

Python File Open

Open a File on the Server

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

Demofile.txt

```
Hello! Welcome to demofile.txt
```

```
This file is for testing purposes.
```

```
Good Luck!
```

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:

Example

```
f = open("demofile.txt", "r")  
  
print(f.read())
```

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

Example

Open a file on a different location:

```
f = open("D:\\myfiles\\welcome.txt", "r")  
  
print(f.read())
```

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:

```
f = open("demofile.txt", "r")  
  
print(f.read(5))
```

Read Lines

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

Example

Read one line of the file:

```
f = open("demofile.txt", "r")  
  
print(f.readline())
```

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

Example

Read two lines of the file:

```
f = open("demofile.txt", "r")  
  
print(f.readline())  
  
print(f.readline())
```

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

Example

Loop through the file line by line:

```
f = open("demofile.txt", "r")

for x in f:

    print(x)
```

Close Files

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

Example

Close the file when you are finished with it:

```
f = open("demofile.txt", "r")

print(f.readline())

f.close()
```

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.