DAY 7

(MODULES)

Modules in Python are files containing Python code (functions, classes, variables, etc.) that can be imported and used in other Python programs. They help in organizing code into manageable sections and promote code reusability.

### **Creating a Module**

To create a module, simply save a Python script with a .py extension.

### **Using a Module**

You can import a module using the import statement:-

import my\_module

Importing Specific Items from a Module:-

from my\_module import greet, pi

Using as to Alias a Module:-

import my\_module as mm

Let’s create two python files.

1. Manager.py
2. Employee.py

Manager.py

print("manager file executed")

manager\_name = "Abhishek"

manager\_address = "Bharatpur"

employee\_list = ['rohit','rahul','mohit','suresh','jagdesh']

rohit\_id = '[hello@gmail.com](mailto:hello@gmail.com)'

rohit\_pass = "1234@5678"

def rohit\_hii() :

print("Employee name : ","Rohit")

print("Rohit id : ",rohit\_id)

print("Rohit password : ",rohit\_pass)

if \_\_name\_\_ == "\_\_main\_\_" : # restriction . Now code under if condition can't excess in other module. It can only excess through this file .

def manager\_hii() :

print("Employee name : ","manager")

print("manager id : ",manager\_id)

print("manager password : ",manager\_pass)

manager\_id = "[manager@gamil.com](mailto:manager@gamil.com)"

manager\_pass = "1234@8765"

Now,using manager mudule in employee

# # Rohit employee

# # excess module manager

# import manager

# # excess id and password from module manager.py

# print(manager.rohit\_id)

# print(manager.rohit\_pass)

# print(manager.employee\_list)

# print(len(manager.employee\_list))

# # calling function from module manager

# manager.rohit\_hii()

# # excess module manager by mng name by using alias function

# import manager as mng

# print(mng.rohit\_id)

# print(mng.rohit\_pass)

# print(mng.employee\_list)

# print(len(mng.employee\_list))

# # here employee "rohit" can excess all detail even manager information

# # import only neccessary info from module (excessing only certain contain)

# # from manager import rohit\_id,rohit\_pass,rohit\_hii

# from manager import rohit\_id

# print(rohit\_id)

# from manager import rohit\_pass

# print(rohit\_pass)

# print()

# from manager import rohit\_hii

# print(rohit\_hii)

# from manager import manager\_hii

# print(manager\_hii)

# That's why there should be security mechanism such that employee can't excess unneccessary info or secret info

# for that we restrict a condition within that module

# so if we make a restrict condition in manager module than we can't excess code that is written in that condition

from manager import manager\_hii #now we can't import it because of retrictions

print(manager\_hii)

### **Built-in Modules**

Python comes with a large standard library of modules. Some commonly used built-in modules include math, sys, os, datetime, and random.

import math

print(math.sqrt(16)) # Output: 4.0

print(math.pi) # Output: 3.141592653589793

### **Installing External Modules**

You can install external modules using the package manager pip.

Example: Installing and Using the requests Module

1. Install the requests module:-

pip install requests

1. Use the requests module in your code:

import requests

response = requests.get('https://api.github.com')

print(response.status\_code) # Output: 200 (if successful)

print(response.json()) # Output: JSON response from the API