Libraries

Prof. K. Adisesha

Learning objectives

- **Understanding Library**

- **Importing Modules in Python Program**

- **Using Standard Library's functions**

- **Creating a Python Library**

What is a Library?

- A Library refers to a collection of modules that together cater to specific types of needs or applications.
- Some Commonly used Python library are:

Library	Functions
standard library	This library is distributed with Python that contains modules for various types of functionalities.
NumPy library	This library provides some advance math functionalities along with tools to create and manipulate numeric arrays.
SciPy library	This is another useful library that offers algorithmic and mathematical tools for scientific calculations.
tkinter library	This library provides traditional Python user interface toolkit and helps you to create user friendly GUI interface for different types of applications.
Malplotlib library	This library offers many functions and tools to produce quality output in variety of formats such as plots, charts, graphs

Python standard library

- Some commonly used modules of Python standard library are:

Library	Functions
math module	Provides mathematical functions to support different types of calculations.
cmath module	provides mathematical functions for complex numbers
random module	provides functions for generating pseudo-random numbers
statistics module	provides mathematical statistics functions
Urllib module	provides URL handling functions so that you can access websites from within your program

What is a Module?

- Python module is a file(.py file) containing variables, class definitions, statements and functions related to a particular task.
- A Module, in general:
  - Is independent grouping of code and data
  - Can be re-used in other programs.
  - Can depend on other modules
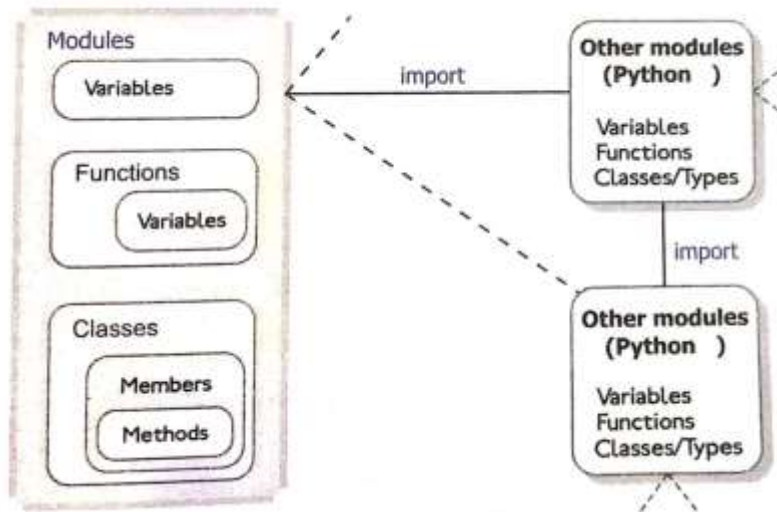
Structure of a Python Module

- A Python module can contain much more than just functions.
- A Python module is a normal Python file (.py file) containing one or more of the following objects related to a particular task:

Structure	Task
docstrings	triple quoted comments; useful for documentation purposes.
variables and constants	labels for data
classes	templates/blueprint to create objects of a certain kind
objects	instances of classes. In general, objects are representation of some real or abstract entity
statements	Instructions for processing
functions	named group of instructions

- So, we can say that the module 'XYZ' means it is file 'XYZ.py'

Importing Modules

- Python module files have an extension .py
- These modules can be imported in the following ways:
    - **import statement**
        - to import entire module
    - **from statement**
        - to import selected object
    - **from * statement**
        - import all names from the module

IMPORTING MODULES- import

- import statement is used to include the modules in other programs.
    - syntax : import <filename>
    - example: import math
- more than one module can be inserted in a python program
    - syntax : import <filename> ,<filename>,<filename>……
    - example: import math,os
- using import statement one can view all the functions and other attributes of a particular module
    - example:
        import math
        dir(math)

IMPORTING MODULES- from

- importing module can be done using from statement specific attributes can be included in other programs.

- syntax :

- from <filename> import function name

- example:

- from math import math.sqrt

IMPORTING MODULES- from*

- from* statement can be used to import all names from the module in to the current calling name space.

- syntax :

  from <filename> import *

- example:

  from math import *

  math.sqrt(4)

- we can access any function by using dot notation.

NAMESPACES

When we import modules in a particular program these modules will become part of that program and are called as namespace.

Python implements namespaces in the form of dictionaries. It maintains a name to object mapping.

There are three types of namespaces

1) Global

2) Local

3) Built in

Namespaces

MODULE ALIASING

- One can create an alias while importing module in a program
- Syntax:

    import <filename> as <alias name>

- Example:

    import math as m
    m.sqrt(4)

- Like module aliasing members are also aliased
- syntax:

    import <filename> as <alias name>,member as alias name

- Example:

    import test as t, add as sum

  test.py is module file and is referred to as t and add is the function, it is referred to as sum.

PACKAGE/LIBRARY

- Python packages are the collection of related modules. You can import a package or create your own.
- Python package is a simply directory of python modules
- Steps to create and import a package
    1. create a directory named 'Gemetry'
    2. add modules area.py and volume.py
    3. create a file __init__.py in directory 'Geometry'. The __init__.py files are required to make python treat the directory as containing package
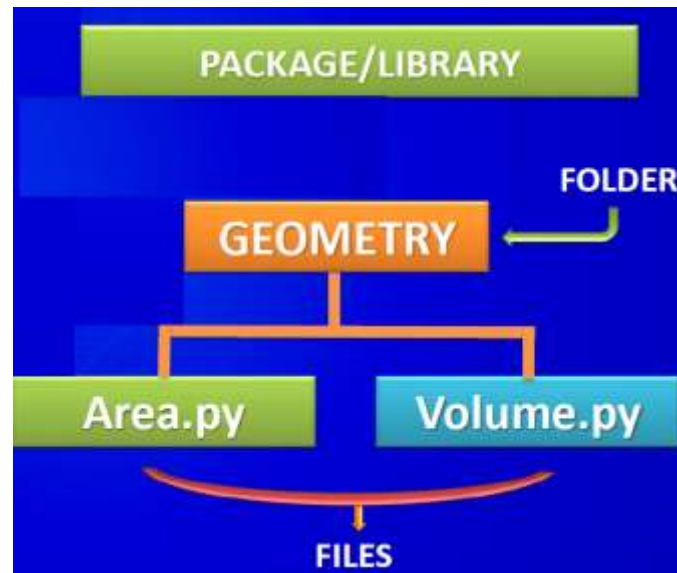
PACKAGE/LIBRARY

- What is __init__.py file?
- __init__.py is simply a file used to consider directories on the disk as package of python.
- It is basically used to initilize the python package
- Python searches module in the following manner
- 1) Searches in current directory
- 2) If the module is not found then searches each directory in the shell variable PYTHONPATH
- 3) If all else fails, python checks the default path which is the installation location of the python

pip

- What is pip?

- pip is a package-management system used to install and manage software packages written in Python.

- To check pip version run, pip --version at dos prompt

Conclusion!

- We learned about the Python Library .
- Types of Libraries
- Modules in Python
- Packages in Python.
- We also looked at the creating PACKAGE/LIBRARY.

Thank you