**Modules:**

Module is nothing but a file that contains python code in that in form of functions classes and variables which we can use in other programs.

Modules are used to reuse code.

Any file with .py extension is called as a module.

So here assume we are developing a big project ABCD so we divide this into module A,B,C,D so we will be using this modules in another project if it is needed so code reusability is a very important thing in programming.

In modules we have 2 types of modules:

* User defined modules
* Built-in modules

**Built-in modules :**

The modules which are already predefined in python

So lets have a look on some built-in modules:

1.Math

2. Random

3. Datetime

These are few predefined modules in python let’s see with one simple examples

Import math

Print(math.pi) #output: 3.141592653589793

\*\*\*\*

import random

print(random.randint(1, 10)) # Output: Random integer between 1 and 10

\*\*\*\*

import datetime

now = datetime.datetime.now()

print(now) # Output: Current date and time

**Userdefined modules :**

The modules which are created by programmers is called userdefined modules

Here I have defined some functions in the module file

def add(a,b):  
 return a+b  
def sub(a,b):  
 return a-b  
def mul(a,b):  
 return a\*b  
def div(a,b):  
 return a/b

So I need to do some operations on add.sub,mul and div to do these operations no need to do the coding from the scratch so just I will import those modules.

from module import \*

now just I will do the operations which ever I needed.

a = 9  
b = 7  
  
c = add(a,b)  
print(c) # 16  
  
c = sub(a,b)  
print(c) # 2

c= mul(a,b)  
print(c) # 63

c = div(a,b)  
print(c) # 1.2857142857142858

**Package**

Package is the collection of modules and subpackages

Each package contains a special file called as \_\_init\_\_.py

When a directory contains a \_\_init\_\_.py file python treats the directory as a package allowing u to import modules and sub packages.

In python we have larger developer community so code which is written by the developer he will add that package to the Pypl

So knowing about the pypl(python package index) we call this pypl as a repository more than 2 lakh packages are present in this repository so we will install those packages from this repository with the help of pip we can say this as package manager we can install, uninstall, search , upgrade, list operations in with this pip.