In [6]:

```
#As a Part of programming requirement, we have to store our data permanently for future purpose. for this requriment we
#should go for files.
#Files are vary common permanent storages areas to store our data.
```

In [7]:

```
#There are two types of files
# 1. Text Files - can be used to store character data
# 2. Binary Files - can be used to store binary data like images, videos, audiofiles etc...
```

In [8]:

```
#Opening a file:
# Before performing any operation like read or write on the file, first we have to open that file.
# for this we will use python inbuilt function open()
# But at the time of open we have to specify mode. which represents the purpose of opening file.
# Syntax: f=open(filename,mode)
```

In [9]:

```
#The allowed modes in python are:
# r => open an existing file for read operation. The file pointer is positioned at the beginning of the file. If the
#      specified file does not exist then we will get FileNotFoundError
# w => open an existing file for write operation. If the file already contains some data then it will be overrridden.
#       If the specified file is not available then this mode will create that file
# a => open an existing for append operation. It won't override existing data. If the specified file is not already
#       available then this mode will create a new file.
# r+ => To read and write data into the file. The previous data in the file will not be deleted. The file pointer
#        is placed at the beginning of the file.
# w+ => To write and read data. It will override existing data.
# a+ => To append and read data from the file. It won't override existing data.
# x =>  To open a file in exclusive creation mode for write operation. if the file already exists then we will get
#        FileExistsError.
```

In [10]:

```
# Closing a File:
# After completing our operations on the file, It is highly recommended to close the file. For this we want to use close()
# function.
# Syntax: f.close()
```

In [11]:

```
# various properties of a file object
# Once we opened a file, and we got that file object, we can get various details related to that file by using its
# properties.
# 1.name -> Name of opened file.
# 2.mode -> Mode in which the file opened
# 3.closed -> returns that boolean value indicates that file is closed or not.
# 4.readable -> returns that boolean value indicates that file is readable or not.
# 5.writable -> returns that boolean value indicates that file is writable or not.
```

In [16]:

```
f=open("aravind.txt",'w')
print("filename:",f.name)
print("filemode:",f.mode)
print("Is file readble:",f.readable())
print("Is file writable",f.writable())
print("Is file closed",f.closed)
f.close()
print("Is file closed",f.closed)
```

```
filename: aravind.txt
filemode: w
Is file readble: False
Is file writable True
Is file closed False
Is file closed True
```

In [17]:

```
# how we will write data to text files
# We can write character data to text files by using 2 methods
# 1.write(str)
# 2.writelines(listoflines)
```

In [18]:

```
f=open('aravind.txt','w')
f.write("aravind\n")
f.write("academy\n")
f.write("python\n")
print("Data written to the file successfully")
f.close()
```

```
Data written to the file successfully
```

In [19]:

```
#Note: In the above written program the data present in the file will be overridden everytime if we run the progr
am.Instead
#      of overriding if we want to append operation then we should open the file as follows
# Syntax : open('aravind.txt',"a")
f=open('aravind.txt',"a")
f.write("software\n")
print("data written to the file succesfully")
f.close()
```

```
data written to the file succesfully
```

In [20]:

```
#eg
f=open("aravind.txt",'w')
list=['python\n','java\n','c\n','c++\n']
f.writelines(list)
print("list of lines written to the file successfully")
f.close()
```

```
list of lines written to the file successfully
```

In [21]:

```
#Note: While writing data by using write() methods, compulsory we have to provide the line seperator(\n), otherwi
se total
#      data should written in a single line.
```

In [22]:

```
# Read character data from text files
# We can read character data from text file by using following read methods
# 1.read() -> To read total data from the file.
# 2.read(n) -> To read 'n' characters from the file.
# 3.readline() -> To read only one line.
# 4.readlines() -> To read all lines into a list.
```

In [23]:

```
# To read total data from the file.
f=open('aravind.txt','r')
data=f.read()
print(data)
f.close()
```

```
python
java
c
c++

```

In [24]:

```
# To read only first 10 characters
f=open('aravind.txt','r')
data=f.read(10)
print(data)
f.close()
```

```
python
jav
```

In [25]:

```
# To read data by line
f=open('aravind.txt','r')
line1=f.readline()
print(line1,end="")
line2=f.readline()
print(line2,end="")
line3=f.readline()
print(line3,end="")
f.close()
```

```
python
java
c
```

In [26]:

```
# To read all lines into list
f=open("aravind.txt",'r')
lines=f.readlines()
for line in lines:
    print(line,end='')
f.close()
```

```
python
java
c
c++
```

In [27]:

```
f=open('aravind.txt','r')
print(f.read(3))
print(f.readline())
print(f.read(4))
print("remaining data")
print(f.read())
```

```
pyt
hon

java
remaining data

c
c++

```

In [28]:

```
#The with statement
# The with statement can be used while opening a file. We can use this to group file operation statements within a block.
# The advantage of with statement is it will take care closing of a file, after completing of all operations automatically
# even in the case of exceptions also, and we are not required to close explicitly.
with open('aravind.txt',"w") as f:
    f.write("python\n")
    f.write('software\n')
    f.write('solutions\n')
    print("is file closed:",f.closed)
print("is file closed:",f.closed)
```

```
is file closed: False
is file closed: True
```

In [29]:

```
# The seek() and tell() methods
# tell():
# We can use tell() method to return current position of the cursor(filepointer) from the beginning of the file.
# The position(index) of the first character in files is zero just like string index.
f=open("aravind.txt","r")
print(f.tell())
print(f.read(2))
print(f.tell())
print(f.read(3))
print(f.tell())
```

```
0
py
2
tho
5
```

In [33]:

```
#seek():
# We can use seek() method to move cursor(filepointer) to specified location.
# Syntax: f.seek(offset,fromwhere)
# offset represents the no of positions
# The allowed values for second attribute(fromwhere) are
# 0--> From beginning of the file
# 1--> from current position
# 2--> from end of file
#Note: python 2 supports all three values but python3 supports only zero
data="learning python is very easy"
f=open('aravind.txt','w')
f.write(data)
with open('aravind.txt','r+') as f:
    text=f.read()
    print(text)
    print("The current cursor position:",f.tell())
    f.seek(19)
    print("The current cursor position",f.tell())
    f.write("difficult!")
    f.seek(0)
    text=f.read()
    print("Data after modification")
    print(text)
```

```
learning python is very easy
The current cursor position: 28
The current cursor position 19
Data after modification
learning python is difficult!
```

In [34]:

```
#how to check a particular file exists or not?
# we can use os library to get information about files in a computer
# os module has path sub module, which contains isFile() function to check whether file exists or not?
# Syntax: os.path.isfile(fname)
```

In [35]:

```
# write a program to check whether given file exists or not. if it is available print its content.
import os,sys
fname=input("Enter the file name:")
if os.path.isfile(fname):
    print("file exists",fname)
    f=open(fname,"r")
else:
    print("file not exists",fname)
    sys.exit(0)
print("The content of the file is:")
data=f.read()
print(data)
```

```
Enter the file name:aravind.txt
file exists aravind.txt
Thr content of the file is:
learning python is difficult!
```

In [36]:

```
#Note: sys.exit(0)--> to exit system without executing the rest of program.
#argument represents status code. 0 means normal termination and it is the default value
```

In [38]:

```
#program to print no of lines, words and characters present in the given file?
import os,sys
fname=input("Enter the filename:")
if os.path.isfile(fname):
    print("file exists:",fname)
    f=open(fname,"r")
else:
    print("file does not exist:",fname)
    sys.exit(0)
lcount=wcount=ccount=0
for line in f:
    lcount+=1
    ccount+=len(line)
    words=line.split()
    wcount+=len(words)
print("The no of lines:",lcount)
print("The no of characters:",ccount)
print("Thr no of words:",wcount)
```

```
Enter the filename:aravind.txt
file exists: aravind.txt
The no of lines: 1
The no of characters: 29
Thr no of words: 4
```

In [ ]:

```
```