sorts an associative array in descending order, according to the value</td>
</tr>
<tr>
<td>asort()</td>
<td>Sorts an associative array in ascending order, according to the value</td>
</tr>
<tr>
<td>compact()</td>
<td>Create array containing variables and their values</td>
</tr>
<tr>
<td>count()</td>
<td>Returns the number of elements in an array</td>
</tr>
<tr>
<td>current()</td>
<td>Returns the current element in an array</td>
</tr>
<tr>
<td>each()</td>
<td>Deprecated from PHP 7.2. Returns the current key and value pair from an array</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-------------</td>
</tr>
<tr>
<td>end()</td>
<td>Sets the internal pointer of an array to its last element</td>
</tr>
<tr>
<td>extract()</td>
<td>Imports variables into the current symbol table from an array</td>
</tr>
<tr>
<td>in_array()</td>
<td>Checks if a specified value exists in an array</td>
</tr>
<tr>
<td>key()</td>
<td>Fetches a key from an array</td>
</tr>
<tr>
<td>krsort()</td>
<td>Sorts an associative array in descending order, according to the key</td>
</tr>
<tr>
<td>ksort()</td>
<td>Sorts an associative array in ascending order, according to the key</td>
</tr>
<tr>
<td>list()</td>
<td>Assigns variables as if they were an array</td>
</tr>
<tr>
<td>natcasesort()</td>
<td>Sorts an array using a case insensitive &quot;natural order&quot; algorithm</td>
</tr>
<tr>
<td>natsort()</td>
<td>Sorts an array using a &quot;natural order&quot; algorithm</td>
</tr>
<tr>
<td>next()</td>
<td>Advance the internal array pointer of an array</td>
</tr>
<tr>
<td>pos()</td>
<td>Alias of current()</td>
</tr>
<tr>
<td>prev()</td>
<td>Rewinds the internal array pointer</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>range()</td>
<td>Creates an array containing a range of elements</td>
</tr>
<tr>
<td>reset()</td>
<td>Sets the internal pointer of an array to its first element</td>
</tr>
<tr>
<td>rsort()</td>
<td>Sorts an indexed array in descending order</td>
</tr>
<tr>
<td>shuffle()</td>
<td>Shuffles an array</td>
</tr>
<tr>
<td>sizeof()</td>
<td>Alias of count()</td>
</tr>
<tr>
<td>sort()</td>
<td>Sorts an indexed array in ascending order</td>
</tr>
<tr>
<td>uasort()</td>
<td>Sorts an array by values using a user-defined comparison function</td>
</tr>
<tr>
<td>uksort()</td>
<td>Sorts an array by keys using a user-defined comparison function</td>
</tr>
<tr>
<td>usort()</td>
<td>Sorts an array using a user-defined comparison function</td>
</tr>
</tbody>
</table>

**PHP Calendar Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
</table>

576
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cal_days_in_month()</td>
<td>Returns the number of days in a month for a specified year and calendar</td>
</tr>
<tr>
<td>cal_from jd()</td>
<td>Converts a Julian Day Count into a date of a specified calendar</td>
</tr>
<tr>
<td>cal_info()</td>
<td>Returns information about a specified calendar</td>
</tr>
<tr>
<td>cal_to_jd()</td>
<td>Converts a date in a specified calendar to Julian Day Count</td>
</tr>
<tr>
<td>easter_date()</td>
<td>Returns the Unix timestamp for midnight on Easter of a specified year</td>
</tr>
<tr>
<td>easter_days()</td>
<td>Returns the number of days after March 21, that the Easter Day is in a specified year</td>
</tr>
<tr>
<td>frenchtojd()</td>
<td>Converts a French Republican date to a Julian Day Count</td>
</tr>
<tr>
<td>gregorianto jd()</td>
<td>Converts a Gregorian date to a Julian Day Count</td>
</tr>
<tr>
<td>jdayofweek()</td>
<td>Returns the day of the week</td>
</tr>
<tr>
<td>jdmonthname()</td>
<td>Returns a month name</td>
</tr>
<tr>
<td>jdtotrench()</td>
<td>Converts a Julian Day Count to a French Republican date</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>jdtogregorian()</td>
<td>Converts a Julian Day Count to a Gregorian date</td>
</tr>
<tr>
<td>jdtojewish()</td>
<td>Converts a Julian Day Count to a Jewish date</td>
</tr>
<tr>
<td>jdtojulian()</td>
<td>Converts a Julian Day Count to a Julian date</td>
</tr>
<tr>
<td>jdtounix()</td>
<td>Converts Julian Day Count to Unix timestamp</td>
</tr>
<tr>
<td>jewishtojd()</td>
<td>Converts a Jewish date to a Julian Day Count</td>
</tr>
<tr>
<td>juliantojd()</td>
<td>Converts a Julian date to a Julian Day Count</td>
</tr>
<tr>
<td>unixtojd()</td>
<td>Converts Unix timestamp to Julian Day Count</td>
</tr>
</tbody>
</table>

### PHP Predefined Calendar Constants

<table>
<thead>
<tr>
<th>Constant</th>
<th>Type</th>
<th>PHP Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAL_GREGORIAN</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_JULIAN</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_JEWISH</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_FRENCH</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>Variable</td>
<td>Type</td>
<td>PHP Version</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>CAL_NUM_CALS</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_DOW_DAYNO</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_DOW_SHORT</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_DOW_LONG</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_GREGORIAN_SHORT</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_GREGORIAN_LONG</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_JULIAN_SHORT</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_JULIAN_LONG</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_JEWISH</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_MONTH_FRENCH</td>
<td>Integer</td>
<td>PHP 4</td>
</tr>
<tr>
<td>CAL_EASTER_DEFAULT</td>
<td>Integer</td>
<td>PHP 4.3</td>
</tr>
<tr>
<td>CAL_EASTER_ROMAN</td>
<td>Integer</td>
<td>PHP 4.3</td>
</tr>
<tr>
<td>CAL_EASTER_ALWAYS_GREGORIAN</td>
<td>Integer</td>
<td>PHP 4.3</td>
</tr>
<tr>
<td>CAL_EASTER_ALWAYS_JULIAN</td>
<td>Integer</td>
<td>PHP 4.3</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>checkdate()</td>
<td>Validates a Gregorian date</td>
<td></td>
</tr>
<tr>
<td>date_add()</td>
<td>Adds days, months, years, hours, minutes, and seconds to a date</td>
<td></td>
</tr>
<tr>
<td>date_create_from_format()</td>
<td>Returns a new DateTime object formatted according to a specified format</td>
<td></td>
</tr>
<tr>
<td>date_create()</td>
<td>Returns a new DateTime object</td>
<td></td>
</tr>
<tr>
<td>date_date_set()</td>
<td>Sets a new date</td>
<td></td>
</tr>
<tr>
<td>date_default_timezone_get()</td>
<td>Returns the default timezone used by all date/time functions</td>
<td></td>
</tr>
<tr>
<td>date_default_timezone_set()</td>
<td>Sets the default timezone used by all date/time functions</td>
<td></td>
</tr>
</tbody>
</table>

**PHP Date/Time Functions**
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>e_set()</code></td>
<td></td>
</tr>
<tr>
<td><code>date_diff()</code></td>
<td>Returns the difference between two dates</td>
</tr>
<tr>
<td><code>date_format()</code></td>
<td>Returns a date formatted according to a specified format</td>
</tr>
<tr>
<td><code>date_get_last_errors()</code></td>
<td>Returns the warnings/errors found in a date string</td>
</tr>
<tr>
<td><code>date_interval_create_from_date_string()</code></td>
<td>Sets up a DateInterval from the relative parts of the string</td>
</tr>
<tr>
<td><code>date_interval_format()</code></td>
<td>Formats the interval</td>
</tr>
<tr>
<td><code>date_isodate_set()</code></td>
<td>Sets the ISO date</td>
</tr>
<tr>
<td><code>date_modify()</code></td>
<td>Modifies the timestamp</td>
</tr>
<tr>
<td><code>date_offset_get()</code></td>
<td>Returns the timezone offset</td>
</tr>
<tr>
<td><code>date_parse_from_format()</code></td>
<td>Returns an associative array with detailed info about a specified date, according to a specified format</td>
</tr>
<tr>
<td><code>date_parse()</code></td>
<td>Returns an associative array with detailed info about a specified date</td>
</tr>
<tr>
<td><code>date_sub()</code></td>
<td>Subtracts days, months, years, hours, minutes, and seconds from a date</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>date_sun_info()</td>
<td>Returns an array containing info about sunset/sunrise and twilight begin/end, for a specified day and location</td>
</tr>
<tr>
<td>date_sunrise()</td>
<td>Returns the sunrise time for a specified day and location</td>
</tr>
<tr>
<td>date_sunset()</td>
<td>Returns the sunset time for a specified day and location</td>
</tr>
<tr>
<td>date_time_set()</td>
<td>Sets the time</td>
</tr>
<tr>
<td>date_timestamp_get()</td>
<td>Returns the Unix timestamp</td>
</tr>
<tr>
<td>date_timestamp_set()</td>
<td>Sets the date and time based on a Unix timestamp</td>
</tr>
<tr>
<td>date_timezone_get()</td>
<td>Returns the time zone of the given DateTime object</td>
</tr>
<tr>
<td>date_timezone_set()</td>
<td>Sets the time zone for the DateTime object</td>
</tr>
<tr>
<td>date()</td>
<td>Formats a local date and time</td>
</tr>
<tr>
<td>getdate()</td>
<td>Returns date/time information of a timestamp or the current local date/time</td>
</tr>
<tr>
<td>gettimeofday()</td>
<td>Returns the current time</td>
</tr>
<tr>
<td>gmdate()</td>
<td>Formats a GMT/UTC date and time</td>
</tr>
<tr>
<td>gmmktime()</td>
<td>Returns the Unix timestamp for a GMT date</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>gmstrftime()</td>
<td>Formats a GMT/UTC date and time according to locale settings</td>
</tr>
<tr>
<td>strftime()</td>
<td>Formats a local time and/or date according to locale settings</td>
</tr>
<tr>
<td>strftime()</td>
<td>Parsed a time/date generated with strftime()</td>
</tr>
<tr>
<td>localtime()</td>
<td>Returns the local time</td>
</tr>
<tr>
<td>microtime()</td>
<td>Returns the current Unix timestamp with microseconds</td>
</tr>
<tr>
<td>mktime()</td>
<td>Returns the Unix timestamp for a date</td>
</tr>
<tr>
<td>mktime()</td>
<td>Parsed a time/date generated with strftime()</td>
</tr>
<tr>
<td>mktime()</td>
<td>Parses an English textual datetime into a Unix timestamp</td>
</tr>
<tr>
<td>strftime()</td>
<td>Returns the current time as a Unix timestamp</td>
</tr>
<tr>
<td>strftime()</td>
<td>Returns an associative array containing dst, offset, and the timezone name</td>
</tr>
<tr>
<td>strftime()</td>
<td>Returns an indexed array with all timezone identifiers</td>
</tr>
<tr>
<td>strftime()</td>
<td>Returns location information for a specified timezone</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>timezone_name_from_abbr()</td>
<td>Returns the timezone name from abbreviation</td>
</tr>
<tr>
<td>timezone_name_get()</td>
<td>Returns the name of the timezone</td>
</tr>
<tr>
<td>timezone_offset_get()</td>
<td>Returns the timezone offset from GMT</td>
</tr>
<tr>
<td>timezone_open()</td>
<td>Creates new DateTimeZone object</td>
</tr>
<tr>
<td>timezone_transitions_get()</td>
<td>Returns all transitions for the timezone</td>
</tr>
<tr>
<td>timezone_version_get()</td>
<td>Returns the version of the timezonedb</td>
</tr>
</tbody>
</table>

**PHP Predefined Date/Time Constants**

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE_ATOM</td>
<td>Atom (example: 2019-01-18T14:13:03+00:00)</td>
</tr>
<tr>
<td>DATE_COOKIE</td>
<td>HTTP Cookies (example: Fri, 18 Jan 2019 14:13:03 UTC)</td>
</tr>
<tr>
<td>DATE_ISO8601</td>
<td>ISO-8601 (example: 2019-01-18T14:13:03+0000)</td>
</tr>
<tr>
<td>DATE_RFC822</td>
<td>RFC 822 (example: Fri, 18 Jan 2019 14:13:03 +0000)</td>
</tr>
<tr>
<td>DATE_RFC850</td>
<td>RFC 850 (example: Friday, 18-Jan-19 14:13:03 UTC)</td>
</tr>
<tr>
<td>--------------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>DATE_RFC1036</td>
<td>RFC 1036 (example: Friday, 18-Jan-19 14:13:03 +0000)</td>
</tr>
<tr>
<td>DATE_RFC1123</td>
<td>RFC 1123 (example: Fri, 18 Jan 2019 14:13:03 +0000)</td>
</tr>
<tr>
<td>DATE_RFC2822</td>
<td>RFC 2822 (example: Fri, 18 Jan 2019 14:13:03 +0000)</td>
</tr>
<tr>
<td>DATE_RFC3339</td>
<td>Same as DATE_ATOM (since PHP 5.1.3)</td>
</tr>
<tr>
<td>DATE_RFC3339_EXTENDED</td>
<td>RFC3339 Extended format (since PHP 7.0.0) (example: 2019-01-18T16:34:01.000+00:00)</td>
</tr>
<tr>
<td>DATE_RSS</td>
<td>RSS (Fri, 18 Jan 2019 14:13:03 +0000)</td>
</tr>
<tr>
<td>DATE_W3C</td>
<td>World Wide Web Consortium (example: 2019-01-18T14:13:03+00:00)</td>
</tr>
<tr>
<td>SUNFUNCS_RET_TIMESTAMP</td>
<td>Timestamp (since PHP 5.1.2)</td>
</tr>
<tr>
<td>SUNFUNCS_RET_STRING</td>
<td>Hours:minutes (example: 09:41) (since PHP 5.1.2)</td>
</tr>
<tr>
<td>SUNFUNCS_RET_DOUBLE</td>
<td>Hours as a floating point number (example: 9.75) (since PHP 5.1.2)</td>
</tr>
</tbody>
</table>

**PHP Directory Functions**
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>chdir()</td>
<td>Changes the current directory</td>
</tr>
<tr>
<td>chroot()</td>
<td>Changes the root directory</td>
</tr>
<tr>
<td>closedir()</td>
<td>Closes a directory handle</td>
</tr>
<tr>
<td>dir()</td>
<td>Returns an instance of the Directory class</td>
</tr>
<tr>
<td>getcwd()</td>
<td>Returns the current working directory</td>
</tr>
<tr>
<td>opendir()</td>
<td>Opens a directory handle</td>
</tr>
<tr>
<td>readdir()</td>
<td>Returns an entry from a directory handle</td>
</tr>
<tr>
<td>rewinddir()</td>
<td>Resets a directory handle</td>
</tr>
<tr>
<td>scandir()</td>
<td>Returns an array of files and directories of a specified directory</td>
</tr>
</tbody>
</table>

**PHP Error and Logging Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>debug_backtrace()</td>
<td>Generates a backtrace</td>
</tr>
<tr>
<td>debug_print_backtrace()</td>
<td>Prints a backtrace</td>
</tr>
<tr>
<td>error_clear_last()</td>
<td>Clears the last error</td>
</tr>
<tr>
<td>error_get_last()</td>
<td>Returns the last error that occurred</td>
</tr>
<tr>
<td>error_log()</td>
<td>Sends an error message to a log, to a file, or to a mail account</td>
</tr>
<tr>
<td>error_reporting()</td>
<td>Specifies which errors are reported</td>
</tr>
<tr>
<td>restore_error_handler()</td>
<td>Restores the previous error handler</td>
</tr>
<tr>
<td>restore_exception_handler()</td>
<td>Restores the previous exception handler</td>
</tr>
<tr>
<td>set_error_handler()</td>
<td>Sets a user-defined error handler function</td>
</tr>
<tr>
<td>set_exception_handler()</td>
<td>Sets a user-defined exception handler function</td>
</tr>
<tr>
<td>trigger_error()</td>
<td>Creates a user-level error message</td>
</tr>
<tr>
<td>user_error()</td>
<td>Alias of trigger_error()</td>
</tr>
</tbody>
</table>
## PHP Predefined Error and Logging Constants

<table>
<thead>
<tr>
<th>Value</th>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>E_ERROR</td>
<td>Fatal run-time errors. Errors that cannot be recovered from. Execution of the script is halted</td>
</tr>
<tr>
<td>2</td>
<td>E_WARNING</td>
<td>Run-time warnings (non-fatal errors). Execution of the script is not halted</td>
</tr>
<tr>
<td>4</td>
<td>EPARSE</td>
<td>Compile-time parse errors. Parse errors should only be generated by the parser</td>
</tr>
<tr>
<td>8</td>
<td>E_NOTICE</td>
<td>Run-time notices. The script found something that might be an error, but could also happen when running a script normally</td>
</tr>
<tr>
<td>16</td>
<td>E_CORE_ERROR</td>
<td>Fatal errors at PHP startup. This is like E_ERROR, except it is generated by the core of PHP</td>
</tr>
<tr>
<td>32</td>
<td>E_CORE_WARNING</td>
<td>Non-fatal errors at PHP startup. This is like E_WARNING, except it is generated by the core of PHP</td>
</tr>
<tr>
<td>64</td>
<td>E_COMPILE_ERROR</td>
<td>Fatal compile-time errors. This is like E_ERROR, except it is generated by the Zend Scripting Engine</td>
</tr>
<tr>
<td>Code</td>
<td>Constant</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>128</td>
<td>E_COMPILE_WARNING</td>
<td>Non-fatal compile-time errors. This is like E_WARNING, except it is generated by the Zend Scripting Engine</td>
</tr>
<tr>
<td>256</td>
<td>E_USER_ERROR</td>
<td>Fatal user-generated error. This is like E_ERROR, except it is generated in PHP code by using the PHP function trigger_error()</td>
</tr>
<tr>
<td>512</td>
<td>E_USER_WARNING</td>
<td>Non-fatal user-generated warning. This is like E_WARNING, except it is generated in PHP code by using the PHP function trigger_error()</td>
</tr>
<tr>
<td>1024</td>
<td>E_USER_NOTICE</td>
<td>User-generated notice. This is like E_NOTICE, except it is generated in PHP code by using the PHP function trigger_error()</td>
</tr>
<tr>
<td>2048</td>
<td>E_STRICT</td>
<td>Enable to have PHP suggest changes to your code which will ensure the best interoperability and forward compatibility of your code (Since PHP 5 but not included in E_ALL until PHP 5.4)</td>
</tr>
<tr>
<td>4096</td>
<td>E_RECOVERABLE_ERROR</td>
<td>Catchable fatal error. Indicates that a probably dangerous error occurred, but did not leave the Engine in an unstable state. If the error is not caught by a user defined handle, the application aborts as it was an E_ERROR (Since PHP 5.2)</td>
</tr>
<tr>
<td>8192</td>
<td>E_DEPRECATED</td>
<td>Run-time notices. Enable this to receive warnings about code that will not work in future versions (Since PHP 5.3)</td>
</tr>
</tbody>
</table>
User-generated warning message. This is like E_DEPRECATED, except it is generated in PHP code by using the PHP function trigger_error() (Since PHP 5.3)

Enable all PHP errors and warnings (except E_STRICT in versions < 5.4)

**PHP File system Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>basename()</td>
<td>Returns the filename component of a path</td>
</tr>
<tr>
<td>chgrp()</td>
<td>Changes the file group</td>
</tr>
<tr>
<td>chmod()</td>
<td>Changes the file mode</td>
</tr>
<tr>
<td>chown()</td>
<td>Changes the file owner</td>
</tr>
<tr>
<td>clearstatcache()</td>
<td>Clears the file status cache</td>
</tr>
<tr>
<td>copy()</td>
<td>Copies a file</td>
</tr>
<tr>
<td>delete()</td>
<td>See unlink()</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>dirname()</td>
<td>Returns the directory name component of a path</td>
</tr>
<tr>
<td>disk_free_space()</td>
<td>Returns the free space of a filesystem or disk</td>
</tr>
<tr>
<td>disk_total_space()</td>
<td>Returns the total size of a filesystem or disk</td>
</tr>
<tr>
<td>diskfreespace()</td>
<td>Alias of disk_free_space()</td>
</tr>
<tr>
<td>fclose()</td>
<td>Closes an open file</td>
</tr>
<tr>
<td>feof()</td>
<td>Checks if the &quot;end-of-file&quot; (EOF) has been reached for an open file</td>
</tr>
<tr>
<td>fflush()</td>
<td>Flushes buffered output to an open file</td>
</tr>
<tr>
<td>fgetc()</td>
<td>Returns a single character from an open file</td>
</tr>
<tr>
<td>fgetcsv()</td>
<td>Returns a line from an open CSV file</td>
</tr>
<tr>
<td>fgets()</td>
<td>Returns a line from an open file</td>
</tr>
<tr>
<td>fgetss()</td>
<td>Deprecated from PHP 7.3. Returns a line from an open file - stripped from HTML and PHP tags</td>
</tr>
<tr>
<td>file()</td>
<td>Reads a file into an array</td>
</tr>
<tr>
<td>file_exists()</td>
<td>Checks whether or not a file or directory exists</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>file_get_contents()</code></td>
<td>Reads a file into a string</td>
</tr>
<tr>
<td><code>file_put_contents()</code></td>
<td>Writes data to a file</td>
</tr>
<tr>
<td><code>fileatime()</code></td>
<td>Returns the last access time of a file</td>
</tr>
<tr>
<td><code>filectime()</code></td>
<td>Returns the last change time of a file</td>
</tr>
<tr>
<td><code>filegroup()</code></td>
<td>Returns the group ID of a file</td>
</tr>
<tr>
<td><code>fileinode()</code></td>
<td>Returns the inode number of a file</td>
</tr>
<tr>
<td><code>filemtime()</code></td>
<td>Returns the last modification time of a file</td>
</tr>
<tr>
<td><code>fileowner()</code></td>
<td>Returns the user ID (owner) of a file</td>
</tr>
<tr>
<td><code>fileperms()</code></td>
<td>Returns the file's permissions</td>
</tr>
<tr>
<td><code>filesize()</code></td>
<td>Returns the file size</td>
</tr>
<tr>
<td><code>filetype()</code></td>
<td>Returns the file type</td>
</tr>
<tr>
<td><code>flock()</code></td>
<td>Locks or releases a file</td>
</tr>
<tr>
<td><code>fnmatch()</code></td>
<td>Matches a filename or string against a specified pattern</td>
</tr>
<tr>
<td><code>fopen()</code></td>
<td>Opens a file or URL</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>fpassthru()</td>
<td>Reads from the current position in a file - until EOF, and writes the result to the output buffer</td>
</tr>
<tr>
<td>fputcsv()</td>
<td>Formats a line as CSV and writes it to an open file</td>
</tr>
<tr>
<td>fputs()</td>
<td>Alias of fwrite()</td>
</tr>
<tr>
<td>fread()</td>
<td>Reads from an open file (binary-safe)</td>
</tr>
<tr>
<td>fscanf()</td>
<td>Parses input from an open file according to a specified format</td>
</tr>
<tr>
<td>fseek()</td>
<td>Seeks in an open file</td>
</tr>
<tr>
<td>fstat()</td>
<td>Returns information about an open file</td>
</tr>
<tr>
<td>ftell()</td>
<td>Returns the current position in an open file</td>
</tr>
<tr>
<td>ftruncate()</td>
<td>Truncates an open file to a specified length</td>
</tr>
<tr>
<td>fwrite()</td>
<td>Writes to an open file (binary-safe)</td>
</tr>
<tr>
<td>glob()</td>
<td>Returns an array of filenames / directories matching a specified pattern</td>
</tr>
<tr>
<td>is_dir()</td>
<td>Checks whether a file is a directory</td>
</tr>
<tr>
<td>is_executable()</td>
<td>Checks whether a file is executable</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>is_file()</code></td>
<td>Checks whether a file is a regular file</td>
</tr>
<tr>
<td><code>is_link()</code></td>
<td>Checks whether a file is a link</td>
</tr>
<tr>
<td><code>is_readable()</code></td>
<td>Checks whether a file is readable</td>
</tr>
<tr>
<td><code>is_uploaded_file()</code></td>
<td>Checks whether a file was uploaded via HTTP POST</td>
</tr>
<tr>
<td><code>is_writable()</code></td>
<td>Checks whether a file is writable</td>
</tr>
<tr>
<td><code>is_writeable()</code></td>
<td>Alias of <code>is_writable()</code></td>
</tr>
<tr>
<td><code>lchgrp()</code></td>
<td>Changes the group ownership of a symbolic link</td>
</tr>
<tr>
<td><code>lchown()</code></td>
<td>Changes the user ownership of a symbolic link</td>
</tr>
<tr>
<td><code>link()</code></td>
<td>Creates a hard link</td>
</tr>
<tr>
<td><code>linkinfo()</code></td>
<td>Returns information about a hard link</td>
</tr>
<tr>
<td><code>lstat()</code></td>
<td>Returns information about a file or symbolic link</td>
</tr>
<tr>
<td><code>mkdir()</code></td>
<td>Creates a directory</td>
</tr>
<tr>
<td><code>move_uploaded_file()</code></td>
<td>Moves an uploaded file to a new location</td>
</tr>
<tr>
<td><code>parse_ini_file()</code></td>
<td>Parses a configuration file</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>parse_ini_string()</td>
<td>Parses a configuration string</td>
</tr>
<tr>
<td>pathinfo()</td>
<td>Returns information about a file path</td>
</tr>
<tr>
<td>pclose()</td>
<td>Closes a pipe opened by popen()</td>
</tr>
<tr>
<td>popen()</td>
<td>Opens a pipe</td>
</tr>
<tr>
<td>readline()</td>
<td>Reads a file and writes it to the output buffer</td>
</tr>
<tr>
<td>readlink()</td>
<td>Returns the target of a symbolic link</td>
</tr>
<tr>
<td>realpath()</td>
<td>Returns the absolute pathname</td>
</tr>
<tr>
<td>realpath_cache_get()</td>
<td>Returns realpath cache entries</td>
</tr>
<tr>
<td>realpath_cache_size()</td>
<td>Returns realpath cache size</td>
</tr>
<tr>
<td>rename()</td>
<td>Renames a file or directory</td>
</tr>
<tr>
<td>rewind()</td>
<td>Rewinds a file pointer</td>
</tr>
<tr>
<td>rmdir()</td>
<td>Removes an empty directory</td>
</tr>
<tr>
<td>set_file_buffer()</td>
<td>Alias of stream_set_write_buffer(). Sets the buffer size for write operations on the given file</td>
</tr>
<tr>
<td>stat()</td>
<td>Returns information about a file</td>
</tr>
</tbody>
</table>
### Function Table

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>symlink()</td>
<td>Creates a symbolic link</td>
</tr>
<tr>
<td>tempnam()</td>
<td>Creates a unique temporary file</td>
</tr>
<tr>
<td>tmpfile()</td>
<td>Creates a unique temporary file</td>
</tr>
<tr>
<td>touch()</td>
<td>Sets access and modification time of a file</td>
</tr>
<tr>
<td>umask()</td>
<td>Changes file permissions for files</td>
</tr>
<tr>
<td>unlink()</td>
<td>Deletes a file</td>
</tr>
</tbody>
</table>

### PHP FTP Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ftp_alloc()</td>
<td>Allocates space for a file to be uploaded to the FTP server</td>
</tr>
<tr>
<td>ftp_cdup()</td>
<td>Changes to the parent directory on the FTP server</td>
</tr>
<tr>
<td>ftp_chdir()</td>
<td>Changes the current directory on the FTP server</td>
</tr>
<tr>
<td>ftp_chmod()</td>
<td>Sets permissions on a file via FTP</td>
</tr>
<tr>
<td>ftp_close()</td>
<td>Closes an FTP connection</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ftp_connect()</td>
<td>Opens an FTP connection</td>
</tr>
<tr>
<td>ftp_delete()</td>
<td>Deletes a file on the FTP server</td>
</tr>
<tr>
<td>ftp_exec()</td>
<td>Executes a command on the FTP server</td>
</tr>
<tr>
<td>ftp_fget()</td>
<td>Downloads a file from the FTP server and saves it into an open local file</td>
</tr>
<tr>
<td>ftp_fput()</td>
<td>Uploads from an open file and saves it to a file on the FTP server</td>
</tr>
<tr>
<td>ftp_get()</td>
<td>Downloads a file from the FTP server</td>
</tr>
<tr>
<td>ftp_get_option()</td>
<td>Returns runtime options of the FTP connection</td>
</tr>
<tr>
<td>ftp_login()</td>
<td>Logs in to the FTP connection</td>
</tr>
<tr>
<td>ftp_mdtm()</td>
<td>Returns the last modified time of a specified file</td>
</tr>
<tr>
<td>ftp_mkdir()</td>
<td>Creates a new directory on the FTP server</td>
</tr>
<tr>
<td>ftp_mlsd()</td>
<td>Returns the list of files in the specified directory</td>
</tr>
<tr>
<td>ftp_nb_continue()</td>
<td>Continues retrieving/sending a file (non-blocking)</td>
</tr>
<tr>
<td>ftp_nb_fget()</td>
<td>Downloads a file from the FTP server and saves it into an open file (non-blocking)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ftp_nb_fput()</td>
<td>Uploads from an open file and saves it to a file on the FTP server (non-blocking)</td>
</tr>
<tr>
<td>ftp_nb_get()</td>
<td>Downloads a file from the FTP server (non-blocking)</td>
</tr>
<tr>
<td>ftp_nb_put()</td>
<td>Uploads a file to the FTP server (non-blocking)</td>
</tr>
<tr>
<td>ftp_nlist()</td>
<td>Returns a list of files in the specified directory on the FTP server</td>
</tr>
<tr>
<td>ftp_pasv()</td>
<td>Turns passive mode on or off</td>
</tr>
<tr>
<td>ftp_put()</td>
<td>Uploads a file to the FTP server</td>
</tr>
<tr>
<td>ftp_pwd()</td>
<td>Returns the current directory name</td>
</tr>
<tr>
<td>ftp_quit()</td>
<td>Alias of ftp_close()</td>
</tr>
<tr>
<td>ftp_raw()</td>
<td>Sends a raw command to the FTP server</td>
</tr>
<tr>
<td>ftp_rawlist()</td>
<td>Returns a list of files with file information from a specified directory</td>
</tr>
<tr>
<td>ftp_rename()</td>
<td>Renames a file or directory on the FTP server</td>
</tr>
<tr>
<td>ftp_rmdir()</td>
<td>Deletes an empty directory on the FTP server</td>
</tr>
<tr>
<td>ftp_set_option()</td>
<td>Sets runtime options for the FTP connection</td>
</tr>
</tbody>
</table>
ftp_site() Sends an FTP SITE command to the FTP server
ftp_size() Returns the size of the specified file
ftp_ssl_connect() Opens a secure SSL-FTP connection
ftp_systype() Returns the system type identifier of the FTP server

**PHP Predefined FTP Constants**

<table>
<thead>
<tr>
<th>Constant</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FTP_ASCII</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>FTP_AUTOSEEK</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>FTP_AUTORESUME</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>FTP_BINARY</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>FTP_FAILED</td>
<td>Integer</td>
<td>Asynchronous transfer has failed</td>
</tr>
<tr>
<td>FTP_FINISHED</td>
<td>Integer</td>
<td>Asynchronous transfer is completed</td>
</tr>
<tr>
<td>FTP_IMAGE</td>
<td>Integer</td>
<td>Alias of FTP_BINARY</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>FTP_MOREDATA</td>
<td>Integer</td>
<td>Asynchronous transfer is in progress</td>
</tr>
<tr>
<td>FTP_TEXT</td>
<td>Integer</td>
<td>Alias of FTP_ASCII</td>
</tr>
<tr>
<td>FTP_TIMEOUT_SEC</td>
<td>Integer</td>
<td>The timeout used for network operations</td>
</tr>
<tr>
<td>FTP_USEPASVADDRESS</td>
<td>Boolean</td>
<td></td>
</tr>
</tbody>
</table>

**PHP Filter Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>filter_has_var()</td>
<td>Checks whether a variable of a specified input type exist</td>
</tr>
<tr>
<td>filter_id()</td>
<td>Returns the filter ID of a specified filter name</td>
</tr>
<tr>
<td>filter_input()</td>
<td>Gets an external variable (e.g. from form input) and optionally filters it</td>
</tr>
<tr>
<td>filter_input_array()</td>
<td>Gets external variables (e.g. from form input) and optionally filters them</td>
</tr>
<tr>
<td>filter_list()</td>
<td>Returns a list of all supported filter names</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>filter_var()</td>
<td>Filters a variable with a specified filter</td>
</tr>
<tr>
<td>filter_var_array()</td>
<td>Gets multiple variables and filter them</td>
</tr>
</tbody>
</table>

**PHP libxml Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>libxml_clear_errors()</td>
<td>Clears the libxml error buffer</td>
</tr>
<tr>
<td>libxml_disable_entity_loader()</td>
<td>Enables the ability to load external entities</td>
</tr>
<tr>
<td>libxml_get_errors()</td>
<td>Gets the errors from the the libxml error buffer</td>
</tr>
<tr>
<td>libxml_get_last_error()</td>
<td>Gets the last error from the the libxml error buffer</td>
</tr>
<tr>
<td>libxml_set_external_entity_loader()</td>
<td>Changes the default external entity loader</td>
</tr>
<tr>
<td>libxml_set_streams_context()</td>
<td>Sets the streams context for the next libxml document load or write</td>
</tr>
<tr>
<td>libxml_use_internal_errors()</td>
<td>Disables the standard libxml errors and enables user error handling</td>
</tr>
</tbody>
</table>
### PHP Predefined libxml Constants

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIBXML_BIGLINES</td>
<td>Make line numbers greater than 65535 to be reported correctly</td>
</tr>
<tr>
<td>LIBXML_COMPACT</td>
<td>Set small nodes allocation optimization. This may improve the application performance</td>
</tr>
<tr>
<td>LIBXML_DTDATTR</td>
<td>Set default DTD attributes</td>
</tr>
<tr>
<td>LIBXML_DTDLOAD</td>
<td>Load external subset</td>
</tr>
<tr>
<td>LIBXMLDTDVALID</td>
<td>Validate with the DTD</td>
</tr>
<tr>
<td>LIBXML_HTML_NOIMPLIED</td>
<td>Set HTML_PARSE_NOIMPLIED flag. This turns off automatic adding of implied html/body elements</td>
</tr>
<tr>
<td>LIBXML_HTML_NODEFDTD</td>
<td>Set HTML_PARSE_NODEFDTD flag. This prevents a default doctype to be added, if no doctype is found</td>
</tr>
<tr>
<td>LIBXML_NOBLANKS</td>
<td>Remove blank nodes</td>
</tr>
<tr>
<td>LIBXML_NOCDATA</td>
<td>Set CDATA as text nodes</td>
</tr>
<tr>
<td>Macro</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LIBXML_NOEMPTYTAG</td>
<td>Change empty tags (e.g. <code>&lt;br/&gt;</code> to <code>&lt;br&gt;&lt;/br&gt;</code>), only available in the DOMDocument-&gt;save() and DOMDocument-&gt;saveXML() functions</td>
</tr>
<tr>
<td>LIBXML_NOENT</td>
<td>Substitute entities</td>
</tr>
<tr>
<td>LIBXML_NOERROR</td>
<td>Do not show error reports</td>
</tr>
<tr>
<td>LIBXML_NONET</td>
<td>Stop network access while loading documents</td>
</tr>
<tr>
<td>LIBXML_NOWARNING</td>
<td>Do not show warning reports</td>
</tr>
<tr>
<td>LIBXML_NOXMLDECL</td>
<td>Drop the XML declaration when saving a document</td>
</tr>
<tr>
<td>LIBXML_NSCLEAN</td>
<td>Remove excess namespace declarations</td>
</tr>
<tr>
<td>LIBXML_PARSEHUGE</td>
<td>Set XMLPARSE_HUGE flag. This relaxes any hardcoded limit from the parser, such as maximum depth of a document or the size of text nodes</td>
</tr>
<tr>
<td>LIBXML_PEDANTIC</td>
<td>Set XMLPARSE_PEDANTIC flag. This enables pedantic error reporting</td>
</tr>
<tr>
<td>LIBXML_XINCLUDE</td>
<td>Use XInclude substitution</td>
</tr>
<tr>
<td>LIBXML_ERR_ERROR</td>
<td>Get recoverable errors</td>
</tr>
<tr>
<td>LIBXML_ERR_FATAL</td>
<td>Get fatal errors</td>
</tr>
</tbody>
</table>
LIBXML_ERR_NONE | Get no errors
LIBXML_ERR_WARNING | Get simple warnings
LIBXML_VERSION | Get libxml version (e.g. 20605 or 20617)
LIBXML_DOTTED_VERSION | Get dotted libxml version (e.g. 2.6.5 or 2.6.17)
LIBXML_SCHEMA_CREATE | Create default or fixed value nodes during XSD schema validation

**PHP Mail Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ezmlm_hash()</td>
<td>Calculates the hash value needed by EZMLM</td>
</tr>
<tr>
<td>mail()</td>
<td>Allows you to send emails directly from a script</td>
</tr>
</tbody>
</table>

**PHP Math Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs()</td>
<td>Returns the absolute (positive) value of a number</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>acos()</td>
<td>Returns the arc cosine of a number</td>
</tr>
<tr>
<td>acosh()</td>
<td>Returns the inverse hyperbolic cosine of a number</td>
</tr>
<tr>
<td>asin()</td>
<td>Returns the arc sine of a number</td>
</tr>
<tr>
<td>asinh()</td>
<td>Returns the inverse hyperbolic sine of a number</td>
</tr>
<tr>
<td>atan()</td>
<td>Returns the arc tangent of a number in radians</td>
</tr>
<tr>
<td>atan2()</td>
<td>Returns the arc tangent of two variables x and y</td>
</tr>
<tr>
<td>atanh()</td>
<td>Returns the inverse hyperbolic tangent of a number</td>
</tr>
<tr>
<td>base_convert()</td>
<td>Converts a number from one number base to another</td>
</tr>
<tr>
<td>bindec()</td>
<td>Converts a binary number to a decimal number</td>
</tr>
<tr>
<td>ceil()</td>
<td>Rounds a number up to the nearest integer</td>
</tr>
<tr>
<td>cos()</td>
<td>Returns the cosine of a number</td>
</tr>
<tr>
<td>cosh()</td>
<td>Returns the hyperbolic cosine of a number</td>
</tr>
<tr>
<td>dechex()</td>
<td>Converts a decimal number to a hexadecimal number</td>
</tr>
<tr>
<td>decbin()</td>
<td>Converts a decimal number to a binary number</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>decoct()</td>
<td>Converts a decimal number to an octal number</td>
</tr>
<tr>
<td>deg2rad()</td>
<td>Converts a degree value to a radian value</td>
</tr>
<tr>
<td>exp()</td>
<td>Calculates the exponent of e</td>
</tr>
<tr>
<td>expm1()</td>
<td>Returns exp(x) - 1</td>
</tr>
<tr>
<td>floor()</td>
<td>Rounds a number down to the nearest integer</td>
</tr>
<tr>
<td>fmod()</td>
<td>Returns the remainder of x/y</td>
</tr>
<tr>
<td>getrandmax()</td>
<td>Returns the largest possible value returned by rand()</td>
</tr>
<tr>
<td>hexdec()</td>
<td>Converts a hexadecimal number to a decimal number</td>
</tr>
<tr>
<td>hypot()</td>
<td>Calculates the hypotenuse of a right-angle triangle</td>
</tr>
<tr>
<td>intdiv()</td>
<td>Performs integer division</td>
</tr>
<tr>
<td>is_finite()</td>
<td>Checks whether a value is finite or not</td>
</tr>
<tr>
<td>is_infinite()</td>
<td>Checks whether a value is infinite or not</td>
</tr>
<tr>
<td>is_nan()</td>
<td>Checks whether a value is 'not-a-number'</td>
</tr>
<tr>
<td>lcg_value()</td>
<td>Returns a pseudo random number in a range between 0 and 1</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>log()</td>
<td>Returns the natural logarithm of a number</td>
</tr>
<tr>
<td>log10()</td>
<td>Returns the base-10 logarithm of a number</td>
</tr>
<tr>
<td>log1p()</td>
<td>Returns log(1+number)</td>
</tr>
<tr>
<td>max()</td>
<td>Returns the highest value in an array, or the highest value of several specified values</td>
</tr>
<tr>
<td>min()</td>
<td>Returns the lowest value in an array, or the lowest value of several specified values</td>
</tr>
<tr>
<td>mt_getrandmax()</td>
<td>Returns the largest possible value returned by mt_rand()</td>
</tr>
<tr>
<td>mt_rand()</td>
<td>Generates a random integer using Mersenne Twister algorithm</td>
</tr>
<tr>
<td>mt_srand()</td>
<td>Seeds the Mersenne Twister random number generator</td>
</tr>
<tr>
<td>octdec()</td>
<td>Converts an octal number to a decimal number</td>
</tr>
<tr>
<td>pi()</td>
<td>Returns the value of PI</td>
</tr>
<tr>
<td>pow()</td>
<td>Returns x raised to the power of y</td>
</tr>
<tr>
<td>rad2deg()</td>
<td>Converts a radian value to a degree value</td>
</tr>
<tr>
<td>rand()</td>
<td>Generates a random integer</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>round()</code></td>
<td>Rounds a floating-point number</td>
</tr>
<tr>
<td><code>sin()</code></td>
<td>Returns the sine of a number</td>
</tr>
<tr>
<td><code>sinh()</code></td>
<td>Returns the hyperbolic sine of a number</td>
</tr>
<tr>
<td><code>sqrt()</code></td>
<td>Returns the square root of a number</td>
</tr>
<tr>
<td><code>srand()</code></td>
<td>Seeds the random number generator</td>
</tr>
<tr>
<td><code>tan()</code></td>
<td>Returns the tangent of a number</td>
</tr>
<tr>
<td><code>tanh()</code></td>
<td>Returns the hyperbolic tangent of a number</td>
</tr>
</tbody>
</table>

### PHP Predefined Math Constants

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INF</td>
<td>INF</td>
<td>The infinite</td>
</tr>
<tr>
<td>M_E</td>
<td>2.7182818284590452354</td>
<td>Returns e</td>
</tr>
<tr>
<td>M_EULER</td>
<td>0.57721566490153286061</td>
<td>Returns Euler constant</td>
</tr>
<tr>
<td>Symbol</td>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>M_LNPI</td>
<td>1.14472988584940017414</td>
<td>Returns the natural logarithm of PI: ( \log_e(\pi) )</td>
</tr>
<tr>
<td>M_LN2</td>
<td>0.69314718055994530942</td>
<td>Returns the natural logarithm of 2: ( \log_e 2 )</td>
</tr>
<tr>
<td>M_LN10</td>
<td>2.30258509299404568402</td>
<td>Returns the natural logarithm of 10: ( \log_e 10 )</td>
</tr>
<tr>
<td>M_LOG2E</td>
<td>1.4426950408889634074</td>
<td>Returns the base-2 logarithm of E: ( \log_2 e )</td>
</tr>
<tr>
<td>M_LOG10E</td>
<td>0.43429448190325182765</td>
<td>Returns the base-10 logarithm of E: ( \log_{10} e )</td>
</tr>
<tr>
<td>M_PI</td>
<td>3.14159265358979323846</td>
<td>Returns Pi</td>
</tr>
<tr>
<td>M_PI_2</td>
<td>1.57079632679489661923</td>
<td>Returns ( \pi/2 )</td>
</tr>
<tr>
<td>M_PI_4</td>
<td>0.78539816339744830962</td>
<td>Returns ( \pi/4 )</td>
</tr>
<tr>
<td>M_1_PI</td>
<td>0.31830988618379067154</td>
<td>Returns 1/Pi</td>
</tr>
<tr>
<td>M_2_PI</td>
<td>0.63661977236758134308</td>
<td>Returns 2/Pi</td>
</tr>
<tr>
<td>M_SQRTPI</td>
<td>1.77245385090551602729</td>
<td>Returns the square root of PI: ( \sqrt{\pi} )</td>
</tr>
<tr>
<td>M_2_SQRTPI</td>
<td>1.12837916709551257390</td>
<td>Returns 2/square root of PI: ( 2/\sqrt{\pi} )</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>--------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>2/sqrt(pi)</td>
<td>Returns the square root of 1/2: 1/sqrt(2)</td>
<td></td>
</tr>
<tr>
<td>M_SQRT1_2</td>
<td>0.70710678118654752440</td>
<td></td>
</tr>
<tr>
<td>M_SQRT2</td>
<td>1.41421356237309504880</td>
<td></td>
</tr>
<tr>
<td>M_SQRT3</td>
<td>1.73205080756887729352</td>
<td></td>
</tr>
<tr>
<td>NAN</td>
<td>NAN</td>
<td></td>
</tr>
<tr>
<td>PHP_ROUND_HALF_UP</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PHP_ROUND_HALF_DOWN</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>PHP_ROUND_HALF_EVEN</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>PHP_ROUND_HALF_ODD</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

**PHP Miscellaneous Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connection_aborted()</td>
<td>Checks whether the client has disconnected</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>connection_status()</td>
<td>Returns the current connection status</td>
</tr>
<tr>
<td>connection_timeout()</td>
<td>Deprecated from PHP 4.0.5. Checks whether the script has timed out</td>
</tr>
<tr>
<td>constant()</td>
<td>Returns the value of a constant</td>
</tr>
<tr>
<td>define()</td>
<td>Defines a constant</td>
</tr>
<tr>
<td>defined()</td>
<td>Checks whether a constant exists</td>
</tr>
<tr>
<td>die()</td>
<td>Alias of exit()</td>
</tr>
<tr>
<td>eval()</td>
<td>Evaluates a string as PHP code</td>
</tr>
<tr>
<td>exit()</td>
<td>Prints a message and exits the current script</td>
</tr>
<tr>
<td>get_browser()</td>
<td>Returns the capabilities of the user's browser</td>
</tr>
<tr>
<td>__halt_compiler()</td>
<td>Halts the compiler execution</td>
</tr>
<tr>
<td>highlight_file()</td>
<td>Outputs a file with the PHP syntax highlighted</td>
</tr>
<tr>
<td>highlight_string()</td>
<td>Outputs a string with the PHP syntax highlighted</td>
</tr>
<tr>
<td>hrtime()</td>
<td>Returns the system's high resolution time</td>
</tr>
<tr>
<td>ignore_user_abort()</td>
<td>Sets whether a remote client can abort the running of a</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>pack()</td>
<td>Packs data into a binary string</td>
</tr>
<tr>
<td>php_strip_whitespace()</td>
<td>Returns the source code of a file with PHP comments and whitespace removed</td>
</tr>
<tr>
<td>show_source()</td>
<td>Alias of highlight_file()</td>
</tr>
<tr>
<td>sleep()</td>
<td>Delays code execution for a number of seconds</td>
</tr>
<tr>
<td>sys_getloadavg()</td>
<td>Returns the system load average</td>
</tr>
<tr>
<td>time_nanosleep()</td>
<td>Delays code execution for a number of seconds and nanoseconds</td>
</tr>
<tr>
<td>time_sleep_until()</td>
<td>Makes a script sleep until the specified time</td>
</tr>
<tr>
<td>uniqid()</td>
<td>Generates a unique ID</td>
</tr>
<tr>
<td>unpack()</td>
<td>Unpacks data from a binary string</td>
</tr>
<tr>
<td>usleep()</td>
<td>Delays code execution for a number of microseconds</td>
</tr>
</tbody>
</table>

### PHP Predefined Miscellaneous Constants

<table>
<thead>
<tr>
<th>Constant</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONNECTION_ABORTED</td>
<td>Connection is aborted by user or network error</td>
</tr>
</tbody>
</table>
**PHP MySQLi Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>affected_rows()</td>
<td>Returns the number of affected rows in the previous MySQL operation</td>
</tr>
<tr>
<td>autocommit()</td>
<td>Turns on or off auto-committing database modifications</td>
</tr>
<tr>
<td>begin_transaction()</td>
<td>Starts a transaction</td>
</tr>
<tr>
<td>change_user()</td>
<td>Changes the user of the specified database connection</td>
</tr>
<tr>
<td>character_set_name()</td>
<td>Returns the default character set for the database connection</td>
</tr>
<tr>
<td>close()</td>
<td>Closes a previously opened database connection</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>commit()</td>
<td>Commits the current transaction</td>
</tr>
<tr>
<td>connect()</td>
<td>Opens a new connection to the MySQL server</td>
</tr>
<tr>
<td>connect_errno()</td>
<td>Returns the error code from the last connection error</td>
</tr>
<tr>
<td>connect_error()</td>
<td>Returns the error description from the last connection error</td>
</tr>
<tr>
<td>data_seek()</td>
<td>Adjusts the result pointer to an arbitrary row in the result-set</td>
</tr>
<tr>
<td>debug()</td>
<td>Performs debugging operations</td>
</tr>
<tr>
<td>dump_debug_info()</td>
<td>Dumps debugging info into the log</td>
</tr>
<tr>
<td>errno()</td>
<td>Returns the last error code for the most recent function call</td>
</tr>
<tr>
<td>error()</td>
<td>Returns the last error description for the most recent function call</td>
</tr>
<tr>
<td>error_list()</td>
<td>Returns a list of errors for the most recent function call</td>
</tr>
<tr>
<td>fetch_all()</td>
<td>Fetches all result rows as an associative array, a numeric array, or both</td>
</tr>
<tr>
<td>fetch_array()</td>
<td>Fetches a result row as an associative, a numeric</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>fetch_assoc()</td>
<td>Fetches a result row as an associative array</td>
</tr>
<tr>
<td>fetch_field()</td>
<td>Returns the next field in the result-set, as an object</td>
</tr>
<tr>
<td>fetch_field_direct()</td>
<td>Returns meta-data for a single field in the result-set, as an object</td>
</tr>
<tr>
<td>fetch_fields()</td>
<td>Returns an array of objects that represent the fields in a result-set</td>
</tr>
<tr>
<td>fetch_lengths()</td>
<td>Returns the lengths of the columns of the current row in the result-set</td>
</tr>
<tr>
<td>fetch_object()</td>
<td>Returns the current row of a result-set, as an object</td>
</tr>
<tr>
<td>fetch_row()</td>
<td>Fetches one row from a result-set and returns it as an enumerated array</td>
</tr>
<tr>
<td>field_count()</td>
<td>Returns the number of columns for the most recent query</td>
</tr>
<tr>
<td>field_seek()</td>
<td>Sets the field cursor to the given field offset</td>
</tr>
<tr>
<td>get_charset()</td>
<td>Returns a character set object</td>
</tr>
<tr>
<td>get_client_info()</td>
<td>Returns the MySQL client library version</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>get_client_stats()</td>
<td>Returns statistics about client per-process</td>
</tr>
<tr>
<td>get_client_version()</td>
<td>Returns the MySQL client library version as an integer</td>
</tr>
<tr>
<td>get_connection_stats()</td>
<td>Returns statistics about the client connection</td>
</tr>
<tr>
<td>get_host_info()</td>
<td>Returns the MySQL server hostname and the connection type</td>
</tr>
<tr>
<td>get_proto_info()</td>
<td>Returns the MySQL protocol version</td>
</tr>
<tr>
<td>get_server_info()</td>
<td>Returns the MySQL server version</td>
</tr>
<tr>
<td>get_server_version()</td>
<td>Returns the MySQL server version as an integer</td>
</tr>
<tr>
<td>info()</td>
<td>Returns information about the last executed query</td>
</tr>
<tr>
<td>init()</td>
<td>Initializes MySQLi and returns a resource for use with real_connect()</td>
</tr>
<tr>
<td>insert_id()</td>
<td>Returns the auto-generated id from the last query</td>
</tr>
<tr>
<td>kill()</td>
<td>Asks the server to kill a MySQL thread</td>
</tr>
<tr>
<td>more_results()</td>
<td>Checks if there are more results from a multi query</td>
</tr>
<tr>
<td>multi_query()</td>
<td>Performs one or more queries on the database</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>next_result()</td>
<td>Prepares the next result-set from multi_query()</td>
</tr>
<tr>
<td>options()</td>
<td>Sets extra connect options and affect behavior for a connection</td>
</tr>
<tr>
<td>ping()</td>
<td>Pings a server connection, or tries to reconnect if the connection has gone down</td>
</tr>
<tr>
<td>poll()</td>
<td>Polls connections</td>
</tr>
<tr>
<td>prepare()</td>
<td>Prepares an SQL statement for execution</td>
</tr>
<tr>
<td>query()</td>
<td>Performs a query against a database</td>
</tr>
<tr>
<td>real_connect()</td>
<td>Opens a new connection to the MySQL server</td>
</tr>
<tr>
<td>real_escape_string()</td>
<td>Escapes special characters in a string for use in an SQL statement</td>
</tr>
<tr>
<td>real_query()</td>
<td>Executes a single SQL query</td>
</tr>
<tr>
<td>reap_async_query()</td>
<td>Returns result from an async SQL query</td>
</tr>
<tr>
<td>refresh()</td>
<td>Refreshes/flushes tables or caches, or resets the replication server information</td>
</tr>
<tr>
<td>rollback()</td>
<td>Rolls back the current transaction for the database</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>select_db()</code></td>
<td>Select the default database for database queries</td>
</tr>
<tr>
<td><code>set_charset()</code></td>
<td>Sets the default client character set</td>
</tr>
<tr>
<td><code>set_local_infile_default()</code></td>
<td>Unsets user defined handler for load local infile command</td>
</tr>
<tr>
<td><code>set_local_infile_handler()</code></td>
<td>Set callback function for LOAD DATA LOCAL INFILE command</td>
</tr>
<tr>
<td><code>sqlstate()</code></td>
<td>Returns the SQLSTATE error code for the error</td>
</tr>
<tr>
<td><code>ssl_set()</code></td>
<td>Used to establish secure connections using SSL</td>
</tr>
<tr>
<td><code>stat()</code></td>
<td>Returns the current system status</td>
</tr>
<tr>
<td><code>stmt_init()</code></td>
<td>Initializes a statement and returns an object for use with stmt_prepare()</td>
</tr>
<tr>
<td><code>store_result()</code></td>
<td>Transfers a result-set from the last query</td>
</tr>
<tr>
<td><code>thread_id()</code></td>
<td>Returns the thread ID for the current connection</td>
</tr>
<tr>
<td><code>thread_safe()</code></td>
<td>Returns whether the client library is compiled as thread-safe</td>
</tr>
<tr>
<td><code>use_result()</code></td>
<td>Initiates the retrieval of a result-set from the last query executed</td>
</tr>
</tbody>
</table>
### PHP Network Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>checkdnsrr()</td>
<td>Checks DNS records for type corresponding to host</td>
</tr>
<tr>
<td>closelog()</td>
<td>Closes the connection of system logger</td>
</tr>
<tr>
<td>define_syslog_variables()</td>
<td>Deprecated and removed in PHP 5.4. Initializes the variables used in syslog functions</td>
</tr>
<tr>
<td>dns_check_record()</td>
<td>Alias of checkdnsrr()</td>
</tr>
<tr>
<td>dns_get_mx()</td>
<td>Alias of getmxrr()</td>
</tr>
<tr>
<td>dns_get_record()</td>
<td>Gets the DNS resource records associated with the specified hostname</td>
</tr>
<tr>
<td>fsockopen()</td>
<td>Opens an Internet or Unix domain socket connection</td>
</tr>
<tr>
<td>gethostbyaddr()</td>
<td>Returns the domain name for a given IP address</td>
</tr>
<tr>
<td>gethostbyname()</td>
<td>Returns the IPv4 address for a given domain/host name</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>gethostbynamel()</td>
<td>Returns a list of IPv4 address for a given domain/host name</td>
</tr>
<tr>
<td>gethostname()</td>
<td>Returns the host name</td>
</tr>
<tr>
<td>getmxr()</td>
<td>Returns the MX records for the specified internet host name</td>
</tr>
<tr>
<td>getprotobynumber()</td>
<td>Returns the protocol name for a given protocol number</td>
</tr>
<tr>
<td>getprotobynumber()</td>
<td>Returns the protocol name for a given protocol number</td>
</tr>
<tr>
<td>getservbyname()</td>
<td>Returns the port number for a given Internet service and protocol</td>
</tr>
<tr>
<td>getservbyport()</td>
<td>Returns the Internet service for a given port and protocol</td>
</tr>
<tr>
<td>header_register_callback()</td>
<td>Calls a header function</td>
</tr>
<tr>
<td>header_remove()</td>
<td>Removes an HTTP header previously set with the header() function</td>
</tr>
<tr>
<td>header()</td>
<td>Sends a raw HTTP header to a client</td>
</tr>
<tr>
<td>headers_list()</td>
<td>Returns a list of response headers to be sent to the browser</td>
</tr>
<tr>
<td>headers_sent()</td>
<td>Checks if/where headers have been sent</td>
</tr>
<tr>
<td>Function Name</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------------</td>
</tr>
<tr>
<td>http_response_code()</td>
<td>Sets or returns the HTTP response status code</td>
</tr>
<tr>
<td>inet_ntop()</td>
<td>Converts a 32bit IPv4 or 128bit IPv6 address into a readable format</td>
</tr>
<tr>
<td>inet_pton()</td>
<td>Converts a readable IP address into a packed 32bit IPv4 or 128bit IPv6 format</td>
</tr>
<tr>
<td>ip2long()</td>
<td>Converts an IPv4 address into a long integer</td>
</tr>
<tr>
<td>long2ip()</td>
<td>Converts a long integer address into a string in IPv4 format</td>
</tr>
<tr>
<td>openlog()</td>
<td>Opens the connection of system logger</td>
</tr>
<tr>
<td>pfsockopen()</td>
<td>Opens a persistent Internet or Unix domain socket connection</td>
</tr>
<tr>
<td>setcookie()</td>
<td>Defines a cookie to be sent along with the rest of the HTTP headers</td>
</tr>
<tr>
<td>setrawcookie()</td>
<td>Defines a cookie (without URL encoding) to be sent along with the rest of the HTTP headers</td>
</tr>
<tr>
<td>socket_get_status()</td>
<td>Alias of stream_get_meta_data()</td>
</tr>
<tr>
<td>socket_set_blocking()</td>
<td>Alias of stream_set_blocking()</td>
</tr>
<tr>
<td>socket_set_timeout()</td>
<td>Alias of stream_set_timeout()</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>__construct()</td>
<td>Creates a new SimpleXMLElement object</td>
</tr>
<tr>
<td>__toString()</td>
<td>Returns the string content of an element</td>
</tr>
<tr>
<td>addAttribute()</td>
<td>Appends an attribute to the SimpleXML element</td>
</tr>
<tr>
<td>addChild()</td>
<td>Appends a child element the SimpleXML element</td>
</tr>
<tr>
<td>asXML()</td>
<td>Returns a well-formed XML string (XML version 1.0) from a SimpleXML object</td>
</tr>
<tr>
<td>attributes()</td>
<td>Returns the attributes/values of an element</td>
</tr>
<tr>
<td>children()</td>
<td>Returns the children of a specified node</td>
</tr>
<tr>
<td>count()</td>
<td>Counts the children of a specified node</td>
</tr>
<tr>
<td>getDocNamespaces()</td>
<td>Returns the namespaces declared in document</td>
</tr>
<tr>
<td>getName()</td>
<td>Returns the name of an element</td>
</tr>
</tbody>
</table>

syslog() Generates a system log message

**PHP SimpleXML Functions**
getNamespaces() | Returns the namespaces used in document

registerXPathNamespace() | Creates a namespace context for the next XPath query

saveXML() | Alias of asXML()

simplexml_import_dom() | Returns a SimpleXMLElement object from a DOM node

simplexml_load_file() | Converts an XML document to an object

simplexml_load_string() | Converts an XML string to an object

xpath() | Runs an XPath query on XML data

**PHP SimpleXML Iteration Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>current()</td>
<td>Returns the current element</td>
</tr>
<tr>
<td>getChildren()</td>
<td>Returns the child elements of the current element</td>
</tr>
<tr>
<td>hasChildren()</td>
<td>Checks whether the current element has children</td>
</tr>
<tr>
<td>key()</td>
<td>Returns the XML tag name of the current element</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>next()</td>
<td>Moves to the next element</td>
</tr>
<tr>
<td>rewind()</td>
<td>Rewinds to the first element</td>
</tr>
<tr>
<td>valid()</td>
<td>Checks whether the current element is valid</td>
</tr>
</tbody>
</table>

**PHP Stream Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>set_socket_blocking()</td>
<td>Deprecated in PHP 5.4, and removed in PHP 7.0. Alias of stream_set_blocking()</td>
</tr>
<tr>
<td>stream_bucket_prepend()</td>
<td></td>
</tr>
<tr>
<td>stream_context_create()</td>
<td></td>
</tr>
<tr>
<td>stream_context_get_default()</td>
<td></td>
</tr>
<tr>
<td>stream_context_get_options()</td>
<td></td>
</tr>
<tr>
<td>stream_context_get_params()</td>
<td></td>
</tr>
<tr>
<td>stream_context_set_default()</td>
<td></td>
</tr>
<tr>
<td>stream_context_set_options()</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>stream_context_set_params()</td>
<td></td>
</tr>
<tr>
<td>stream_copy_to_stream()</td>
<td>Copies data from one stream to another</td>
</tr>
<tr>
<td>stream_filter_append()</td>
<td>Appends a filter to a stream</td>
</tr>
<tr>
<td>stream_filter_prepend()</td>
<td></td>
</tr>
<tr>
<td>stream_filter_register()</td>
<td></td>
</tr>
<tr>
<td>stream_filter_remove()</td>
<td></td>
</tr>
<tr>
<td>stream_get_contents()</td>
<td></td>
</tr>
<tr>
<td>stream_get_filters()</td>
<td></td>
</tr>
<tr>
<td>stream_get_line()</td>
<td></td>
</tr>
<tr>
<td>stream_get_meta_data()</td>
<td></td>
</tr>
<tr>
<td>stream_get_transports()</td>
<td></td>
</tr>
<tr>
<td>stream_get_wrappers()</td>
<td></td>
</tr>
<tr>
<td>stream_is_local()</td>
<td></td>
</tr>
<tr>
<td>stream_isatty()</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------</td>
<td>------------------------------------------------</td>
</tr>
<tr>
<td>stream_notification_callback()</td>
<td></td>
</tr>
<tr>
<td>stream_register_wrapper()</td>
<td>Alias of stream_wrapper_register()</td>
</tr>
<tr>
<td>stream_resolve_include_path()</td>
<td></td>
</tr>
<tr>
<td>stream_select()</td>
<td></td>
</tr>
<tr>
<td>stream_set_blocking()</td>
<td></td>
</tr>
<tr>
<td>stream_set_chunk_size()</td>
<td></td>
</tr>
<tr>
<td>stream_set_read_buffer()</td>
<td></td>
</tr>
<tr>
<td>stream_set_timeout()</td>
<td></td>
</tr>
<tr>
<td>stream_set_write_buffer()</td>
<td></td>
</tr>
<tr>
<td>stream_socket_accept()</td>
<td></td>
</tr>
<tr>
<td>stream_socket_client()</td>
<td></td>
</tr>
<tr>
<td>stream_socket_enable_crypto()</td>
<td></td>
</tr>
<tr>
<td>stream_socket_get_name()</td>
<td></td>
</tr>
<tr>
<td>stream_socket_pair()</td>
<td></td>
</tr>
</tbody>
</table>
### Stream Functions

<table>
<thead>
<tr>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>stream_socket_recvfrom()</code></td>
</tr>
<tr>
<td><code>stream_socket_sendto()</code></td>
</tr>
<tr>
<td><code>stream_socket_server()</code></td>
</tr>
<tr>
<td><code>stream_socket_shutdown()</code></td>
</tr>
<tr>
<td><code>stream_supports_lock()</code></td>
</tr>
<tr>
<td><code>stream_wrapper_register()</code></td>
</tr>
<tr>
<td><code>stream_wrapper_restore()</code></td>
</tr>
<tr>
<td><code>stream_wrapper_unregister()</code></td>
</tr>
</tbody>
</table>

### PHP String Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>addcslashes()</code></td>
<td>Returns a string with backslashes in front of the specified characters</td>
</tr>
<tr>
<td><code>addslashes()</code></td>
<td>Returns a string with backslashes in front of predefined characters</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>bin2hex()</code></td>
<td>Converts a string of ASCII characters to hexadecimal values</td>
</tr>
<tr>
<td><code>chop()</code></td>
<td>Removes whitespace or other characters from the right end of a string</td>
</tr>
<tr>
<td><code>chr()</code></td>
<td>Returns a character from a specified ASCII value</td>
</tr>
<tr>
<td><code>chunk_split()</code></td>
<td>Splits a string into a series of smaller parts</td>
</tr>
<tr>
<td><code>convert_cyr_string()</code></td>
<td>Converts a string from one Cyrillic character-set to another</td>
</tr>
<tr>
<td><code>convert_uudecode()</code></td>
<td>Decodes a uuencoded string</td>
</tr>
<tr>
<td><code>convert_uuencode()</code></td>
<td>Encodes a string using the uuencode algorithm</td>
</tr>
<tr>
<td><code>count_chars()</code></td>
<td>Returns information about characters used in a string</td>
</tr>
<tr>
<td><code>crc32()</code></td>
<td>Calculates a 32-bit CRC for a string</td>
</tr>
<tr>
<td><code>crypt()</code></td>
<td>One-way string hashing</td>
</tr>
<tr>
<td><code>echo()</code></td>
<td>Outputs one or more strings</td>
</tr>
<tr>
<td><code>explode()</code></td>
<td>Breaks a string into an array</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>fprintf()</code></td>
<td>Writes a formatted string to a specified output stream</td>
</tr>
<tr>
<td><code>get_html_translation_table()</code></td>
<td>Returns the translation table used by <code>htmlspecialchars()</code> and <code>htmlentities()</code></td>
</tr>
<tr>
<td><code>hebrew()</code></td>
<td>Converts Hebrew text to visual text</td>
</tr>
<tr>
<td><code>hebrevc()</code></td>
<td>Converts Hebrew text to visual text and new lines (<code>\n</code>) into <code>&lt;br&gt;</code></td>
</tr>
<tr>
<td><code>hex2bin()</code></td>
<td>Converts a string of hexadecimal values to ASCII characters</td>
</tr>
<tr>
<td><code>html_entity_decode()</code></td>
<td>Converts HTML entities to characters</td>
</tr>
<tr>
<td><code>htmlentities()</code></td>
<td>Converts characters to HTML entities</td>
</tr>
<tr>
<td><code>htmlspecialchars_decode()</code></td>
<td>Converts some predefined HTML entities to characters</td>
</tr>
<tr>
<td><code>htmlspecialchars()</code></td>
<td>Converts some predefined characters to HTML entities</td>
</tr>
<tr>
<td><code>implode()</code></td>
<td>Returns a string from the elements of an array</td>
</tr>
<tr>
<td><code>join()</code></td>
<td>Alias of <code>implode()</code></td>
</tr>
<tr>
<td><code>lcfirst()</code></td>
<td>Converts the first character of a string to...</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>lowercase()</td>
<td>Returns the Levenshtein distance between two strings</td>
</tr>
<tr>
<td>localeconv()</td>
<td>Returns locale numeric and monetary formatting information</td>
</tr>
<tr>
<td>ltrim()</td>
<td>Removes whitespace or other characters from the left side of a string</td>
</tr>
<tr>
<td>md5()</td>
<td>Calculates the MD5 hash of a string</td>
</tr>
<tr>
<td>md5_file()</td>
<td>Calculates the MD5 hash of a file</td>
</tr>
<tr>
<td>metaphone()</td>
<td>Calculates the metaphone key of a string</td>
</tr>
<tr>
<td>money_format()</td>
<td>Returns a string formatted as a currency string</td>
</tr>
<tr>
<td>nl_langinfo()</td>
<td>Returns specific local information</td>
</tr>
<tr>
<td>nl2br()</td>
<td>Inserts HTML line breaks in front of each newline in a string</td>
</tr>
<tr>
<td>number_format()</td>
<td>Formats a number with grouped thousands</td>
</tr>
<tr>
<td>ord()</td>
<td>Returns the ASCII value of the first character of a string</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>------------------------------------------------------------------</td>
</tr>
<tr>
<td>parse_str()</td>
<td>Parses a query string into variables</td>
</tr>
<tr>
<td>print()</td>
<td>Outputs one or more strings</td>
</tr>
<tr>
<td>printf()</td>
<td>Outputs a formatted string</td>
</tr>
<tr>
<td>quoted_printable_decode()</td>
<td>Converts a quoted-printable string to an 8-bit string</td>
</tr>
<tr>
<td>quoted_printable_encode()</td>
<td>Converts an 8-bit string to a quoted printable string</td>
</tr>
<tr>
<td>quotemeta()</td>
<td>Quotes meta characters</td>
</tr>
<tr>
<td>rtrim()</td>
<td>Removes whitespace or other characters from the right side of a string</td>
</tr>
<tr>
<td>setlocale()</td>
<td>Sets locale information</td>
</tr>
<tr>
<td>sha1()</td>
<td>Calculates the SHA-1 hash of a string</td>
</tr>
<tr>
<td>sha1_file()</td>
<td>Calculates the SHA-1 hash of a file</td>
</tr>
<tr>
<td>similar_text()</td>
<td>Calculates the similarity between two strings</td>
</tr>
<tr>
<td>soundex()</td>
<td>Calculates the soundex key of a string</td>
</tr>
<tr>
<td>sprintf()</td>
<td>Writes a formatted string to a variable</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>sscanf()</td>
<td>Parses input from a string according to a format</td>
</tr>
<tr>
<td>str_getcsv()</td>
<td>Parses a CSV string into an array</td>
</tr>
<tr>
<td>str_ireplace()</td>
<td>Replaces some characters in a string (case-insensitive)</td>
</tr>
<tr>
<td>str_pad()</td>
<td>Pads a string to a new length</td>
</tr>
<tr>
<td>str_repeat()</td>
<td>Repeats a string a specified number of times</td>
</tr>
<tr>
<td>str_replace()</td>
<td>Replaces some characters in a string (case-sensitive)</td>
</tr>
<tr>
<td>str_rot13()</td>
<td>Performs the ROT13 encoding on a string</td>
</tr>
<tr>
<td>str_shuffle()</td>
<td>Randomly shuffles all characters in a string</td>
</tr>
<tr>
<td>str_split()</td>
<td>Splits a string into an array</td>
</tr>
<tr>
<td>str_word_count()</td>
<td>Count the number of words in a string</td>
</tr>
<tr>
<td>strcasecmp()</td>
<td>Compares two strings (case-insensitive)</td>
</tr>
<tr>
<td>strchr()</td>
<td>Finds the first occurrence of a string inside another string (alias of strstr())</td>
</tr>
<tr>
<td>strcmp()</td>
<td>Compares two strings (case-sensitive)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td><code>strcoll()</code></td>
<td>Compares two strings (locale based string comparison)</td>
</tr>
<tr>
<td><code>strcspn()</code></td>
<td>Returns the number of characters found in a string before any part of some specified characters are found</td>
</tr>
<tr>
<td><code>strip_tags()</code></td>
<td>Strips HTML and PHP tags from a string</td>
</tr>
<tr>
<td><code>stripcslashes()</code></td>
<td>Unquotes a string quoted with addcslashes()</td>
</tr>
<tr>
<td><code>stripslashes()</code></td>
<td>Unquotes a string quoted with addslashes()</td>
</tr>
<tr>
<td><code>stripos()</code></td>
<td>Returns the position of the first occurrence of a string inside another string (case-insensitive)</td>
</tr>
<tr>
<td><code>stristr()</code></td>
<td>Finds the first occurrence of a string inside another string (case-insensitive)</td>
</tr>
<tr>
<td><code>strlen()</code></td>
<td>Returns the length of a string</td>
</tr>
<tr>
<td><code>strnatcasecmp()</code></td>
<td>Compares two strings using a &quot;natural order&quot; algorithm (case-insensitive)</td>
</tr>
<tr>
<td><code>strnatcmp()</code></td>
<td>Compares two strings using a &quot;natural order&quot; algorithm (case-sensitive)</td>
</tr>
<tr>
<td><code>strncasecmp()</code></td>
<td>String comparison of the first n characters (case-insensitive)</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>strncmp()</td>
<td>String comparison of the first <code>n</code> characters (case-sensitive)</td>
</tr>
<tr>
<td>strpbrk()</td>
<td>Searches a string for any of a set of characters</td>
</tr>
<tr>
<td>strpos()</td>
<td>Returns the position of the first occurrence of a string inside another string (case-sensitive)</td>
</tr>
<tr>
<td>strrchr()</td>
<td>Finds the last occurrence of a string inside another string</td>
</tr>
<tr>
<td>strrev()</td>
<td>Reverses a string</td>
</tr>
<tr>
<td>stripos()</td>
<td>Finds the position of the last occurrence of a string inside another string (case-insensitive)</td>
</tr>
<tr>
<td>strrpos()</td>
<td>Finds the position of the last occurrence of a string inside another string (case-sensitive)</td>
</tr>
<tr>
<td>strspn()</td>
<td>Returns the number of characters found in a string that contains only characters from a specified charlist</td>
</tr>
<tr>
<td>strstr()</td>
<td>Finds the first occurrence of a string inside another string (case-sensitive)</td>
</tr>
<tr>
<td>strtok()</td>
<td>Splits a string into smaller strings</td>
</tr>
<tr>
<td>strtolower()</td>
<td>Converts a string to lowercase letters</td>
</tr>
<tr>
<td>Scripting language</td>
<td>Programming language</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>Platform-specific</td>
<td>Platform-agnostic (cross-platform)</td>
</tr>
<tr>
<td>Interpreted</td>
<td>Compiled</td>
</tr>
<tr>
<td>Faster at runtime</td>
<td>Slower at runtime</td>
</tr>
<tr>
<td>More code-intensive</td>
<td>Less code-intensive</td>
</tr>
<tr>
<td>Creates standalone applications</td>
<td>Creates applications as part of a stack</td>
</tr>
</tbody>
</table>

A step-by-step procedure to get expected output from the given input

Priori Analysis | Posterior Analysis
--- | ---
checking the algorithm before its implementation | checking the algorithm after its implementation

Design → Analyze → Implement → Experiment → Algorithms → Design
Characteristics of an Algorithm

- Well-defined input
- Desired output
- Finiteness (not end up in an infinite loops)
- Effectiveness (executed in finite time)
- Definiteness (precisely defined)

Algorithm + Data structure = Program

The way in which various program data elements are organized and stored into the memory so that the data can be used efficiently.

Algorithms are not arbiters of objective truth and fairness simply because they're math.

― Zoe Quinn

Problem

Algorithm

Program code

Program

Input

Computer system

Output
Performance analysis of an algorithm depends on 2 factors:

- **Space Complexity**: The amount of memory space required by an algorithm to complete its task
- **Time Complexity**: The amount of time required by an algorithm complete its task

The number of operations an algorithm performs to complete its task (considering that each operation takes the same amount of time)

<table>
<thead>
<tr>
<th>Linear Data Structure</th>
<th>Non-linear Data Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data items are arranged in sequential order (one after the other)</td>
<td>Data items are arranged in non-sequential order (hierarchical manner)</td>
</tr>
<tr>
<td>Memory is not utilized in an efficient way</td>
<td>Memory is utilized in an efficient way</td>
</tr>
</tbody>
</table>

Time complexity increase with the data | Time complexity remains the same

The algorithm to find the largest number among 3 numbers:

1. **Step 1**: Start
2. **Step 2**: Declare variables a, b and c.
3. **Step 3**: Read variables a, b and c.
4. **Step 4**: If a > b
   
   If a > c
Display a is the largest number.

Else

Display c is the largest number.

Else

If b > c

Display b is the largest number.

Else

Display c is the greatest number.

Step 5: Stop

The time taken by the computer to run code = number of instructions × time to execute each instruction

**Factors that affect run time of an algorithm:**

- The hardware platform used
- Representation of abstract data types
- Efficiency of compiler
- Implementer programming skill
- Complexity of underlying algorithm
- Size of the input

**Worst Case Complexity:** The maximum time taken by an algorithm to complete its task.

**Best Case Complexity:** The minimum time taken by an algorithm to complete its task.

**Average Case Complexity:** The average time taken by an algorithm to complete its task.

**Run time**

- Polynomial time
- Superpolynomial time
**Polynomial time** → run time that does not increase faster than \(n^k\), which includes:

- constant time \((n^0)\)
- logarithmic time \((\log n)\)
- linear time \((n^1)\)
- quadratic time \((n^2)\) and other higher degree polynomials (like \(n^3\))

**Superpolynomial time** → run time that does increase faster than \(n^k\), which includes:

- exponential time \((2^n)\)
- factorial time \((n!)\) and anything else faster.

An algorithm is said to take **constant time** \((n^0)\), if

- The run time of an algorithm → **constant** (doesn't increase) − no matter how large the input size increases

An algorithm is said to take **logarithmic time** \((\log n)\), if

- The run time of an algorithm increases in direct proportion to the logarithm of the input size

An algorithm is said to take **linear time** \((n^1)\), if

- The run time of an algorithm increases in direct proportion to the input size

Whenever input size doubles, the running time increases twofold

An algorithm is said to take **quadratic time** \((n^2)\), if

- The run time of an algorithm increases in direct proportion to the input size squared

Whenever input size doubles, the running time increases fourfold
An algorithm is said to take **cubic time** \( (n^3) \), if

- The run time of an algorithm increases in direct proportion to the cube of the input size

Whenever input size doubles, the running time increases eightfold

An algorithm is said to take **factorial time** \( (!n) \), if

- The run time of an algorithm increases in direct proportion to the factorial of the input size

Whenever input size increases by 1, the running time increases by a factor of input size

**The better the time complexity of an algorithm is, the faster the algorithm will complete its task**

An algorithm is said to take **linearithmic time** \( (n \log n) \), if

- The run time of an algorithm increases in direct proportion to the input size times the logarithm of the input size

The total amount of memory space used by the algorithm to execute and produce the result

**Space Complexity** = Auxiliary space + Memory space used by input values

The extra memory space used by the algorithm during its execution
For any program, **memory space** is required for the following purposes:

- To store compiled version of instructions (**Instruction Space**)
- To store information of partially executed functions at the time of function call (**Environmental Stack**)
- To store all the variables and constants (**Data Space**)

<table>
<thead>
<tr>
<th>Constant space complexity</th>
<th>takes the same amount of memory space regardless of the input size (n)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logarithmic space complexity</td>
<td>takes memory space proportional to log n</td>
</tr>
<tr>
<td>Linear space complexity</td>
<td>takes memory space directly proportional to n</td>
</tr>
<tr>
<td>Linearithmic space complexity</td>
<td>takes memory space directly proportional to n log n</td>
</tr>
<tr>
<td>Quadratic space complexity</td>
<td>takes memory space directly proportional to n²</td>
</tr>
<tr>
<td>Cubic space complexity</td>
<td>takes memory space directly proportional to n³</td>
</tr>
<tr>
<td>Exponential space complexity</td>
<td>takes memory space directly proportional to 2^n</td>
</tr>
<tr>
<td>Factorial space complexity</td>
<td>takes memory space directly proportional to !n</td>
</tr>
</tbody>
</table>

```c
#include<stdio.h>
int main()
{
    int x = 4, y = 6, z;
    z = x + y;
    printf("%d", z);
    return 0;
}
```

- In the above program, 3 integer variables are used, hence they will take up 4 bytes each, so the total space occupied by the above-given program is 4 × 3 = 12 bytes.
In this program, we have three integer variables. Therefore, this program always takes 12 bytes of memory space to complete its execution. And because this memory space requirement is fixed for the above program, hence space complexity is said to be **constant space complexity** or O(1) space complexity.

```java
public int sumArray(int[] array) {
    int size = 0;
    int sum = 0;

    for (int iterator = 0; iterator < size; iterator++) {
        sum += array[iterator];
    }

    return sum;
}
```

In the above program:

- **array** – the function’s only argument – the space taken by the array is equal to 4n bytes (where n is the length of the array)
- **size** – a 4-byte integer
- **sum** – a 4-byte integer
- **iterator** – a 4-byte integer

The total memory space needed for this program to execute is $4n + 4 + 4 + 4 = 4n + 12$ bytes – which will increasing linearly with the increase in the input value n, hence it is called as **linear space complexity** or O(n) space complexity.
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>strtoupper()</td>
<td>Converts a string to uppercase letters</td>
</tr>
<tr>
<td>strtr()</td>
<td>Translates certain characters in a string</td>
</tr>
<tr>
<td>substr()</td>
<td>Returns a part of a string</td>
</tr>
<tr>
<td>substr_compare()</td>
<td>Compares two strings from a specified start position (binary safe and optionally case-sensitive)</td>
</tr>
<tr>
<td>substr_count()</td>
<td>Counts the number of times a substring occurs in a string</td>
</tr>
<tr>
<td>substr_replace()</td>
<td>Replaces a part of a string with another string</td>
</tr>
<tr>
<td>trim()</td>
<td>Removes whitespace or other characters from both sides of a string</td>
</tr>
<tr>
<td>ucfirst()</td>
<td>Converts the first character of a string to uppercase</td>
</tr>
<tr>
<td>ucwords()</td>
<td>Converts the first character of each word in a string to uppercase</td>
</tr>
<tr>
<td>vfprintf()</td>
<td>Writes a formatted string to a specified output stream</td>
</tr>
<tr>
<td>vprintf()</td>
<td>Outputs a formatted string</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>vsprintf()</td>
<td>Writes a formatted string to a variable</td>
</tr>
<tr>
<td>wordwrap()</td>
<td>Wraps a string to a given number of characters</td>
</tr>
</tbody>
</table>

**PHP XML Parser Functions**

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>utf8_decode()</td>
<td>Decodes an UTF-8 string to ISO-8859-1</td>
</tr>
<tr>
<td>utf8_encode()</td>
<td>Encodes an ISO-8859-1 string to UTF-8</td>
</tr>
<tr>
<td>xml_error_string()</td>
<td>Returns an error string from the XML parser</td>
</tr>
<tr>
<td>xml_get_current_byte_index()</td>
<td>Returns the current byte index from the XML parser</td>
</tr>
<tr>
<td>xml_get_current_column_number()</td>
<td>Returns the current column number from the XML parser</td>
</tr>
<tr>
<td>xml_get_current_line_number()</td>
<td>Returns the current line number from the XML parser</td>
</tr>
<tr>
<td>xml_get_error_code()</td>
<td>Returns an error code from the XML parser</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>xml_parse()</code></td>
<td>Parses an XML document</td>
</tr>
<tr>
<td><code>xml_parse_into_struct()</code></td>
<td>Parses XML data into an array</td>
</tr>
<tr>
<td><code>xml_parser_create_ns()</code></td>
<td>Creates an XML parser with namespace support</td>
</tr>
<tr>
<td><code>xml_parser_create()</code></td>
<td>Creates an XML parser</td>
</tr>
<tr>
<td><code>xml_parser_free()</code></td>
<td>Frees an XML parser</td>
</tr>
<tr>
<td><code>xml_parser_get_option()</code></td>
<td>Returns options from an XML parser</td>
</tr>
<tr>
<td><code>xml_parser_set_option()</code></td>
<td>Sets options in an XML parser</td>
</tr>
<tr>
<td><code>xml_set_character_data_handler()</code></td>
<td>Sets up the character data handler for the XML parser</td>
</tr>
<tr>
<td><code>xml_set_default_handler()</code></td>
<td>Sets up the default data handler for the XML parser</td>
</tr>
<tr>
<td><code>xml_set_element_handler()</code></td>
<td>Sets up start and end element handlers for the XML parser</td>
</tr>
<tr>
<td><code>xml_set_end_namespace_decl_handler()</code></td>
<td>Sets up the end namespace declaration handler</td>
</tr>
<tr>
<td><code>xml_set_external_entity_ref_handler()</code></td>
<td>Sets up the external entity reference handler for the XML parser</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td><code>xml_set_notation_decl_handler()</code></td>
<td>Sets up notation declaration handler for the XML parser</td>
</tr>
<tr>
<td><code>xml_set_object()</code></td>
<td>Allows to use XML parser within an object</td>
</tr>
<tr>
<td><code>xml_set_processing_instruction_handler()</code></td>
<td>Sets up processing instruction handler</td>
</tr>
<tr>
<td><code>xml_set_start_namespace_decl_handler()</code></td>
<td>Sets up the start namespace declaration handler</td>
</tr>
<tr>
<td><code>xml_set_unparsed_entity_decl_handler()</code></td>
<td>Sets handler function for unparsed entity declarations</td>
</tr>
</tbody>
</table>

**PHP XML Parser Constants**

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>XML_ERROR_NONE</code></td>
<td>(integer)</td>
</tr>
<tr>
<td><code>XML_ERROR_NO_MEMORY</code></td>
<td>(integer)</td>
</tr>
<tr>
<td><code>XML_ERROR_SYNTAX</code></td>
<td>(integer)</td>
</tr>
<tr>
<td><code>XML_ERROR_NO_ELEMENTS</code></td>
<td>(integer)</td>
</tr>
<tr>
<td><code>XML_ERROR_INVALID_TOKEN</code></td>
<td>(integer)</td>
</tr>
<tr>
<td>Error Code</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>XML_ERROR_UNCLOSED_TOKEN</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_PARTIAL_CHAR</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_TAG_MISMATCH</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_DUPLICATE_ATTRIBUTE</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_JUNK_AFTER_DOC_ELEMENT</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_PARAM_ENTITY_REF</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_UNDEFINED_ENTITY</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_RECURSIVE_ENTITY_REF</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_ASYNC_ENTITY</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_BAD_CHAR_REF</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_BINARY_ENTITY_REF</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_ATTRIBUTE_EXTERNAL_ENTITY_REF</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_MISPLACED_XML_PI</td>
<td>(integer)</td>
</tr>
<tr>
<td>XML_ERROR_UNKNOWN_ENCODING</td>
<td>(integer)</td>
</tr>
</tbody>
</table>
XML_ERROR_INCORRECT_ENCODING (integer)

XML_ERROR_UNCLOSED_CDATA_SECTION (integer)

XML_ERROR_EXTERNAL_ENTITY_HANDLING (integer)

XML_OPTION_CASE_FOLDING (integer)

XML_OPTION_TARGET_ENCODING (integer)

XML_OPTION_SKIP_TAGSTART (integer)

XML_OPTION_SKIP_WHITE (integer)

XML_SAX_IMPL (string)

### PHP Zip Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>zip_close()</td>
<td>Closes a ZIP file archive</td>
</tr>
<tr>
<td>zip_entry_close()</td>
<td>Closes a ZIP directory entry</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td><code>zip_entry_compressedsize()</code></td>
<td>Returns the compressed file size of a ZIP directory entry</td>
</tr>
<tr>
<td><code>zip_entry_compressionmethod()</code></td>
<td>Returns the compression method of a ZIP directory entry</td>
</tr>
<tr>
<td><code>zip_entry_filesize()</code></td>
<td>Returns the actual file size of a ZIP directory entry</td>
</tr>
<tr>
<td><code>zip_entry_name()</code></td>
<td>Returns the name of a ZIP directory entry</td>
</tr>
<tr>
<td><code>zip_entry_open()</code></td>
<td>Opens a directory entry in a ZIP file for reading</td>
</tr>
<tr>
<td><code>zip_entry_read()</code></td>
<td>Reads from an open directory entry in the ZIP file</td>
</tr>
<tr>
<td><code>zip_open()</code></td>
<td>Opens a ZIP file archive</td>
</tr>
<tr>
<td><code>zip_read()</code></td>
<td>Reads the next file in a open ZIP file archive</td>
</tr>
</tbody>
</table>

**Java Reference**

**Java Keywords**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
</table>

641
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abstract</td>
<td>A non-access modifier. Used for classes and methods: An abstract class cannot be used to create objects (to access it, it must be inherited from another class). An abstract method can only be used in an abstract class, and it does not have a body. The body is provided by the subclass (inherited from)</td>
</tr>
<tr>
<td>assert</td>
<td>For debugging</td>
</tr>
<tr>
<td>boolean</td>
<td>A data type that can only store true and false values</td>
</tr>
<tr>
<td>break</td>
<td>Breaks out of a loop or a switch block</td>
</tr>
<tr>
<td>byte</td>
<td>A data type that can store whole numbers from -128 and 127</td>
</tr>
<tr>
<td>case</td>
<td>Marks a block of code in switch statements</td>
</tr>
<tr>
<td>catch</td>
<td>Catches exceptions generated by try statements</td>
</tr>
<tr>
<td>char</td>
<td>A data type that is used to store a single character</td>
</tr>
<tr>
<td>class</td>
<td>Defines a class</td>
</tr>
<tr>
<td>continue</td>
<td>Continues to the next iteration of a loop</td>
</tr>
<tr>
<td>const</td>
<td>Defines a constant. Not in use – use final instead</td>
</tr>
<tr>
<td>default</td>
<td>Specifies the default block of code in a switch statement</td>
</tr>
<tr>
<td>Word</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
</tr>
<tr>
<td>do</td>
<td>Used together with while to create a do-while loop</td>
</tr>
<tr>
<td>double</td>
<td>A data type that can store whole numbers from 1.7e-308 to 1.7e+308</td>
</tr>
<tr>
<td>else</td>
<td>Used in conditional statements</td>
</tr>
<tr>
<td>enum</td>
<td>Declares an enumerated (unchangeable) type</td>
</tr>
<tr>
<td>exports</td>
<td>Exports a package with a module. New in Java 9</td>
</tr>
<tr>
<td>extends</td>
<td>Extends a class (indicates that a class is inherited from another class)</td>
</tr>
<tr>
<td>final</td>
<td>A non-access modifier used for classes, attributes and methods, which makes them non-changeable (impossible to inherit or override)</td>
</tr>
<tr>
<td>finally</td>
<td>Used with exceptions, a block of code that will be executed no matter if there is an exception or not</td>
</tr>
<tr>
<td>float</td>
<td>A data type that can store whole numbers from 3.4e-038 to 3.4e+038</td>
</tr>
<tr>
<td>for</td>
<td>Create a for loop</td>
</tr>
<tr>
<td>goto</td>
<td>Not in use, and has no function</td>
</tr>
<tr>
<td>if</td>
<td>Makes a conditional statement</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>implements</td>
<td>Implements an interface</td>
</tr>
<tr>
<td>import</td>
<td>Used to import a package, class or interface</td>
</tr>
<tr>
<td>instanceof</td>
<td>Checks whether an object is an instance of a specific class or an interface</td>
</tr>
<tr>
<td>int</td>
<td>A data type that can store whole numbers from -2147483648 to 2147483647</td>
</tr>
<tr>
<td>interface</td>
<td>Used to declare a special type of class that only contains abstract methods</td>
</tr>
<tr>
<td>long</td>
<td>A data type that can store whole numbers from -9223372036854775808 to 9223372036854775808</td>
</tr>
<tr>
<td>module</td>
<td>Declares a module. New in Java 9</td>
</tr>
<tr>
<td>native</td>
<td>Specifies that a method is not implemented in the same Java source file (but in another language)</td>
</tr>
<tr>
<td>new</td>
<td>Creates new objects</td>
</tr>
<tr>
<td>package</td>
<td>Declares a package</td>
</tr>
<tr>
<td>private</td>
<td>An access modifier used for attributes, methods and constructors, making them only accessible within the declared class</td>
</tr>
<tr>
<td><strong>protected</strong></td>
<td>An access modifier used for attributes, methods and constructors, making them accessible in the same package and subclasses</td>
</tr>
<tr>
<td><strong>public</strong></td>
<td>An access modifier used for classes, attributes, methods and constructors, making them accessible by any other class</td>
</tr>
<tr>
<td><strong>requires</strong></td>
<td>Specifies required libraries inside a module. New in Java 9</td>
</tr>
<tr>
<td><strong>return</strong></td>
<td>Finished the execution of a method, and can be used to return a value from a method</td>
</tr>
<tr>
<td><strong>short</strong></td>
<td>A data type that can store whole numbers from -32768 to 32767</td>
</tr>
<tr>
<td><strong>static</strong></td>
<td>A non-access modifier used for methods and attributes. Static methods/attributes can be accessed without creating an object of a class</td>
</tr>
<tr>
<td><strong>strictfp</strong></td>
<td>Restrict the precision and rounding of floating point calculations</td>
</tr>
<tr>
<td><strong>super</strong></td>
<td>Refers to superclass (parent) objects</td>
</tr>
<tr>
<td><strong>switch</strong></td>
<td>Selects one of many code blocks to be executed</td>
</tr>
<tr>
<td><strong>synchronized</strong></td>
<td>A non-access modifier, which specifies that methods can only be accessed by one thread at a time</td>
</tr>
<tr>
<td><strong>this</strong></td>
<td>Refers to the current object in a method or constructor</td>
</tr>
<tr>
<td></td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>throw</td>
<td>Creates a custom error</td>
</tr>
<tr>
<td>throws</td>
<td>Indicates what exceptions may be thrown by a method</td>
</tr>
<tr>
<td>transient</td>
<td>A non-accesss modifier, which specifies that an attribute is not part of an object's persistent state</td>
</tr>
<tr>
<td>try</td>
<td>Creates a try...catch statement</td>
</tr>
<tr>
<td>var</td>
<td>Declares a variable. New in Java 10</td>
</tr>
<tr>
<td>void</td>
<td>Specifies that a method should not have a return value</td>
</tr>
<tr>
<td>volatile</td>
<td>Indicates that an attribute is not cached thread-locally, and is always read from the &quot;main memory&quot;</td>
</tr>
<tr>
<td>while</td>
<td>Creates a while loop</td>
</tr>
</tbody>
</table>

**Java Math Methods**

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Return Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>abs(x)</td>
<td>Returns the absolute value of x</td>
<td>double</td>
</tr>
<tr>
<td>acos(x)</td>
<td>Returns the arccosine of x, in radians</td>
<td>double</td>
</tr>
<tr>
<td>asin(x)</td>
<td>Returns the arcsine of x, in radians</td>
<td>double</td>
</tr>
<tr>
<td>atan(x)</td>
<td>Returns the arctangent of x as a numeric value between -PI/2 and PI/2</td>
<td>double</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
<td>Return Type</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>atan2(y, x)</td>
<td>Returns the angle theta from the conversion of rectangular coordinates (x, y) to polar coordinates (r, theta).</td>
<td>double</td>
</tr>
<tr>
<td>cbrt(x)</td>
<td>Returns the cube root of x</td>
<td>double</td>
</tr>
<tr>
<td>ceil(x)</td>
<td>Returns the value of x rounded up to its nearest integer</td>
<td>double</td>
</tr>
<tr>
<td>copySign(x, y)</td>
<td>Returns the first floating point x with the sign of the second floating point y</td>
<td>double</td>
</tr>
<tr>
<td>cos(x)</td>
<td>Returns the cosine of x (x is in radians)</td>
<td>double</td>
</tr>
<tr>
<td>cosh(x)</td>
<td>Returns the hyperbolic cosine of a double value</td>
<td>double</td>
</tr>
<tr>
<td>exp(x)</td>
<td>Returns the value of E^x</td>
<td>double</td>
</tr>
<tr>
<td>expm1(x)</td>
<td>Returns e^x - 1</td>
<td>double</td>
</tr>
<tr>
<td>floor(x)</td>
<td>Returns the value of x rounded down to its nearest integer</td>
<td>double</td>
</tr>
<tr>
<td>getExponent(x)</td>
<td>Returns the unbiased exponent used in x</td>
<td>int</td>
</tr>
<tr>
<td>hypot(x, y)</td>
<td>Returns sqrt(x^2 + y^2) without intermediate overflow or underflow</td>
<td>double</td>
</tr>
<tr>
<td>IEEEremainder(x, y)</td>
<td>Computes the remainder operation on x and y as prescribed by the IEEE 754 standard</td>
<td>double</td>
</tr>
<tr>
<td>log(x)</td>
<td>Returns the natural logarithm (base E) of x</td>
<td>double</td>
</tr>
<tr>
<td>Function</td>
<td>Description</td>
<td>Return Type</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>log10(x)</td>
<td>Returns the base 10 logarithm of x</td>
<td>double</td>
</tr>
<tr>
<td>log1p(x)</td>
<td>Returns the natural logarithm (base E) of the sum of x and 1</td>
<td>double</td>
</tr>
<tr>
<td>max(x, y)</td>
<td>Returns the number with the highest value</td>
<td>double</td>
</tr>
<tr>
<td>min(x, y)</td>
<td>Returns the number with the lowest value</td>
<td>double</td>
</tr>
<tr>
<td>nextAfter(x, y)</td>
<td>Returns the floating point number adjacent to x in the direction of y</td>
<td>double</td>
</tr>
<tr>
<td>nextUp(x)</td>
<td>Returns the floating point value adjacent to x in the direction of positive infinity</td>
<td>double</td>
</tr>
<tr>
<td>pow(x, y)</td>
<td>Returns the value of x to the power of y</td>
<td>double</td>
</tr>
<tr>
<td>random()</td>
<td>Returns a random number between 0 and 1</td>
<td>double</td>
</tr>
<tr>
<td>round(x)</td>
<td>Returns the value of x rounded to its nearest integer</td>
<td>int</td>
</tr>
<tr>
<td>rint()</td>
<td>Returns the double value that is closest to x and equal to a mathematical integer</td>
<td>double</td>
</tr>
<tr>
<td>signum(x)</td>
<td>Returns the sign of x</td>
<td>double</td>
</tr>
<tr>
<td>sin(x)</td>
<td>Returns the sine of x (x is in radians)</td>
<td>double</td>
</tr>
<tr>
<td>sinh(x)</td>
<td>Returns the hyperbolic sine of a double value</td>
<td>double</td>
</tr>
<tr>
<td>sqrt(x)</td>
<td>Returns the square root of x</td>
<td>double</td>
</tr>
<tr>
<td>tan(x)</td>
<td>Returns the tangent of an angle</td>
<td>double</td>
</tr>
</tbody>
</table>
### Java Math Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Return Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>tanh(x)</code></td>
<td>Returns the hyperbolic tangent of a double value</td>
<td>double</td>
</tr>
<tr>
<td><code>toDegrees(x)</code></td>
<td>Converts an angle measured in radians to an approx. equivalent angle measured in degrees</td>
<td>double</td>
</tr>
<tr>
<td><code>toRadians(x)</code></td>
<td>Converts an angle measured in degrees to an approx. angle measured in radians</td>
<td>double</td>
</tr>
<tr>
<td><code>ulp(x)</code></td>
<td>Returns the size of the unit of least precision (ulp) of x</td>
<td>double</td>
</tr>
</tbody>
</table>

### Java String Methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
<th>Return Type</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>charAt()</code></td>
<td>Returns the character at the specified index (position)</td>
<td>char</td>
</tr>
<tr>
<td><code>codePointAt()</code></td>
<td>Returns the Unicode of the character at the specified index</td>
<td>int</td>
</tr>
<tr>
<td><code>codePointBefore()</code></td>
<td>Returns the Unicode of the character before the specified index</td>
<td>int</td>
</tr>
<tr>
<td><code>codePointCount()</code></td>
<td>Returns the Unicode in the specified text range of this String</td>
<td>int</td>
</tr>
<tr>
<td><code>compareTo()</code></td>
<td>Compares two strings lexicographically</td>
<td>int</td>
</tr>
<tr>
<td><code>compareToIgnoreCase()</code></td>
<td>Compares two strings lexicographically, ignoring case differences</td>
<td>int</td>
</tr>
<tr>
<td><code>concat()</code></td>
<td>Appends a string to the end of another string</td>
<td>String</td>
</tr>
<tr>
<td><code>contains()</code></td>
<td>Checks whether a string contains a sequence of characters</td>
<td>boolean</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td>Return Type</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>contentEquals()</td>
<td>Checks whether a string contains the exact same sequence of characters of the specified CharSequence or StringBuffer</td>
<td>boolean</td>
</tr>
<tr>
<td>copyValueOf()</td>
<td>Returns a String that represents the characters of the character array</td>
<td>String</td>
</tr>
<tr>
<td>endsWith()</td>
<td>Checks whether a string ends with the specified character(s)</td>
<td>boolean</td>
</tr>
<tr>
<td>equals()</td>
<td>Compares two strings. Returns true if the strings are equal, and false if not</td>
<td>boolean</td>
</tr>
<tr>
<td>equalsIgnoreCase()</td>
<td>Compares two strings, ignoring case considerations</td>
<td>boolean</td>
</tr>
<tr>
<td>format()</td>
<td>Returns a formatted string using the specified locale, format string, and arguments</td>
<td>String</td>
</tr>
<tr>
<td>getBytes()</td>
<td>Encodes this String into a sequence of bytes using the named charset, storing the result into a new byte array</td>
<td>byte[]</td>
</tr>
<tr>
<td>getChars()</td>
<td>Copies characters from a string to an array of chars</td>
<td>void</td>
</tr>
<tr>
<td>hashCode()</td>
<td>Returns the hash code of a string</td>
<td>int</td>
</tr>
<tr>
<td>indexOf()</td>
<td>Returns the position of the first found occurrence of specified characters in a string</td>
<td>int</td>
</tr>
<tr>
<td>intern()</td>
<td>Returns the index within this string of the first occurrence of the specified character, starting the search at the specified index</td>
<td>String</td>
</tr>
<tr>
<td>isEmpty()</td>
<td>Checks whether a string is empty or not</td>
<td>boolean</td>
</tr>
<tr>
<td>lastIndexOf()</td>
<td>Returns the position of the last found occurrence of specified characters in a string</td>
<td>int</td>
</tr>
<tr>
<td>Method</td>
<td>Description</td>
<td>Return Type</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>length()</td>
<td>Returns the length of a specified string</td>
<td>int</td>
</tr>
<tr>
<td>matches()</td>
<td>Searches a string for a match against a regular expression, and returns the matches</td>
<td>boolean</td>
</tr>
<tr>
<td>offsetByCodePoints()</td>
<td>Returns the index within this String that is offset from the given index by codePointOffset code points</td>
<td>int</td>
</tr>
<tr>
<td>regionMatches()</td>
<td>Tests if two string regions are equal</td>
<td>boolean</td>
</tr>
<tr>
<td>replace()</td>
<td>Searches a string for a specified value, and returns a new string where the specified values are replaced</td>
<td>String</td>
</tr>
<tr>
<td>replaceFirst()</td>
<td>Replaces the first occurrence of a substring that matches the given regular expression with the given replacement</td>
<td>String</td>
</tr>
<tr>
<td>replaceAll()</td>
<td>Replaces each substring of this string that matches the given regular expression with the given replacement</td>
<td>String</td>
</tr>
<tr>
<td>split()</td>
<td>Splits a string into an array of substrings</td>
<td>String[]</td>
</tr>
<tr>
<td>startsWith()</td>
<td>Checks whether a string starts with specified characters</td>
<td>boolean</td>
</tr>
<tr>
<td>subSequence()</td>
<td>Returns a new character sequence that is a subsequence of this sequence</td>
<td>CharSequence</td>
</tr>
<tr>
<td>substring()</td>
<td>Extracts the characters from a string, beginning at a specified start position, and through the specified number of character</td>
<td>String</td>
</tr>
<tr>
<td>toCharArray()</td>
<td>Converts this string to a new character array</td>
<td>char[]</td>
</tr>
<tr>
<td>toLowerCase()</td>
<td>Converts a string to lower case letters</td>
<td>String</td>
</tr>
<tr>
<td>toString()</td>
<td>Returns the value of a String object</td>
<td>String</td>
</tr>
<tr>
<td>toUpperCase()</td>
<td>Converts a string to upper case letters</td>
<td>String</td>
</tr>
<tr>
<td>trim()</td>
<td>Removes whitespace from both ends of a string</td>
<td>String</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>valueOf()</td>
<td>Returns the primitive value of a String object</td>
<td>String</td>
</tr>
</tbody>
</table>

**What is the difference between SQL and SQLite?**

- SQL (Structured Query Language) – a standard interactive and programming language for getting information from a database
- SQLite – database

**XML DOM Node Types**

**Node Types**

<table>
<thead>
<tr>
<th>Node Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document</td>
<td>Represents the entire document (the root-node of the DOM tree)</td>
</tr>
<tr>
<td>DocumentFragment</td>
<td>Represents a &quot;lightweight&quot; Document object, which can hold a portion of a document</td>
</tr>
<tr>
<td>DocumentType</td>
<td>Provides an interface to the entities defined for the document</td>
</tr>
<tr>
<td>ProcessingInstruction</td>
<td>Represents a processing instruction</td>
</tr>
<tr>
<td>EntityReference</td>
<td>Represents an entity reference</td>
</tr>
<tr>
<td>Node Type</td>
<td>nodeName returns</td>
</tr>
<tr>
<td>--------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Document</td>
<td>#document</td>
</tr>
<tr>
<td>DocumentFragment</td>
<td>#document fragment</td>
</tr>
<tr>
<td>DocumentType</td>
<td>doctype name</td>
</tr>
<tr>
<td>EntityReference</td>
<td>entity reference name</td>
</tr>
<tr>
<td>Element</td>
<td>element name</td>
</tr>
<tr>
<td>NodeType</td>
<td>Named Constant</td>
</tr>
<tr>
<td>------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>1</td>
<td>ELEMENT_NODE</td>
</tr>
<tr>
<td>2</td>
<td>ATTRIBUTE_NODE</td>
</tr>
<tr>
<td>3</td>
<td>TEXT_NODE</td>
</tr>
</tbody>
</table>

### NodeTypes - Named Constants
Glossary of computer programming

**abstract class**

A class that cannot be directly constructed, one that can be constructed only through construction of some of its subclasses.

**abstract type**

A type in a nominative type system that cannot be instantiated.

**actual argument**

A value, or reference to a value, passed to a function.
An application that executes on a small, handheld device.

A program or integrated suite of programs that has a defined function.

A value, or reference to a value, passed to a function; an actual argument.

A parameter in a function definition; a formal argument.

An ordered sequence of same-typed values whose elements are fast to access by their numerical index in the array.

A data type for yes or no, true or false values.

A set of objects having the same behavior (but typically differing in state), or a template defining such a set.

A computer program which transforms source code into object code.

An identifier that is bound to an invariant value.

A class method (in object-oriented programming) that creates and initializes each instance of an object.

A classification or category of various types of data, that states the possible values that can be taken, how they are stored, and what range of operations are allowed on them.

A computer program which transforms source code into object code.
In object-oriented programming, the command sequence that is launched when the execution of an object is finished.

dump

A formatted listing of the contents of program storage, especially when produced automatically by a failing program

enumeration

A data type whose values are a set of mutually exclusive named constants.

exception

An interruption in normal processing, especially as caused by an error condition.

floating point

A method of representing real numbers as a pair of integers (the mantissa and characteristic)

flow chart

A schematic representation of the logic that defines the flow of control through a program

formal argument

A parameter in a function definition.

function

A routine that receives zero or more arguments and may return a result.

functional programming

A programming paradigm that treats computation as the evaluation of mathematical functions, avoids state and mutable data, and makes it easy to construct functions as if they were data objects.

goto / go to

A statement (in source code) that transfers control unconditionally to another part of a program

heap
An area of memory reserved for dynamically allocated data objects, contrasted to the stack.

**identifier**

A formal name used in source code to refer to a variable, function, procedure, package, etc.

**integer**

A data type for integer values.

**interpreter**

A program which executes another program written in a programming language other than machine code.

**linker / link editor**

A computer program that takes one or more objects generated by compilers and assembles them into a single executable program.

**logic programming**

A style or paradigm of computer programming exemplified by the language Prolog.

**machine code**

System of instructions and data directly understandable by a computer's central processing unit.

**method**

In object-oriented languages, a subroutine or function belonging to a class or object.

**module**

A program that is linked with others to form a functioning application; one method of implementing a subroutine.

**object**

An instance of a class.

**object code**

The output of a compiler or assembler, not necessarily executable directly without linking to other modules.
**object-oriented**

Using entities called objects that can process data and exchange messages with other objects.

**paradigm**

A fundamental style of computer programming to which the design of a programming language typically has to cater, such as imperative programming, declarative programming, or, on a finer level, functional programming, logic programming or object-oriented programming.

**parameter**

A name in a function or subroutine definition that is replaced by, or bound to, the corresponding actual argument when the function or subroutine is called.

**procedure**

A subroutine or function coded to perform a specific task.

**program**

A software application, or a collection of software applications, designed to perform a specific task.

**real**

A number containing a decimal point, e.g. the number pi is a real number with a value of approximately 3.14159268.

**run time**

The time during which a program is executing, as oppose to the compile time.

**source code**

Human-readable instructions in a programming language, to be transformed into machine instructions by a compiler, interpreter, assembler or other such system.

**stack**

The portion of the computer memory used to keep track of called procedures or call instructions.

**string**

A data type for a sequence of characters such as letters of English alphabet.
subclass

In object-oriented programming, an object class derived from another class (its superclass) from which it inherits a base set of properties and methods.

subprogram

A program contained within a larger program.

subroutine

A section of code that implements a task. While it may be used at more than one point in a program, it need not be.

superclass

A class that passes attributes and methods down the hierarchy to subclasses.

type

A tag attached to variables and values used in determining what values may be assigned to what variables.

variable

A named memory location in which a program can store intermediate results and from which it can read them.

"Remember that code is really the language in which we ultimately express the requirements. We may create languages that are closer to the requirements. We may create tools that help us parse and assemble those requirements into formal structures. But we will never eliminate necessary precision—so there will always be code."

— Robert C. Martin
Java Design Patterns:

- **Creational design patterns** → design patterns that deal with the way of creating objects.
- **Structural design patterns** → design patterns that deal with how classes and objects can be composed to form larger structures.
- **Behavioral design patterns** → design patterns that deal with algorithms and the assignment of responsibilities between objects.
- **JEE Design Patterns** → design patterns that deal with providing solutions to the Java EE-based software applications and frameworks.

**Automation** → Reduces execution time

**Robotic Process Automation** → Reduces the number of people and automate task just like a human being

- Robotic Desktop automation
- Robotic Process automation

Process driven

- Machine learning
- Artificial intelligence

Data driven
Robotic Process Automation:

Identify → Analyze → Design → Develop → Test → Implement

Robotic process automation + Cognitive automation + Workforce Analytics → Intelligent Automation

<table>
<thead>
<tr>
<th>Type of Data Processed</th>
<th>Big Data Analytics</th>
<th>Data Science</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Structured</td>
<td>• Structured Data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Unstructured Data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Semi-structured Data</td>
</tr>
</tbody>
</table>

3 Components of Cloud Computing:

- Software-as-a-Service (SaaS)
- Infrastructure-as-a-Service (IaaS)
- Platform-as-a-Service (PaaS)

Artificial intelligence

Machine Learning

Symbolic logic: rules engines, explain systems, and knowledge

Self Learning and Adaptive systems
<table>
<thead>
<tr>
<th>Data Mining</th>
<th>Statistics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methodology</td>
<td>Inductive</td>
</tr>
<tr>
<td>Variables</td>
<td>Large</td>
</tr>
<tr>
<td>Used for</td>
<td>Exploration</td>
</tr>
</tbody>
</table>

- **Supervised Learning**
  - **Classification**
    - Fraud Detection
    - Email Spam Detection
    - Diagnosis
    - Image Classification
  - **Regression**
    - Risk Assignment
    - Score Prediction

- **Unsupervised Learning**
  - **Clustering**
    - Biology
    - City Planning
    - Targeted Marketing
  - **Dimensionality reduction**
    - Text Mining
    - Face Recognition
    - Big Data Visualization
    - Image Recognition
Reinforcement Learning

- Gaming
- Finance Sector
- Manufacturing
- Inventory
- Management
- Robot Navigation

<table>
<thead>
<tr>
<th>Keras</th>
<th>Tensorflow</th>
</tr>
</thead>
<tbody>
<tr>
<td>A simple Python library for Deep Learning</td>
<td>A set of libraries for Machine Learning</td>
</tr>
<tr>
<td>Easy to debug</td>
<td>Complex to debug</td>
</tr>
<tr>
<td>Used for small datasets</td>
<td>Used for massive datasets</td>
</tr>
</tbody>
</table>

Disadvantages of Cassandra:
- No support to Data Integration
- Large outputs must be physically paged

<table>
<thead>
<tr>
<th>Manual Testing</th>
<th>Automated Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less accurate</td>
<td>Highly reliable</td>
</tr>
<tr>
<td>Needs lots of time</td>
<td>Very fast</td>
</tr>
<tr>
<td>Test cases are run once or twice</td>
<td>Test cases are run repeatedly</td>
</tr>
<tr>
<td>Human intervention</td>
<td>No Human intervention</td>
</tr>
</tbody>
</table>
The main phases of risk based testing are:

- Identify risks
- Assess risks
- Mitigate risks

Data Science Life Cycle:

Step 1: Define the Problem

Step 2: Collect the Data

Step 3: Process the Data

Step 4: Explore the Data

Step 5: Analyze the Data

Step 6: Consolidate the Result

Tensorflow architecture consists of three parts:

- Data preprocessing
- Model building
- Model training and evaluation
## Node.js Built-in Modules

<table>
<thead>
<tr>
<th>Module</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>assert</td>
<td>Provides a set of assertion tests</td>
</tr>
<tr>
<td>buffer</td>
<td>To handle binary data</td>
</tr>
<tr>
<td>child_process</td>
<td>To run a child process</td>
</tr>
<tr>
<td>cluster</td>
<td>To split a single Node process into multiple processes</td>
</tr>
<tr>
<td>crypto</td>
<td>To handle OpenSSL cryptographic functions</td>
</tr>
<tr>
<td>dgram</td>
<td>Provides implementation of UDP datagram sockets</td>
</tr>
<tr>
<td>dns</td>
<td>To do DNS lookups and name resolution functions</td>
</tr>
<tr>
<td>domain</td>
<td>Deprecated. To handle unhandled errors</td>
</tr>
<tr>
<td>events</td>
<td>To handle events</td>
</tr>
<tr>
<td>fs</td>
<td>To handle the file system</td>
</tr>
<tr>
<td>http</td>
<td>To make Node.js act as an HTTP server</td>
</tr>
<tr>
<td>https</td>
<td>To make Node.js act as an HTTPS server.</td>
</tr>
<tr>
<td>net</td>
<td>To create servers and clients</td>
</tr>
<tr>
<td>Module</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>os</td>
<td>Provides information about the operation system</td>
</tr>
<tr>
<td>path</td>
<td>To handle file paths</td>
</tr>
<tr>
<td>punycode</td>
<td>Deprecated. A character encoding scheme</td>
</tr>
<tr>
<td>querystring</td>
<td>To handle URL query strings</td>
</tr>
<tr>
<td>readline</td>
<td>To handle readable streams one line at the time</td>
</tr>
<tr>
<td>stream</td>
<td>To handle streaming data</td>
</tr>
<tr>
<td>string_decoder</td>
<td>To decode buffer objects into strings</td>
</tr>
<tr>
<td>timers</td>
<td>To execute a function after a given number of milliseconds</td>
</tr>
<tr>
<td>tls</td>
<td>To implement TLS and SSL protocols</td>
</tr>
<tr>
<td>tty</td>
<td>Provides classes used by a text terminal</td>
</tr>
<tr>
<td>url</td>
<td>To parse URL strings</td>
</tr>
<tr>
<td>util</td>
<td>To access utility functions</td>
</tr>
<tr>
<td>v8</td>
<td>To access information about V8 (the JavaScript engine)</td>
</tr>
<tr>
<td>vm</td>
<td>To compile JavaScript code in a virtual machine</td>
</tr>
</tbody>
</table>
### Complete List of All Bootstrap Classes

<table>
<thead>
<tr>
<th>Class</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>.active</td>
<td>Adds a grey background color to the table row (\texttt{&lt;tr&gt;} or table cell (\texttt{&lt;td&gt;}) (same color used on hover)</td>
</tr>
<tr>
<td>.active</td>
<td>Adds a gray background color to the active link in a default \texttt{navbar}. Adds a black background and a white color to the current link inside an inverted \texttt{navbar}.</td>
</tr>
<tr>
<td>.active</td>
<td>Adds a blue background color to the active list item in a list group</td>
</tr>
<tr>
<td>.active</td>
<td>Adds a blue background color to simulate a &quot;pressed&quot; \texttt{button}</td>
</tr>
<tr>
<td>.active</td>
<td>Animates a striped \texttt{progress bar}</td>
</tr>
<tr>
<td>.active</td>
<td>Adds a blue background color to the active \texttt{dropdown item} in a dropdown</td>
</tr>
<tr>
<td>.active</td>
<td>Adds a blue background color to the active \texttt{pagination} link (to highlight the current page)</td>
</tr>
<tr>
<td>.affix</td>
<td>The Affix plugin allows an element to become affixed (locked/sticky) to an area on the page. It \texttt{toggles} \texttt{position:fixed} on and off</td>
</tr>
<tr>
<td>.alert</td>
<td>Creates an alert message box</td>
</tr>
<tr>
<td>.alert-danger</td>
<td>Red alert box. Indicates a dangerous or potentially negative action</td>
</tr>
<tr>
<td>.alert-dismissible</td>
<td>Together with the \texttt{.close} class, this class is used to close the alert</td>
</tr>
<tr>
<td>.alert-info</td>
<td>Light-blue alert box. Indicates some information</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.alert-link</td>
<td>Used on links inside alerts to add matching colored links</td>
</tr>
<tr>
<td>.alert-success</td>
<td>Green alert box. Indicates a successful or positive action</td>
</tr>
<tr>
<td>.alert-warning</td>
<td>Yellow alert box. Indicates caution should be taken with this action</td>
</tr>
<tr>
<td>.badge</td>
<td>Creates a circular badge (grey circle - often used as a numerical indicator)</td>
</tr>
<tr>
<td>.bg-danger</td>
<td>Adds a red background color to an element. Represents danger or a negative action</td>
</tr>
<tr>
<td>.bg-info</td>
<td>Adds a light-blue background color to an element. Represents some information</td>
</tr>
<tr>
<td>.bg-primary</td>
<td>Adds a blue background color to an element. Represents something important</td>
</tr>
<tr>
<td>.bg-success</td>
<td>Adds a green background color to an element. Indicates success or a positive action</td>
</tr>
<tr>
<td>.bg-warning</td>
<td>Adds a yellow background color to an element. Represents a warning or a negative action</td>
</tr>
<tr>
<td>.breadcrumb</td>
<td>A pagination. Indicates the current page's location within a navigational hierarchy</td>
</tr>
<tr>
<td>.btn</td>
<td>Creates a basic button (gray background and rounded corners)</td>
</tr>
<tr>
<td>.btn-block</td>
<td>Creates a block level button that spans the entire width of the parent element</td>
</tr>
<tr>
<td>.btn-danger</td>
<td>Red button. Indicates danger or a negative action</td>
</tr>
<tr>
<td>.btn-default</td>
<td>Default button. White background and grey border</td>
</tr>
<tr>
<td>.btn-group</td>
<td>Groups buttons together on a single line</td>
</tr>
<tr>
<td>.btn-group-justified</td>
<td>Makes a group of buttons span the entire width of the screen</td>
</tr>
<tr>
<td>.btn-group-lg</td>
<td>Large button group (makes all buttons in a button group larger - increased font-size and padding)</td>
</tr>
<tr>
<td>.btn-group-sm</td>
<td>Small button group (makes all buttons in a button group smaller)</td>
</tr>
<tr>
<td>.btn-group-xs</td>
<td>Extra small button group (makes all buttons in a button group extra small)</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.btn-group-vertical</td>
<td>Makes a button group appear vertically stacked</td>
</tr>
<tr>
<td>.btn-info</td>
<td>Light-blue button. Represents information</td>
</tr>
<tr>
<td>.btn-link</td>
<td>Makes a button look like a link (get button behavior)</td>
</tr>
<tr>
<td>.btn-lg</td>
<td>Large button</td>
</tr>
<tr>
<td>.btn-primary</td>
<td>Blue button.</td>
</tr>
<tr>
<td>.btn-sm</td>
<td>Small button</td>
</tr>
<tr>
<td>.btn-success</td>
<td>Green button. Indicates success or a positive action</td>
</tr>
<tr>
<td>.btn-warning</td>
<td>Yellow button. Represents warning or a negative action</td>
</tr>
<tr>
<td>.btn-xs</td>
<td>Extra small button</td>
</tr>
<tr>
<td>.caption</td>
<td>Adds a caption text inside a .thumbnail</td>
</tr>
<tr>
<td>.caret</td>
<td>Creates a caret arrow icon, which indicates that the button is a dropdown</td>
</tr>
<tr>
<td>.carousel</td>
<td>Creates a carousel (slideshow)</td>
</tr>
<tr>
<td>.carousel-caption</td>
<td>Creates a caption text for each slide in the carousel</td>
</tr>
<tr>
<td>.carousel-control</td>
<td>Container for next and previous links</td>
</tr>
<tr>
<td>.carousel-indicators</td>
<td>Adds little dots/indicators at the bottom of each slide (which indicates how many slides there is in the carousel, and which slide the user are currently viewing)</td>
</tr>
<tr>
<td>.carousel-inner</td>
<td>Container for slide items</td>
</tr>
<tr>
<td>.center-block</td>
<td>Centers any element (Sets an element to display:block with margin-right:auto and margin-left:auto)</td>
</tr>
<tr>
<td>.checkbox</td>
<td>Container for checkboxes</td>
</tr>
<tr>
<td>.checkbox-inline</td>
<td>Makes multiple checkboxes appear on the same line</td>
</tr>
<tr>
<td>.clearfix</td>
<td>Clears floats</td>
</tr>
<tr>
<td>.close</td>
<td>Indicates a close icon</td>
</tr>
<tr>
<td>.col-<em>-</em></td>
<td>Responsive grid (span 1-12 column). Extra small devices Phones (&lt; 768px), Small devices Tablets (≥768px), Medium devices</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>.col-<em>-offset-</em></td>
<td>Move columns to the right. These classes increase the left margin of a column by * columns</td>
</tr>
<tr>
<td>.col-<em>-pull-</em></td>
<td>Changes the order of the grid columns</td>
</tr>
<tr>
<td>.col-<em>-push-</em></td>
<td>Changes the order of the grid columns</td>
</tr>
<tr>
<td>.collapse</td>
<td>Indicates collapsible content - which can be hidden or shown on demand</td>
</tr>
<tr>
<td>.collapse in</td>
<td>Show the collapsible content by default</td>
</tr>
<tr>
<td>.container</td>
<td>Fixed width container with widths determined by screen sites. Equal margin on the left and right.</td>
</tr>
<tr>
<td>.container-fluid</td>
<td>A container that spans the full width of the screen</td>
</tr>
<tr>
<td>.control-label</td>
<td>Allows a label to be used for form validation</td>
</tr>
<tr>
<td>.danger</td>
<td>Adds a red background to the table row (&lt;tr&gt; or table cell (&lt;td&gt;). Indicates a dangerous or potentially negative action</td>
</tr>
<tr>
<td>.disabled</td>
<td>Disables a button (adds opacity and a &quot;no-parking-sign&quot; icon on hover)</td>
</tr>
<tr>
<td>.disabled</td>
<td>Disables a dropdown item (adds a grey text color and a &quot;no-parking-sign&quot; icon on hover)</td>
</tr>
<tr>
<td>.disabled</td>
<td>Disables a pagination link (cannot be clicked - adds a grey text color and a &quot;no-parking-sign&quot; icon on hover)</td>
</tr>
<tr>
<td>.disabled</td>
<td>Disables a list item in a list group (cannot be clicked - adds a grey background color and a &quot;no-parking-sign&quot; icon on hover)</td>
</tr>
<tr>
<td>.divider</td>
<td>Used to separate links in the dropdown menu with a thin horizontal border</td>
</tr>
<tr>
<td>.dl-horizontal</td>
<td>Lines up the terms &lt;dt&gt; and descriptions &lt;dd&gt; in &lt;dl&gt; elements side-by-side. Starts off like default &lt;dl&gt;s, but when the browser window expands, it will line up side-by-side</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.dropdown</td>
<td>Creates a toggleable menu that allows the user to choose one value from a predefined list</td>
</tr>
<tr>
<td>.dropdown-header</td>
<td>Used to add headers inside the dropdown menu</td>
</tr>
<tr>
<td>.dropdown-menu</td>
<td>Adds the default styles for the dropdown menu container</td>
</tr>
<tr>
<td>.dropdown-menu-right</td>
<td>Right-aligns a dropdown menu</td>
</tr>
<tr>
<td>.dropdown-toggle</td>
<td>Used on the button that should hide and show (toggle) the dropdown menu</td>
</tr>
<tr>
<td>.dropup</td>
<td>Indicates a dropup menu (upwards instead of downwards)</td>
</tr>
<tr>
<td>.embed-responsive</td>
<td>Container for embedded content. Makes videos or slideshows scale properly on any device</td>
</tr>
<tr>
<td>.embed-responsive-16by9</td>
<td>Container for embedded content. Creates an 16:9 aspect ratio embedded content</td>
</tr>
<tr>
<td>.embed-responsive-4by3</td>
<td>Container for embedded content. Creates an 4:3 aspect ratio embedded content</td>
</tr>
<tr>
<td>.embed-responsive-item</td>
<td>Used inside .embed-responsive. Scales the video nicely to the parent element</td>
</tr>
<tr>
<td>.fade</td>
<td>Adds a fading effect when closing an alert box</td>
</tr>
<tr>
<td>.form-control</td>
<td>Used on input, textarea, and select elements to span the entire width of the page and make them responsive</td>
</tr>
<tr>
<td>.form-control-feedback</td>
<td>Form validation class</td>
</tr>
<tr>
<td>.form-control-static</td>
<td>Adds plain text next to a form label within a horizontal form</td>
</tr>
<tr>
<td>.form-group</td>
<td>Container for form input and label</td>
</tr>
<tr>
<td>.form-inline</td>
<td>Makes a &lt;form&gt; left-aligned with inline-block controls (This only applies to forms within viewports that are at least 768px wide)</td>
</tr>
<tr>
<td>.form-horizontal</td>
<td>Aligns labels and groups of form controls in a horizontal layout</td>
</tr>
<tr>
<td>.glyphicon</td>
<td>Creates an icon. Bootstrap provides 260 free glyphicons from the Glyphicons Halflings set</td>
</tr>
<tr>
<td>.has-danger</td>
<td>Adds a red color to the label and a red border to the input, as well as an error icon inside the input (used together with .has-</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.has-feedback</td>
<td>Adds feedback icons for inputs (checkmark, warning and error signs)</td>
</tr>
<tr>
<td>.has-success</td>
<td>Adds a green color to the label and a green border to the input, as well as</td>
</tr>
<tr>
<td></td>
<td>a checkmark icon inside the input (used together with .has-feedback)</td>
</tr>
<tr>
<td>.has-warning</td>
<td>Adds a yellow/orange color to the label and a yellow/orange border to the</td>
</tr>
<tr>
<td></td>
<td>input, as well as a checkmark icon inside the input (used together with</td>
</tr>
<tr>
<td></td>
<td>.has-feedback)</td>
</tr>
<tr>
<td>.help-block</td>
<td>A block of help text that breaks onto a new line and may extend beyond one</td>
</tr>
<tr>
<td></td>
<td>line.</td>
</tr>
<tr>
<td>.hidden</td>
<td>Forces an element to be hidden (display:none)</td>
</tr>
<tr>
<td>.hidden-*</td>
<td>Hides content depending on screen size</td>
</tr>
<tr>
<td>.hide</td>
<td>Deprecated. Use .hidden instead</td>
</tr>
<tr>
<td>.h1 - .h6</td>
<td>Makes an element look like a heading of the chosen class (h1-h6)</td>
</tr>
<tr>
<td>.icon-bar</td>
<td>Used in the navbar to create a hamburger menu (three horizontal bars)</td>
</tr>
<tr>
<td>.icon-next</td>
<td>Unicode icon (arrow pointing right), used in carousels. This is often replaced with a glyphicon</td>
</tr>
<tr>
<td>.icon-prev</td>
<td>Unicode icon (arrow pointing left), used in carousels. This is often replaced with a glyphicon</td>
</tr>
<tr>
<td>.img-circle</td>
<td>Shapes an image to a circle (not supported in IE8 and earlier)</td>
</tr>
<tr>
<td>.img-responsive</td>
<td>Makes an image responsive</td>
</tr>
<tr>
<td>.img-rounded</td>
<td>Adds rounded corners to an image</td>
</tr>
<tr>
<td>.img-thumbnail</td>
<td>Shapes an image to a thumbnail (borders)</td>
</tr>
<tr>
<td>.in</td>
<td>Fades in tabs</td>
</tr>
<tr>
<td>.info</td>
<td>Adds a light-blue background to the table row (&lt;tr&gt; or table cell (&lt;td&gt;),</td>
</tr>
<tr>
<td></td>
<td>Indicates a neutral informative change or action</td>
</tr>
<tr>
<td>.initialism</td>
<td>Displays the text inside an &lt;abbr&gt; element in a slightly smaller</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.input-group</td>
<td>Container to enhance an input by adding an icon, text or a button in front or behind it as a “help text”</td>
</tr>
<tr>
<td>.input-group-lg</td>
<td>Large input group</td>
</tr>
<tr>
<td>.input-group-sm</td>
<td>Small input group</td>
</tr>
<tr>
<td>.input-group-addon</td>
<td>Together with the .input-group class, this class makes it possible to add an icon or help text next to the input field</td>
</tr>
<tr>
<td>.input-group-btn</td>
<td>Together with the .input-group class, this class attaches a button next to an input. Often used as a search bar</td>
</tr>
<tr>
<td>.input-lg</td>
<td>Large input field</td>
</tr>
<tr>
<td>.input-sm</td>
<td>Small input field</td>
</tr>
<tr>
<td>.invisible</td>
<td>Makes an element invisible (visibility:hidden). <strong>Note:</strong> Even though the element is invisible, it will take up space on the page</td>
</tr>
<tr>
<td>.item</td>
<td>Class added to each carousel item. May be text or images</td>
</tr>
<tr>
<td>.jumbotron</td>
<td>Creates a padded grey box with rounded corners that enlarges the font sizes of the text inside it. Creates a big box for calling extra attention to some special content or information</td>
</tr>
<tr>
<td>.label</td>
<td>Adds a grey rounded box to an element. Provides additional information about something (e.g. &quot;New&quot;)</td>
</tr>
<tr>
<td>.label-danger</td>
<td>Red label</td>
</tr>
<tr>
<td>.label-info</td>
<td>Light-blue label</td>
</tr>
<tr>
<td>.label-success</td>
<td>Green label</td>
</tr>
<tr>
<td>.label-warning</td>
<td>Yellow label</td>
</tr>
<tr>
<td>.lead</td>
<td>Increase the font size and line height of a paragraph</td>
</tr>
<tr>
<td>.left</td>
<td>Used to identify the left carousel control</td>
</tr>
<tr>
<td>.list-group</td>
<td>Creates a bordered list group for &lt;li&gt; elements</td>
</tr>
<tr>
<td>.list-group-item</td>
<td>Added to each &lt;li&gt; element in the list group</td>
</tr>
<tr>
<td>.list-group-item-heading</td>
<td>Creates a list group heading (used on other elements)</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.list-group-item-text</td>
<td>Used for item text inside the list group (used on other elements besides <code>&lt;li&gt;</code>)</td>
</tr>
<tr>
<td>.list-group-item-danger</td>
<td>Red background color for a list item in a list group</td>
</tr>
<tr>
<td>.list-group-item-info</td>
<td>Light-blue background color for a list item in a list group</td>
</tr>
<tr>
<td>.list-group-item-success</td>
<td>Green background color for a list item in a list group</td>
</tr>
<tr>
<td>.list-group-item-warning</td>
<td>Yellow background color for a list item in a list group</td>
</tr>
<tr>
<td>.list-inline</td>
<td>Places all list items on a single line (horizontal menu)</td>
</tr>
<tr>
<td>.list-unstyled</td>
<td>Removes all default list-style (bullets, left margin, etc.) styling from a <code>&lt;ul&gt;</code> or <code>&lt;ol&gt;</code> list</td>
</tr>
<tr>
<td>.mark</td>
<td>Highlights text: Highlighted text</td>
</tr>
<tr>
<td>.media</td>
<td>Aligns media objects (like images or videos - often used for comments in a blog post etc)</td>
</tr>
<tr>
<td>.media-body</td>
<td>Text that should appear next to a media object</td>
</tr>
<tr>
<td>.media-heading</td>
<td>Creates a heading inside the media object</td>
</tr>
<tr>
<td>.media-list</td>
<td>Nested media lists</td>
</tr>
<tr>
<td>.media-object</td>
<td>Indicates a media object (image or video)</td>
</tr>
<tr>
<td>.modal</td>
<td>Identifies the content as a modal and brings focus to it</td>
</tr>
<tr>
<td>.modal-body</td>
<td>Defines the style for the body of the modal. Add any HTML markup here (p, img, etc)</td>
</tr>
<tr>
<td>.modal-content</td>
<td>Styles the modal (border, background-color, etc). Inside this, add the modal's header, body and footer, if needed</td>
</tr>
<tr>
<td>.modal-dialog</td>
<td>Sets the proper width and margin of the modal</td>
</tr>
<tr>
<td>.modal-footer</td>
<td>The footer of the modal (often contains an action button and a close button)</td>
</tr>
<tr>
<td>.modal-header</td>
<td>The header of the modal (often contains a title and a close button)</td>
</tr>
<tr>
<td>.modal-lg</td>
<td>Large modal (wider than default)</td>
</tr>
<tr>
<td>.modal-open</td>
<td>Used on the <code>&lt;body&gt;</code> element to prevent page scrolling</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.modal-sm</td>
<td>Small modal (less width)</td>
</tr>
<tr>
<td>.modal-title</td>
<td>The title of the modal</td>
</tr>
<tr>
<td>.nav nav-tabs</td>
<td>Indicates a tabbed menu</td>
</tr>
<tr>
<td>.nav nav-pills</td>
<td>Indicates a pill menu</td>
</tr>
<tr>
<td>.nav .navbar-nav</td>
<td>Used on a <code>&lt;ul&gt;</code> container that contains the list items with links inside a navigation bar</td>
</tr>
<tr>
<td>.nav-justified</td>
<td>Centers tabs/pills. Note that on screens smaller than 768px the items are stacked (content will remain centered)</td>
</tr>
<tr>
<td>.nav-stacked</td>
<td>Vertically stack tabs or pills</td>
</tr>
<tr>
<td>.nav-tabs</td>
<td>Creates a tabbed menu</td>
</tr>
<tr>
<td>.navbar</td>
<td>Creates a navigation bar</td>
</tr>
<tr>
<td>.navbar-brand</td>
<td>Added to a link or a header element inside the navbar to represent a logo or a header</td>
</tr>
<tr>
<td>.navbar-btn</td>
<td>Vertically aligns a button inside a navbar</td>
</tr>
<tr>
<td>.navbar-collapse</td>
<td>Collapses the navbar (hidden and replaced with a menu/hamburger icon on mobile phones and small tablets)</td>
</tr>
<tr>
<td>.navbar-default</td>
<td>Creates a default navigation bar (light-grey background color)</td>
</tr>
<tr>
<td>.navbar-fixed-bottom</td>
<td>Makes the navbar stay at the bottom of the screen (sticky/fixed)</td>
</tr>
<tr>
<td>.navbar-fixed-top</td>
<td>Makes the navbar stay at the top of the screen (sticky/fixed)</td>
</tr>
<tr>
<td>.navbar-form</td>
<td>Added to form elements inside the navbar to vertically center them (proper padding)</td>
</tr>
<tr>
<td>.navbar-header</td>
<td>Added to a container element that contains the link/element that represent a logo or a header</td>
</tr>
<tr>
<td>.navbar-inverse</td>
<td>Creates a black navigation bar (instead of light-grey)</td>
</tr>
<tr>
<td>.navbar-left</td>
<td>Aligns nav links, forms, buttons, or text, in the navbar to the left</td>
</tr>
<tr>
<td>.navbar-link</td>
<td>Styles an element to look like a link inside the navbar (anchors get proper padding and an underline on hover, while other elements like p or span gets a default hover effect - white color)</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.navbar-nav</td>
<td>Used on a <code>&lt;ul&gt;</code> container that contains the list items with links inside a navigation bar</td>
</tr>
<tr>
<td>.navbar-right</td>
<td>Aligns nav links, forms, buttons, or text in the navbar to the right.</td>
</tr>
<tr>
<td>.navbar-static-top</td>
<td>Removes left, top and right borders (rounded corners) from the navbar (default navbar has a gray border and a 4px border-radius by default)</td>
</tr>
<tr>
<td>.navbar-text</td>
<td>Vertical align any elements inside the navbar that are not links (ensures proper padding)</td>
</tr>
<tr>
<td>.navbar-toggle</td>
<td>Styles the button that should open the navbar on small screens. Often used together with three <code>.icon-bar</code> classes to indicate a toggleable menu icon (hamburger/bars)</td>
</tr>
<tr>
<td>.next</td>
<td>Used in the carousel control to identify the next control</td>
</tr>
<tr>
<td>.next</td>
<td>Used to align pager buttons to the right side of the page (next button)</td>
</tr>
<tr>
<td>.page-header</td>
<td>Adds a horizontal line under the heading (+ adds some extra space around the element)</td>
</tr>
<tr>
<td>.pager</td>
<td>Creates previous/next buttons (used on <code>&lt;ul&gt;</code> elements)</td>
</tr>
<tr>
<td>.pagination</td>
<td>Creates a pagination (Useful when you have a web site with lots of pages. Used on <code>&lt;ul&gt;</code> elements)</td>
</tr>
<tr>
<td>.pagination-lg</td>
<td>Large pagination (each pagination link gets a font-size of 18px. Default is 14px)</td>
</tr>
<tr>
<td>.pagination-sm</td>
<td>Small pagination (each pagination link gets a font-size of 12px. Default is 14px)</td>
</tr>
<tr>
<td>.panel</td>
<td>Creates a bordered box with some padding around its content</td>
</tr>
<tr>
<td>.panel-body</td>
<td>Container for content inside the panel</td>
</tr>
<tr>
<td>.panel-collapse</td>
<td>Collapsible panel (toggle between hiding and showing panel(s))</td>
</tr>
<tr>
<td>.panel-danger</td>
<td>Red panel. Indicates danger</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.panel-info</td>
<td>Light-blue panel. Indicates information</td>
</tr>
<tr>
<td>.panel-success</td>
<td>Green panel. Indicates success</td>
</tr>
<tr>
<td>.panel-warning</td>
<td>Yellow panel. Indicates warning</td>
</tr>
<tr>
<td>.panel-footer</td>
<td>Creates a panel footer (light background color)</td>
</tr>
<tr>
<td>.panel-group</td>
<td>Used to group many panels together. This removes the bottom margin below each panel</td>
</tr>
<tr>
<td>.panel-heading</td>
<td>Creates a panel header (light background color)</td>
</tr>
<tr>
<td>.panel-title</td>
<td>Used inside a .panel-heading to adjust the styling of the text (removes margins and adds a font-size of 16px)</td>
</tr>
<tr>
<td>.popover</td>
<td>Popup-box that appears when the user clicks on an element</td>
</tr>
<tr>
<td>.pre-scrollable</td>
<td>Makes a <code>&lt;pre&gt;</code> element scrollable (max-height of 350px and provide a y-axis scrollbar)</td>
</tr>
<tr>
<td>.prev</td>
<td>Used in carousels to indicate a &quot;previous&quot; link</td>
</tr>
<tr>
<td>.previous</td>
<td>Used to align pager buttons to the left side of the page (previous button)</td>
</tr>
<tr>
<td>.progress</td>
<td>Container for progress bars</td>
</tr>
<tr>
<td>.progress-bar</td>
<td>Creates a progress bar</td>
</tr>
<tr>
<td>.progress-bar-danger</td>
<td>Red progress bar. Indicates danger</td>
</tr>
<tr>
<td>.progress-bar-info</td>
<td>Light-blue progress bar. Indicates information</td>
</tr>
<tr>
<td>.progress-bar-striped</td>
<td>Creates a striped progress bar</td>
</tr>
<tr>
<td>.progress-bar-success</td>
<td>Green progress bar. Indicates success</td>
</tr>
<tr>
<td>.progress-bar-warning</td>
<td>Yellow progress bar. Indicates warning</td>
</tr>
<tr>
<td>.pull-left</td>
<td>Float an element to the left</td>
</tr>
<tr>
<td>.pull-right</td>
<td>Float an element to the right</td>
</tr>
<tr>
<td>.right</td>
<td>Used to identify the right carousel control</td>
</tr>
<tr>
<td>.row</td>
<td>Container for responsive columns</td>
</tr>
<tr>
<td>.row-no-gutters</td>
<td>Removes the gutters from a row and its columns</td>
</tr>
<tr>
<td>.show</td>
<td>Shows an element (display:block)</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.small</td>
<td>Creates a lighter, secondary text in any heading</td>
</tr>
<tr>
<td>.sr-only</td>
<td>Hides an element on all devices except for screen readers</td>
</tr>
<tr>
<td>.sr-only-focusable</td>
<td>Hides an element on all devices except for screen readers</td>
</tr>
<tr>
<td>.success</td>
<td>Adds a green background color to a table row (&lt;tr&gt; or table cell (&lt;td&gt;). Indicates success or a positive action</td>
</tr>
<tr>
<td>.tab-content</td>
<td>Used together with .tab-pane to creates toggleable/dynamic tabs/pills</td>
</tr>
<tr>
<td>.tab-pane</td>
<td>Used together with .tab-content to creates toggleable/dynamic tabs/pills</td>
</tr>
<tr>
<td>.table</td>
<td>Adds basic styling to a table (padding, bottom borders, etc)</td>
</tr>
<tr>
<td>.table-bordered</td>
<td>Adds borders on all sides of the table and cells</td>
</tr>
<tr>
<td>.table-condensed</td>
<td>Makes a table more compact by cutting cell padding in half</td>
</tr>
<tr>
<td>.table-hover</td>
<td>Creates a hoverable table (adds a grey background color on table rows on hover)</td>
</tr>
<tr>
<td>.table-responsive</td>
<td>Makes a table responsive (adds a horizontal scrollbar when needed)</td>
</tr>
<tr>
<td>.text-capitalize</td>
<td>Indicates capitalized text</td>
</tr>
<tr>
<td>.text-center</td>
<td>Center-aligns text</td>
</tr>
<tr>
<td>.text-danger</td>
<td>Red text color. Indicates danger</td>
</tr>
<tr>
<td>.text-hide</td>
<td>Hides text (helps replace an element's text content with a background image)</td>
</tr>
<tr>
<td>.text-info</td>
<td>Light-blue text color. Indicates information</td>
</tr>
<tr>
<td>.text-justify</td>
<td>Indicates justified text</td>
</tr>
<tr>
<td>.text-left</td>
<td>Aligns the text to the left</td>
</tr>
<tr>
<td>.text-lowercase</td>
<td>Changes text to lowercase</td>
</tr>
<tr>
<td>.text-muted</td>
<td>Grey text color</td>
</tr>
<tr>
<td>.text-nowrap</td>
<td>Prevents the text from wrapping</td>
</tr>
<tr>
<td>.text-primary</td>
<td>Blue text color</td>
</tr>
<tr>
<td>Class</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>.text-right</td>
<td>Aligns text to the right</td>
</tr>
<tr>
<td>.text-success</td>
<td>Green text color. Indicates success</td>
</tr>
<tr>
<td>.text-uppercase</td>
<td>Makes text uppercase</td>
</tr>
<tr>
<td>.text-warning</td>
<td>Yellow/orange text color. Indicates warning</td>
</tr>
<tr>
<td>.thumbnail</td>
<td>Adds a border around an element (often images or videos) to make it look like a thumbnail</td>
</tr>
<tr>
<td>.tooltip</td>
<td>Popup-box that appears when the user moves the mouse pointer over an element</td>
</tr>
<tr>
<td>.visible-*</td>
<td>Deprecated as of v3.2.0. Used to show and/or hide content by device. Note: Use .hidden-* instead</td>
</tr>
<tr>
<td>.visible-print-block</td>
<td>Displays the element (display:block) in print (pre)view</td>
</tr>
<tr>
<td>.visible-print-inline</td>
<td>Displays the element (display:inline) in print (pre)view</td>
</tr>
<tr>
<td>.visible-print-inline-block</td>
<td>Displays the element (display:inline-block) in print (pre)view</td>
</tr>
<tr>
<td>.hidden-print</td>
<td>Hides the element (display:none) in print (pre)view</td>
</tr>
<tr>
<td>.warning</td>
<td>Adds a yellow background color to the table row (&lt;tr&gt; or table cell (&lt;td&gt;). Indicates a warning</td>
</tr>
<tr>
<td>.well</td>
<td>Adds a rounded border around an element with a gray background color and some padding</td>
</tr>
<tr>
<td>.well-lg</td>
<td>Large well (more padding)</td>
</tr>
<tr>
<td>.well-sm</td>
<td>Small well (less padding)</td>
</tr>
</tbody>
</table>

**AngularJS References**

- **AngularJS Directives**

<table>
<thead>
<tr>
<th>Directive</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ng-app</td>
<td>Defines the root element of an application.</td>
</tr>
<tr>
<td>ng-bind</td>
<td>Binds the content of an HTML element to application data.</td>
</tr>
<tr>
<td>Tag</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ng-bind-html</td>
<td>Binds the innerHTML of an HTML element to application data, and also removes dangerous code from the HTML string.</td>
</tr>
<tr>
<td>ng-bind-template</td>
<td>Specifies that the text content should be replaced with a template.</td>
</tr>
<tr>
<td>ng-blur</td>
<td>Specifies a behavior on blur events.</td>
</tr>
<tr>
<td>ng-change</td>
<td>Specifies an expression to evaluate when content is being changed by the user.</td>
</tr>
<tr>
<td>ng-checked</td>
<td>Specifies if an element is checked or not.</td>
</tr>
<tr>
<td>ng-class</td>
<td>Specifies CSS classes on HTML elements.</td>
</tr>
<tr>
<td>ng-class-even</td>
<td>Same as ng-class, but will only take effect on even rows.</td>
</tr>
<tr>
<td>ng-class-odd</td>
<td>Same as ng-class, but will only take effect on odd rows.</td>
</tr>
<tr>
<td>ng-click</td>
<td>Specifies an expression to evaluate when an element is being clicked.</td>
</tr>
<tr>
<td>ng-cloak</td>
<td>Prevents flickering when your application is being loaded.</td>
</tr>
<tr>
<td>ng-controller</td>
<td>Defines the controller object for an application.</td>
</tr>
<tr>
<td>ng-copy</td>
<td>Specifies a behavior on copy events.</td>
</tr>
<tr>
<td>ng-csp</td>
<td>Changes the content security policy.</td>
</tr>
<tr>
<td>ng-cut</td>
<td>Specifies a behavior on cut events.</td>
</tr>
<tr>
<td>ng-dblclick</td>
<td>Specifies a behavior on double-click events.</td>
</tr>
<tr>
<td>ng-disabled</td>
<td>Specifies if an element is disabled or not.</td>
</tr>
<tr>
<td>ng-focus</td>
<td>Specifies a behavior on focus events.</td>
</tr>
<tr>
<td>ng-form</td>
<td>Specifies an HTML form to inherit controls from.</td>
</tr>
<tr>
<td>ng-hide</td>
<td>Hides or shows HTML elements.</td>
</tr>
<tr>
<td>ng-href</td>
<td>Specifies a url for the &lt;a&gt; element.</td>
</tr>
<tr>
<td>ng-if</td>
<td>Removes the HTML element if a condition is false.</td>
</tr>
<tr>
<td>ng-include</td>
<td>Includes HTML in an application.</td>
</tr>
<tr>
<td>ng-init</td>
<td>Defines initial values for an application.</td>
</tr>
<tr>
<td>ng-jq</td>
<td>Specifies that the application must use a library, like jQuery.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>ng-keydown</td>
<td>Specifies a behavior on keydown events.</td>
</tr>
<tr>
<td>ng-keypress</td>
<td>Specifies a behavior on keypress events.</td>
</tr>
<tr>
<td>ng-keyup</td>
<td>Specifies a behavior on keyup events.</td>
</tr>
<tr>
<td>ng-list</td>
<td>Converts text into a list (array).</td>
</tr>
<tr>
<td>ng-maxlength</td>
<td>Specifies the maximum number of characters allowed in the input field.</td>
</tr>
<tr>
<td>ng-minlength</td>
<td>Specifies the minimum number of characters allowed in the input field.</td>
</tr>
<tr>
<td>ng-model</td>
<td>Binds the value of HTML controls to application data.</td>
</tr>
<tr>
<td>ng-model-options</td>
<td>Specifies how updates in the model are done.</td>
</tr>
<tr>
<td>ng-mousedown</td>
<td>Specifies a behavior on mousedown events.</td>
</tr>
<tr>
<td>ng-mouseenter</td>
<td>Specifies a behavior on mouseenter events.</td>
</tr>
<tr>
<td>ng-mouseleave</td>
<td>Specifies a behavior on mouseleave events.</td>
</tr>
<tr>
<td>ng-mousemove</td>
<td>Specifies a behavior on mousemove events.</td>
</tr>
<tr>
<td>ng-mouseover</td>
<td>Specifies a behavior on mouseover events.</td>
</tr>
<tr>
<td>ng-mouseup</td>
<td>Specifies a behavior on mouseup events.</td>
</tr>
<tr>
<td>ng-non-bindable</td>
<td>Specifies that no data binding can happen in this element, or its children.</td>
</tr>
<tr>
<td>ng-open</td>
<td>Specifies the open attribute of an element.</td>
</tr>
<tr>
<td>ng-options</td>
<td>Specifies &lt;options&gt; in a &lt;select&gt; list.</td>
</tr>
<tr>
<td>ng-paste</td>
<td>Specifies a behavior on paste events.</td>
</tr>
<tr>
<td>ng-pluralize</td>
<td>Specifies a message to display according to en-us localization rules.</td>
</tr>
<tr>
<td>ng-readonly</td>
<td>Specifies the readonly attribute of an element.</td>
</tr>
<tr>
<td>ng-repeat</td>
<td>Defines a template for each data in a collection.</td>
</tr>
<tr>
<td>ng-required</td>
<td>Specifies the required attribute of an element.</td>
</tr>
<tr>
<td>ng-selected</td>
<td>Specifies the selected attribute of an element.</td>
</tr>
</tbody>
</table>
**AngularJS Directives on HTML Elements**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>AngularJS modifies the <code>&lt;a&gt;</code> element's default behaviors.</td>
</tr>
<tr>
<td>form</td>
<td>AngularJS modifies the <code>&lt;form&gt;</code> element's default behaviors.</td>
</tr>
<tr>
<td>input</td>
<td>AngularJS modifies the <code>&lt;input&gt;</code> element's default behaviors.</td>
</tr>
<tr>
<td>script</td>
<td>AngularJS modifies the <code>&lt;script&gt;</code> element's default behaviors.</td>
</tr>
<tr>
<td>select</td>
<td>AngularJS modifies the <code>&lt;select&gt;</code> element's default behaviors.</td>
</tr>
<tr>
<td>textarea</td>
<td>AngularJS modifies the <code>&lt;textarea&gt;</code> element's default behaviors.</td>
</tr>
</tbody>
</table>

**AngularJS Filters**

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>currency</td>
<td>Format a number to a currency format.</td>
</tr>
<tr>
<td>date</td>
<td>Format a date to a specified format.</td>
</tr>
<tr>
<td>filter</td>
<td>Select a subset of items from an array.</td>
</tr>
<tr>
<td>json</td>
<td>Format an object to a JSON string.</td>
</tr>
<tr>
<td>limitTo</td>
<td>Limits an array, or a string, into a specified number of</td>
</tr>
</tbody>
</table>
AngularJS Validation Properties

- $dirty
- $invalid
- $error

AngularJS Global API

Converting

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>angular.lowercase()</td>
<td>Converts a string to lowercase</td>
</tr>
<tr>
<td>angular.uppercase()</td>
<td>Converts a string to uppercase</td>
</tr>
<tr>
<td>angular.copy()</td>
<td>Creates a deep copy of an object or an array</td>
</tr>
<tr>
<td>angular.forEach()</td>
<td>Executes a function for each element in an object or array</td>
</tr>
</tbody>
</table>

Comparing
<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>angular.isArray()</td>
<td>Returns true if the reference is an array</td>
</tr>
<tr>
<td>angular.isDate()</td>
<td>Returns true if the reference is a date</td>
</tr>
<tr>
<td>angular.isDefined()</td>
<td>Returns true if the reference is defined</td>
</tr>
<tr>
<td>angular.isElement()</td>
<td>Returns true if the reference is a DOM element</td>
</tr>
<tr>
<td>angular.isFunction()</td>
<td>Returns true if the reference is a function</td>
</tr>
<tr>
<td>angular.isNumber()</td>
<td>Returns true if the reference is a number</td>
</tr>
<tr>
<td>angular.isObject()</td>
<td>Returns true if the reference is an object</td>
</tr>
<tr>
<td>angular.isString()</td>
<td>Returns true if the reference is a string</td>
</tr>
<tr>
<td>angular.isUndefined()</td>
<td>Returns true if the reference is undefined</td>
</tr>
<tr>
<td>angular.equals()</td>
<td>Returns true if two references are equal</td>
</tr>
</tbody>
</table>

- **JSON**

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>angular.fromJson()</td>
<td>Takes a JSON string and returns an JavaScript object</td>
</tr>
<tr>
<td>angular.toJson()</td>
<td>Takes a JavaScript object and returns a JSON string</td>
</tr>
</tbody>
</table>

- **Basic**

<table>
<thead>
<tr>
<th>API</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>angular.bootstrap()</td>
<td>Starts AngularJS manually</td>
</tr>
<tr>
<td>angular.element()</td>
<td>Wraps an HTML element as a jQuery element</td>
</tr>
<tr>
<td>angular.module()</td>
<td>Creates, registers, or retrieves an AngularJS module</td>
</tr>
</tbody>
</table>
9 Interesting Python Facts

In python we can return multiple values:

def XYZ():
    p = 3
    q = 2
    return p, q

a, b = XYZ()
print(a, b)

Output on the screen:

3 2

Allows Negative Indexing:

my_list = ['apple', 'orange', 'grapes']
print(my_list[-2])

Output on the screen:

orange

Combine Multiple Strings:

my_list = ['I', 'Love', 'Python']
print(''.join(my_list))

Output on the screen:

I Love Python

We can swap two objects in Python:
a = 3
b = 2

print('Before Swapping')
print(a, b)

a, b = b, a
print('After Swapping')
print(a, b)

**Output on the screen:**

Before Swapping
3 2
After Swapping
2 3

We can know about Python version:

```python
import sys
print("My Python version Number: {}".format(sys.version))
```

**Output on the screen:**

My Python version Number: 3.7.3 (default, Mar 27 2019, 17:13:21) [MSC v.1915 64 bit (AMD64)]

We can Store all values of List in new separate variables:

```python
x = [1, 2, 3]
a, b, c = x
print(a)
print(b)
print(c)
```

**Output on the screen:**

1
2
3
We can convert nested list into one list:

```python
import itertools
x = [[1, 2], [3, 4], [5, 6]]
print(list(itertools.chain.from_iterable(x)))
```

Output on the screen:

```
[1, 2, 3, 4, 5, 6]
```

We can transpose a Matrix:

```python
import numpy as np
x = np.matrix('5, 1; 14, 2')
y = x.transpose()
print(y)
```

Output on the screen:

```
[[ 5 14]
 [ 1  2]]
```

We can create small anonymous function:

```python
x = lambda a, b, c : a + b + c
print(x(5, 6, 3))
```

Output on the screen:

```
14
```
References:

- Let Us C by Yashavant Kanetkar.
- C for Dummies by Dan Gookin.
- C++: The complete reference by Herbert Schildt.
- Programming with C by Byron S. Gottfried.
- INTRODUCTION to JAVA by Jane Meyerowitz.
- Java 2: The Complete Reference by Herbert Schildt.
- Java For Dummies by Barry Burd.
- C PROGRAMMING NOTE by T K Rajan.
- An Introduction to the C Programming Language and Software Design by Tim Bailey.
- C++: A Beginners Guide, Teach Yourself C++ by Herbert Schildt.
- C, C++ & Java (www.w3schools.com).
- A programming with class: A C++ introduction to computer science by Kamin.
- JAVA hand book by Naughton.
- Teach yourself JAVA by O’Neil.
- AT & T Bell laboratories: The C programmer’s hand book.
- C++ Program Design by Cohoon.
- An introduction to object oriented programming with Java by WU THOMAS.
- Ashok N Kamthane, Programming and Data structures, Pearson Education.
• A TUTORIAL ON POINTERS AND ARRAYS IN C by Ted Jensen.
• Pure basic A beginners guide to Computer Programming by Gary Willoughby.
• C++ for dummies by Stephen Randy Davis.
• C to Java: Converting Pointers into References by Erik D. Demaine.
• Why C++ is not just an Object Oriented Programming Language by Bjarne Stroustrup.
• Java Programming: A Practical Approach by Xavier.
• Herb Schildt's Java Programming Cookbook by Herbert Schildt.
• The Java Programming Language by Arnold.
• Programming in C by Stephen G. Kochan.
• Why learn Linux? By Candace Hazlett
• 11 Reasons Why Linux Is Better Than Windows By Ankush Das
• Linux for Dummies By Richard Blum
• Linux: The Complete Reference By Richard Petersen

Source of information:

• https://www.wikipedia.org/