

Working With Files and Directories

Vivek K. S., Deepak G.

Information Systems Decision Sciences (ISDS)
MUMA College of Business
University of South Florida
Tampa, Florida

2017

Introduction to Files and Directories

- Our data is stored in different file formats within directories.
- Often, we need to create files in our programs to store our output data for later use.
- We can write program to modify data and store it back in a file.
- In this module, we will see how to work with files and folders in Python.
- Python's file operations are modeled after the *Unix System* and is available through the OS module.

Common File Operations

We can create a file by simply using the open method. If the file specified doesn't already exist, it will be created.

```
import os
fout = open('my_file.txt', 'wt')
fout.close()
# Check if the file has been created with
# exists() function
os.path.exists('my_file.txt')
```

The above function will return `True` if the file exists.

Common File Operations

We can check if the given path is a file or a directory by using the `isfile()` and `isdir()` methods.

```
name = 'my_file.txt'  
os.path.isfile(name)  
True
```

```
# to check if its a directory  
os.path.isdir(name)  
False
```

Common File Operations

Some of the other commonly used file operations are:

```
# Get the current working directory
```

```
os.getcwd()
```

```
# Get all the files in the current workind direcc
```

```
os.listdir()
```

```
# Rename a file
```

```
os.rename('old_file_name', 'new_file_name')
```

Summary

- We learned how to use Python's `os` module to perform different file and directory related operations.
- We learned how to use Python to interact with the operating system to accomplish system tasks.