

Python File Open

June 18, 2024

Dr. Labeed Al-Saad

File handling is an important part of any web application.

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

File Handling 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

“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

In addition you can specify if the file should be handled as binary or text mode

“t” - Text - Default value. Text mode

“b” - Binary - Binary mode (e.g. images) for a file:

0.1 Syntax

To open a file for reading it is enough to specify the name of the file:

```
[ ]: f = open("demofile.txt")
```

The code above is the same as:

```
[ ]: f = open("demofile.txt", "rt")
```

Because “r” for read, and “t” for text are the default values, you do not need to specify them. **

Note: Make sure the file exists, or else you will get an error.

1 Python File Open

1.1 Open a File on the Server

Assume we have the following file, located in the same folder as Python:

demofile.

xt

Hello! Welcome to demofil e.txt This file is for testing pur poses. Goo

To open the file, use the built-in open() function.

The open() function returns a file object, which has a read() method for reading the content of the file:

Exa: mpled Luck!

```
[2]: f = open("demofile.txt", "r")
      print(f.read())
```

```
-----
FileNotFoundError                                Traceback (most recent call last)
Cell In[2], line 1
----> 1 f = open("demofile.txt", "r")
      2 print(f.read())

File ~\anaconda3\Lib\site-packages\IPython\core\interactiveshell.py:310, in _modified_open(file, *args, **kwargs)
    303 if file in {0, 1, 2}:
    304     raise ValueError(
    305         f"IPython won't let you open fd={file} by default "
    306         "as it is likely to crash IPython. If you know what you are
    307         doing, "
    308         "you can use builtins' open."
    309     )
--> 310 return io_open(file, *args, **kwargs)

FileNotFoundError: [Errno 2] No such file or directory: 'demofile.txt'
```

The error message above appeared because there is no file named 'demofile.txt' found in current directory.

If the file is located in a different location, you will have to specify the file path, like this:

Examp: le Open a file on a different locati

Note: you need to write a double backslash after the name of drive (C:\) that includes the specified file not like the way that we normally used (C:)on:

```
[6]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
      print(f.read())
```

This is just test file

Dr. Labeed Al-Saad

1.2 Read Only Parts of the File

By default, the `read()` method returns the whole text, but you can also specify how many characters you want to return:

Example:

Return the 5 first characters of the file:

```
[8]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
     print(f.read(5))
```

This

1.3 Read Lines

You can return one line by using the `readline()` method:

Example:

Read one line of the file:

```
[10]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
      print(f.readline())
```

This is just test file

By calling `readline()` two times, you can read the two first lines:

Example:

Read two lines of the file:

```
[6]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
     print(f.readline())
     print(f.readline())
```

This is just test file

Dr. Labeed Al-Saad

By looping through the lines of the file, you can read the whole file, line by line:

Examp: le Loop through the file line by line:

```
[12]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
      for x in f:
          print(x)
```

This is just test file

Dr. Labeed Al-Saad

1.4 Close Files

It is a good practice always to close the file when you are done with it.

Example:

Close the file when you are finished with it:

```
[14]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
      print(f.readline())
      f.close()
```

This is just test file

Note: You should always close your files, in some cases, due to buffering, changes made to a file may not show until you close the file.

2 Python File Write

2.1 Write to an Existing File

To write to an existing file, you must add a parameter to the `open()` function:

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

“w” - Write - will overwrite any existing content

Ex: Open the file “demofile2.txt” and append content to the file:

```
[36]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "a")
      f.write("\n" "Now the file has more content" "\n") # the first "\n" to left line
      ↪ before while the second "\n" to left line after
      f.close()

      #open and read the file after the appending:
      f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
      print(f.read())
```

This is just test file

Dr. Labeed Al-Saad

Now the file has more content

Example: Open the file “demofile3.txt” and overwrite the content:

```
[42]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "w")
      f.write("\n" "Now we overwrite the past content" "\n" "\n" "Dr.Labeed Al-Saad"
      ↪ "\n" "-----" "\n")
      f.close()
```

```
#open and read the file after the appending:
f = open("C:\\Users\\hp\\OneDrive\\Desktop\\demofile.txt", "r")
print(f.read())
```

Now we overwrite the past content

Dr.Labeed Al-Saad

Note: the “w” method will overwrite the entire file.

2.2 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 exist

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

Example:

Create a file called “myfile.txt”:

```
[44]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\myfile.txt", "x")
f.write("This is new created file" "\n"
↪ "-----" "\n" " Dr. Labeed Al-Saad")
f.close()

f = open("C:\\Users\\hp\\OneDrive\\Desktop\\myfile.txt", "r")
print(f.read())
```

This is new created file

Dr. Labeed Al-Saad

[]: Example:

Create a new file if it does not exist:

```
[64]: f = open("C:\\Users\\hp\\OneDrive\\Desktop\\myfile1.txt", "w") # here we created a
↪ new file which is does not exist
f.write("This is new file created by w" "\n"
↪ "-----" "\n") # writing something in it
f.close()

# open the created file to show what we wrote in it
```

```
f = open("C:\\Users\\hp\\OneDrive\\Desktop\\myfile1.txt", "r")
print(f.read())
f.close()
```

This is new file created by w

3 Python Delete File

3.1 Delete a File

To delete a file, you must import the OS module, and run its `os.remove()` function:

Exam: ver Remove the finyfile1.txt.txt”:

```
[66]: import os
      os.remove("C:\\Users\\hp\\OneDrive\\Desktop\\myfile1.txt")
```

3.2 Check if File exist:

To avoid getting an error, you might want to check if the file exists before you try to delete it:

Example: Check if the file exists, then delete it:

```
[68]: import os
      if os.path.exists("C:\\Users\\hp\\OneDrive\\Desktop\\myfile1.txt"):
          os.remove("demofile.txt")
      else:
          print("The file does not exist")
```

The file does not exist

3.3 Delete Folder

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

Example:

Remove the folder “myfolder”:

```
[ ]: import os
      os.rmdir("myfolder")
```

Note: You can only remove empty folders.