

File Methods

Operations on files

<code>file.read([size])</code>
<code>file.readline([size])</code>
<code>file.readlines([sizehint])</code>
<code>file.write(str)</code>
<code>file.writelines(sequence)</code>
<code>file.flush()</code>
<code>file.close()</code>
<code>readable ()</code>
<code>writable ()</code>
<code>file.tell()</code>
<code>file.seek(offset[, whence])</code>

```
file=open('---')
```

While opening a file we are getting object of which class ?

Program:

Input

```
file = open('MyData.txt')  
print(type(file))
```

Output:

<type 'file'>

If we don't give anything it is in readymade by default.
If given 'r' will get the same thing

Program

Input:

Output:

```

file = open('MyData.txt')
print(type(file))
print(dir(file)) # to get members of the class

```

```

['_class__', '__delattr__', '__doc__', '__enter__', '__exit__', '__format__', '__getattr__',
 '__hash__', '__init__', '__iter__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__setattr__',
 '__sizeof__', '__str__', '__subclasshook__', 'close', 'closed', 'encoding', 'errors', 'fileno', 'flush',
 'isatty', 'mode', 'name', 'newlines', 'next', 'read', 'readinto', 'readline', 'readlines', 'seek',
 'softspace', 'tell', 'truncate', 'write', 'writelines', 'xreadlines']

```

Getting members of the class

Properties of a file

Using function read

Program:

Input:

```

file = open('MyData.txt', 'r') #asking file to read
file.close() #closing a file
print(file.name) #asking to print the file name
print(file.mode) #asking to print the mode of file
print(file.close)

```

Output:

MyData.txt

r

<built-in method close of file object at 0x10752ddb0>

Using function read line

Program:

Input:

```

file = open('MyData.txt', 'r')#asking file to read
line= file.readline()#reading a line
print(line)
line=file.readline()# using second time
print(line)

```

Output:

apples are red

grapes are green

To avoid the spaces **write print(line, end='')**

The difference between read and read lines is read will read the entire content of string whereas headline reads whole line

And headline gives just a string and gives sets of strings.

To print for a loop

For line in lines:

print(line, end='')

If it works we will get data line by line

File write(str): write for a string

File.write line (sequences): Write the line for the content of any sequence

file=open('prop.txt', 'w')

str1= 'python is simple\n it is easy\n everything is object'

File.write(str1)

Write lines(sequence)

Program:

```

file = open('Prop.txt', 'w')

list1 = ['python is simple\n', 'it is easy\n', 'everything is an object\n']

file.writelines(list1)

```

File.flush: it is used for flushing the content from buffer onto the file

File.close: close is a method for closing a file

readable(): used to know whether a file is readable or not

`writable()`:used to know whether a file is writable or not.