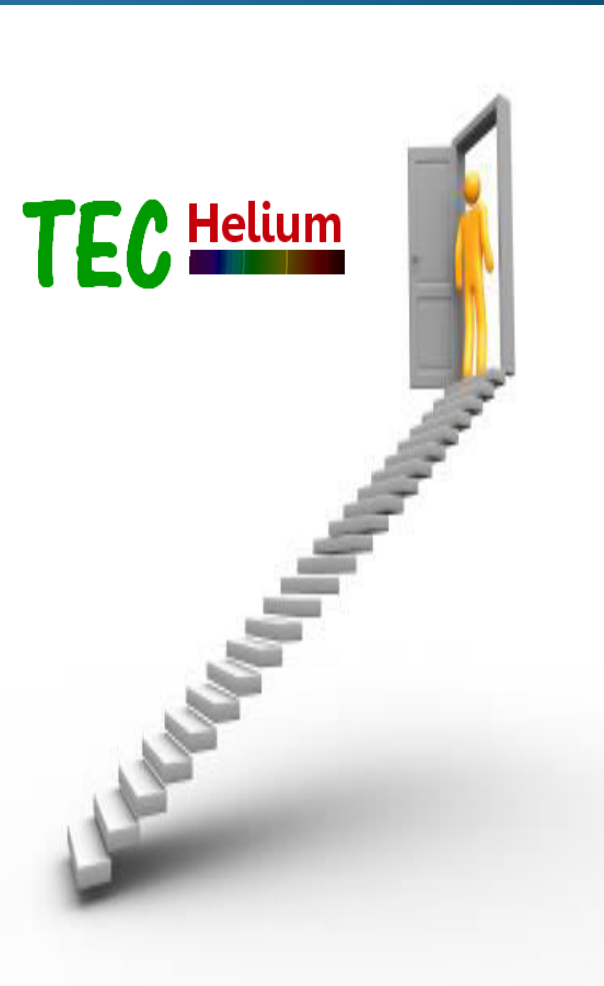


Lets Face it, now

Project Oriented Python

File operation
&
Programs



- Using `input()` or `raw_input()` and `print()` we can read and write on a standard input and output.
- Most program need to read config files/data files or write log files.
- Python provides basic functions and methods necessary to manipulate files.

File Objects

- Before we read or write a file , we have to open it.
- `file = open(filename[, mode[, buffersize]])`
- Buffersize: 0 –no buffering, 1 – line buffering,
-1 – default, N – any integer value greater than 1.

Mode	Description
'r'	Read only . Pointer at start of file.
'r+'	Read and write. Pointer at start of file
'w'	Write only. New file,Overwrite if exist
'w+'	Write and read. New file, Overwrite if exist
'a'	Append. Pointer at end. Create new if do not exist.

File Objects

- methods:
 - `read([nbytes])`, `readline()`, `readlines()`
 - `write(string)`, `writelines(list)`
 - `seek(pos[, how])`, `tell()`
 - `flush()`, `close()`
 - `fileno()`

Example

```
>>> fl=open('PBooo6-biggest and smallest.py','r+')
```

```
>>> fl.read()
```

```
\n\n# program to take three number and find the biggest number\n\nnum1 =  
input("Enter first number : ")\nnum2 = input("Enter Second number :  
")\nnum3 = input("Enter Third number : ")\nnum = [num1, num2,  
num3]\nbiggest = num[0]\nsmallest = num[0]\nfor x in range (1,3):\n    if  
biggest < num[x] :\n        biggest = num[x]\n    elif smallest > num[x]:\n        smallest = num[x]\n\nprint "Biggest number is ", biggest\nprint "Smallest  
number is ", smallest\n'
```

```
>>> fl.read()
```

```
"
```

```
>>> fl.seek(0,0)
```

```
>>> for lines in fl.readlines(): print lines
```

Line-by-line processing

- Reading a file line-by-line:

```
for line in open("filename").readlines():  
    statements
```

Example:

```
count = 0  
for line in open("bankaccount.txt").readlines():  
    count = count + 1  
print "The file contains", count, "lines."
```

Files: Input

<code>inflobj = open('data', 'r')</code>	Open the file 'data' for input
<code>S = inflobj.read()</code>	Read whole file into one String
<code>S = inflobj.read(N)</code>	Reads N bytes ($N \geq 1$)
<code>L = inflobj.readlines()</code>	Returns a list of line strings

Files: Output

<code>outflobj = open('data', 'w')</code>	Open the file 'data' for writing
<code>outflobj.write(S)</code>	Writes the string S to file
<code>outflobj.writelines(L)</code>	Writes each of the strings in list L to file
<code>outflobj.close()</code>	Closes the file

Python OS module

- `import os`
- `os.rename(current_file_name, new_file_name)`
- `os.remove(file_name)`
- `os.getcwd()`

```
>>> for i in os.listdir(os.getcwd()) : print i
```

Exercise

- Write a program to
 - Take name as input,
 - Search a student-database.data file with following data in each line
<NAME> - <AGE> - <Grade>
 - If the name is found, it print the AGE and Grade ,
 - Else take age and grade as input and append the database.data file

Exercise

- Write a program
 - To print a menu
 1. Add records
 2. Search records
 3. Print records
 4. Exit

Please Input your Choice (1-4):
 - Write a function for each choices and call the when input appropriately.
 - Record could be anything like example
 <Device> <IP Address> <user name> <password> <prompt>
 - Search should be on first field.
 - Print records should well formatted example like
 #####Printing Records #####

S.No.	Device	IP Address	User Name	Prompt
1.	Router	192.168.1.1	admin	cisco>
2.	http-server	192.168.1.2	superuser	N/A

Thanks