# Lecture-21

## Renaming and Deleting Files

Python **os** module provides methods that help you perform file-processing operations, such as renaming and deleting files.

To use this module you need to import it first and then you can call any related functions.

## The rename() Method

The *rename()* method takes two arguments, the current filename and the new filename.

Syntax

os.rename(current_file_name, new_file_name)

**Example:**

```
# Open a file
fd = open("D:\\Python directory\\roza.txt", "w+")
fd.write( "Python is a great language.\nYeah its great!!\n")
fd = open("D:\\Python directory\\roza.txt", "r+")
print(fd.read())
# Close opend file
fd.close()
import os
os.rename("D:\\Python directory\\roza.txt", "D:\\Python directory\\raja.txt")
```

- **The *remove()* Method**

You can use the *remove()* method to delete files by supplying the name of the file to be deleted as the argument.

Syntax

os.remove(file_name) Example

```
import os
# Delete file test2.txt
os.remove("text2.txt")
```

Example:

```
import os
os.remove("D:\\Python directory\\raja.txt")
```

## Directories in Python

All files are contained within various directories, and Python has no problem handling these too. The **os** module has several methods that help you create, remove, and change directories.

The *mkdir()* Method

You can use the *mkdir()* method of the **os** module to create directories in the current directory. You need to supply an argument to this method which contains the name of the directory to be created.

Syntax

```
os.mkdir("newdir")
```

## Example

Following is the example to create a directory *test* in the current directory –

```
 import os
 # Create a directory "test“
os.mkdir("D:\\Python directory\\ test")
```

# The *getcwd()* Method

The *getcwd()* method displays the current working directory.

Syntax

```
os.getcwd()
```

**Example**

Following is the example to give current directory –

```
import os
# This would give location of the current directory
test=os.getcwd()
print(test)
```

**OUTPUT:**

```
D:\Python directory
```

## The *rmdir()* Method

The *rmdir()* method deletes the directory, which is passed as an argument in the method.

Before removing a directory, all the contents in it should be removed.

Syntax

os.rmdir('dirname')

Example

Following is the example to remove "/tmp/test" directory. It is required to give fully qualified name of the directory, otherwise it would search for that directory in the current directory.

**# This would remove " D:\\Python directory\\test2" directory.**

**Import os**

os.rmdir("D:\\Python directory\\test2" )

**Note**: The rmdir() method can only remove empty directories.

In order to remove a non-empty directory, we can use the rmtree() method inside the shutil module.

```
import os
os.rmdir("D:\\Python directory\\Love2")
```

**OUTPUT:**

```
Traceback (most recent call last):
File "D:/Python directory/rr.py", line 2, in <module>
  os.rmdir("D:\\Python directory\\Love2")
WindowsError: [Error 145] The directory is not empty: 'D:\\Python directory\\Love2'
```

```
import shutils
hutil.rmtree("D:\\Python directory\\Love2"   )
```

# Testing File Types:

You can extract the file extension of a filename string using the os.path.splitext method. It splits the pathname path into a pair (root, ext) such that root + ext == path, and ext is empty or begins with a period and contains at most one period.

**Example:**

```
import os
files = os.listdir("D:\\Python directory\\Love")
for f in files:
    print(os.path.splitext(f))
```

**Output**

You will get the output –

```
('copy', '.py')
('direcotry tree', '.py')
('foo', '.txt')
```

## References:

1.Introduction to Computation and Programming using Python, by John Guttag, PHI Publisher

2.Python Programming using problem solving Approach by ReemaThareja, Oxford University, Higher Education Oxford University Press; First edition (10 June 2017), ISBN-10: 0199480173

3. Fundamentals of Python first Programmes by Kenneth A Lambert, Copyrighted material Course Technology Inc. 1 st edition (6th February 2009)

4. https://www.geeksforgeeks.org/python-programming-language

5. https://www.w3schools.com/python/

****END OF THE LECTURE***

***THANK YOU***