

File-handling

Generally the programming languages are suitable for processing the data.

the Programming languages are suitable for storing the data because the programming languages program's memory is allocated in RAM.

RAM(Random Access Memory) is a volatile memory,once program execution is over/turnoff our computer then automatically to errase the data from that RAM.

if we are not storing the data it is not possible use that data in future.

if we want to store the permanently in a non-volatile memory,in that case we are using following concept's

- 1).File's
- 2).Databases
- 3).HDFS

what is File?

a file is a Named location on the disk,which is used to store the data permanently in a non-volatile memory like harddisk for future purpose.

the different file formates are .txt
.json
.csv
.pdf
.xlsx
.docx
.....
.....etc.,

we can perform the operation on the files,first we need to opening a file.

we can open a file by calling open() of builtin's module

at the time of opening a file to specify the mode of that file.

| mode | Description |
|------|--|
| ---- | ----- |
| r | --> opening a file in read mode (it's by default) |
| w | --> opening a file in write mode,if file already existed,the old data will be deleted and to write the new data into that file otherwise to create a new file and to write the data into that file. |
| a | --> opening a file in append mode,if file already |

existed the data will be added at the end of the previous data
in that file otherwise to create a new-file and to write the
data into that file.

x --> Opening a file in exclusive creation mode,if
file already existed the operation become fail otherwise to create
a new file and to write the data into that file.

t --> Opening a file in text mode
(it's by default)

b --> Opening a file in byte mode

+ --> to apply the multiple modes
(w+b)/wb

the open() to return the file-object.

the file_object provides some methods,to perform the operation on the files.

| Method | Description |
|--------------|--|
| read() | to read the entire data from the file (to return the string format) |
| read(n) | to read 'n'-charecters from the file |
| readline() | to read the line from the file,by default to read the first line from the file. |
| readlines() | to read the lines from the file. to return the list format. |
| write(s) | to write the data into the file |
| writelines() | to write the lines into the file |
| seek() | to change the file-pointer position |
| tell() | to return the current file pointer position |
| close() | to close the file |

after performeing the operation on the files next we need to close the file.

we can close the file by calling close() of file object.

order of the file-handling

step1:

```
-----
    to opening a file

    file_obj=open(filename,mode)
ex1:
----
    f1=open('sample.txt') #read and text mode

    f2=open('sample.txt','w') #write mode

    f3=open('E:\\siva\\sample.txt','a') #append mode

    f4=open('E:/siva/sample.txt','x') #exclusive mode

step2:
-----
    to perform the operation on the files

    file_obj.methodname()
ex1:
----
    f1.read()

    f2.write(string)

step3:
----
    to close the file

    file_obj.close()

ex1:
----
    f1.close()

    f2.close()
```