**File handling in python**

File handling is an important part of any web application.

Python has several functions for creating, reading, updating, and deleting files.

The key function for working with files in Python is the **open()** function.

The open() function takes two parameters; ***filename***, and ***mode***.

There are four different methods (modes) for opening a file:

**“r” - Read** – (Default value.) Opens a file for reading, error if the file does not exist

**“a” - Append** - Opens a file for appending, creates the file if it does not exist

**“w” - Write** - Opens a file for writing, creates the file if it does not exist

**“x” - Create** - Creates the specified file, returns an error if the file exists

**Syntax:**

File = open(“sample.txt”) or open(“sample.txt”,”r”)

* f = open(“demofile.txt”, “r”)  
  print(f.read())
* Return the 5 first characters of the file:

f = open(“demofile.txt”, “r”)  
print(f.read(**5**))

**Read lines:**

Using the readline method we can read the one line from the document

**Syntax:**

File = open(“sample.txt”,”r”)

Content = file.readline()

Print(content)

* By calling readline two times we can read the first two lines of the data

**Close Files**

Close the file when you are finished with it:

f = open(“demofile.txt”, “r”)  
print(f.readline())  
f.close()

**Write to an Existing File**

“a” - Append - will append to the end of the file

“w” - Write - will overwrite any existing content

The append mode will add more data to the existing file and it doesn’t delete the previous data

Whereas, the write mode will overwrite the existing data and replace with the new one

#appending the content

F = open(“sample.txt”,”a”)

f.write(“more content”)

f.close()

# reading the file:

F = open(“sample.txt”,”r”)

Print(f.read())

#overwriting the content:

F = open(“sample.txt”,”w”)

f.write(“overwriting”)

f.close()

# reading the file:

F = open(“sample.txt”,”r”)

Print(f.read())

**Create a New File:**

To create a new file in Python, use the open() method, with one of the following parameters:

“x” - Create - will create a file, returns an error if the file exists

“a” - Append - will create a file if the specified file does not exists

“w” - Write - will create a file if the specified file does not exists

**Check if File exist:**

Check if file exists, then delete it:

import os  
if os.path.exists(“demofile.txt”):  
  os.remove(“demofile.txt”)  
else:  
   print(“The file does not exist”)

**Delete Folder**

To delete an entire folder, use the os.rmdir() method:

Remove the folder “myfolder”:

import os  
os.rmdir(“myfolder”)

**Advantages:**

* Versatility
* Flexibility
* User friendly
* Cross platform

**Disadvantages:**

* Error prone
* Security risk
* Complexity
* performance