#### **Python Libraries**

#### What is the Python Libraries?

We know that a module is a file with some Python code, and a package is a directory for sub packages and modules. But the line between a package and a Python library is quite blurred.

A Python library is a reusable chunk of code that you may want to include in your programs/ projects. Compared to languages like C++ or C, a Python libraries do not pertain to any specific context in Python. Here, a 'library' loosely describes a collection of core modules. Essentially, then, a library is a collection of modules. A package is a library that can be installed using a package manager like rubygems or npm.

#### **Python Standard Library**

The Python Standard Library is a collection of exact syntax, token, and semantics of Python. It comes bundled with core Python distribution. We mentioned this when we began with an introduction.

It is written in C, and handles functionality like I/O and other core modules. All this functionality together makes Python the language it is. More than 200 core modules sit at the heart of the standard library. This library ships with Python. But in addition to this library, you can also access a growing collection of several thousand components from the Python Package Index (PyPI).

#### **Important Python Libraries**

Next, we will see twenty Python libraries list that will take you places in your journey with Python. These are also the Python libraries for Data Science.

#### a. Matplotlib

Matplotlib helps with data analyzing, and is a numerical plotting library. We talked about it in Python for Data Science.

#### **b.** Pandas

Pandas is used for data-science. It provides fast, expressive, and flexible data structures to easily (and intuitively) work with structured (tabular, multidimensional, potentially heterogeneous) and time-series data.

#### c. Requests

Requests is a Python Library that lets you send HTTP/1.1 requests, add headers, form data, multipart files, and parameters with simple Python dictionaries. It also lets you access the response data in the same way.

# d. NumPy

It has advanced math functions and a rudimentary scientific computing package.

# e. SQLAlchemy

SQLAlchemy is a library with well-known enterprise-level patterns. It was designed for efficient and high-performing database-access.

# f. BeautifulSoup

It may be a bit slow, BeautifulSoup has an excellent XML- and HTML- parsing library for beginners.

## g. Pyglet

Pyglet is an excellent choice for an object-oriented programming interface in developing games. In fact, it also finds use in developing other visually-rich applications for Mac OS X, Windows, and Linux. In the 90s, when people were bored, they resorted to playing Minecraft on their computers. Pyglet is the engine behind Minecraft.

## h. SciPy

Next up is SciPy, one of the libraries we have been talking so much about. It has a number of user-friendly and efficient numerical routines. These include routines for optimization and numerical integration.

# i. Scrapy

If your motive is fast, high-level screen scraping and web crawling, go for Scrapy. You can use it for purposes from data mining to monitoring and automated testing.

# j. PyGame

PyGame provides an extremely easy interface to the Simple Directmedia Library (SDL) platform-independent graphic, audio, and input libraries.

#### k. Python Twisted

An event-driven networking engine, Twisted is written in Python, and licensed under the opensource MIT license.

#### l. Pillow

Pillow is a friendly fork of PIL (Python Imaging Library), but is more user-friendly. If you work with images, Pillow is your best friend.

#### m. pywin32

This provides useful methods and class for interaction with Windows, as the name suggests.

#### n. wxPython

It is a wrapper around wxWidgets for Python.

#### o. iPython

iPython Python Library has an architecture that facilitates parallel and distributed computing. With it, you can develop, execute, debug, and monitor parallel applications.

## p. Nose

Nose delivers an alternate test discovery and running process for unittest. This intends to mimic py.test's behavior as much as it can.

## q. Flask

A web framework, Flask is built with a small core and many extensions.

## r. SymPy

It is an open-source library for symbolic math. With very simple and comprehensible code that is easily extensible, SymPy is a full-fledged Computer Algebra System (CAS). It is written in Python, and hence does not need external libraries.

## s. Fabric

Along with being a library, Fabric is a command-line tool for streamlining the use of SSH for application deployment or systems administration tasks. With it, you can execute local or remote shell commands, upload/download files, and even prompt running user for input, or abort execution.

# t. PyGTK

PyGTK lets you easily create programs with a GUI (Graphical User Interface) with Python.



**1.** Using Module -It is a file which contains python functions/global variables/clases etc. It is just .py file which has python executable code / statement.

For example: Let's create a file usermodule.py

def hello\_message(user\_name):

return "Hello " + name

Now we can import usermodule.py module either in python interpreter or other py file.

import usermodule

```
print usermodule.hello_message( "India" )
```

## How to import modules in Python?

Python module can be accessed in any of following way.

 Python import statement import math print("2 to the power 3 is ", math.pow(2,3))

Just similar to math , user defined module can be accessed using import statement

2. **Import with renaming** import math as mt

print("2 to the power 3 is ", mt.pow(2,3))

3. **Python from...import statement** from math import pow

print("2 to the power 3 is ", pow(2,3))

4 | Page [Computer Science-XII]

#### 4. Import all names from math import \*

print("2 to the power 3 is ", pow(2,3))

# 2. Using Package - It is namespace that contains multiple package or modules. It is a directory which contains a special file \_\_ init \_\_.py

Let's create a directory geometry. Now this package contains multiple packages / modules to handle user related requests.



Now we can import it in following way: (consider root directory is CS2020)

## import CS2020.Students

or

#### from CS2020 import Students

#### **3**. Using Library

It is a collection of various packages. Conceptually, There is no difference between package and python library. In Python, a library is used loosely to describe a collection of the core modules.

'Standard Library' of Python language comes bundled with the core Python distribution are collection of exact syntax, token and semantics of the Python language . The python standard library lists down approx more than 200 such core modules that form the core of Python.

"Additional libraries" refer to those optional components that are commonly included in Python distributions.

The Python installers automatically adds the standard library and some additional libraries.

The additional library is generally provided as a collection of packages. To use such additional library we have to use packaging tools like easyinstall or pip to install such additional libraries.

#### 4. Using Framework

Framework is like a collection of various libraries which architects some more component.

For e.g. Django which has various in-built libraries like Auth, user, database connector etc.

#### Practical Example of Creating and using our own Library

```
Suppose we are creating a file MyLib.py with following functions inside of it:
def F1():
  print("This is Function-1")
def F2():
  print("This is Function-2")
def F3():
 print("This is Function-3")
def F4():
  print("This is Function-4")
After saving this file Press F5 and run it now type following on the python Shell
>>> import Mylib
>>> Mylib.F1()
This is Function-1
>>> Mylib.F2()
This is Function-2
>>> Mylib.F3()
This is Function-3
```