Task 3:-

Date:- 20/3/25

Importing & creating python modules & packages

Aim:- To implement and demonstrate the process of importing built-in modules, creating user-defined module and organizing code into packages in python, thereby promoting code reusability, modularity & maintainability

3.1

1) Perform common math and random operations.
2) Work with the operating system (create/change directories, list contents) and the python version
3) Compute basic statistics mean, median, mode, standard deviation).

Algorithm:

1) Import required modules: math float in (0,0,1,0), a
2. random float integer in [2,6] inclusive π, ceil (2,3) floor (2.3), factorial (5), gcd (5,15), abs (-10), pow(3,5) log base 3 of 2, log 10(a) for a=100, and check NaN/Infinity.

3. Os & Sys:

- Create C:\Pythonlab if not present & print the current working directory.
- Create C:\Pythonslot S24 if not present and change the current working directory to it
- List all files/ directories in new current directory

4. Statistics

- On lists: [5,6,8,10] and [2,5,2,2,8,3,9,4,25,6], compute mean, median, mode, stdev.

5.) Print neatly formatted results.

Expected Sample Output

---MATH & RANDOM---

Sqrt (5) = 2.23606797755982988

radians (30) = 0.523598775982988

random () in [0,1) = 0.37444887175646646

randint (2,6) = 6

pi = 3.141592653589793

Ceil (2.3) = 3

Floor (2.3) = 2

Factorial (5) = 120

gcd (5,15) = 5

abs (-10) = 10

pow (3,5) = 243

log base 3 of 2 = 0.6309297535714574

log 10 (100) = 2.0

isinf (∞) = True, isnan (NaN) = True

---OS & SYS---

Created / ensured : C:\Pythonlab

Current working directory : C:\... (your current path)

Created / ensured & changed into : C:\Pythonslot52L4

Directory contents of C:\Pythonslot52L4 : []

Python version : 3.x.x (... details ...)

Program

```
import math
import random
import os
import sys
import statistics as stats
from pathlib import Path

print("\n--- Math & RANDOM ---")
print("sqrt(5) =", math.sqrt(5))
print("radians(30) =", math.radians(30))
print("random() in [0,1) =", random.random())
print("randint(2,6) =", random.randint(2,6)) # inclusive
print("pi =", math.pi)
print("ceil(2.3) =", math.ceil(2.3))
print("floor(2.3) =", math.floor(2.3))
print("factorial(5) =", math.factorial(5))
print("gcd(5,15) =", math.gcd(5,15))
print "abs(-10) =", abs(-10))
print("pow(3,5) =", pow(3,5))
print("log base 3 of 2 =", math.log(2,3))
a_val = 100
print f"log 10({a_val}) = ", math.log(2,3))
inf_val = float('inf')
nan_val = float('nan')
print(f"is inf (∞) = {math.isinf(inf_val)}; is nan (NaN) =
                    {math.isnan(nan_val)}")
```

--- STATISTICS ---

mean([5,6,8,10]) = 7.25

median([5,6,8,10]) = 7.0

mode([2,5,3,2,8,3,9,4,2,5,6]) = 2

stdev([2,5,3,2,8,3,9,4,2,5,6]) = 2.2715633383201675

```
print ("\n--- os & sys ---")
path_pythonlab = Path (r"C:\Pythonlab")
path_pythonlab.mkdir (parents = True, exist_ok = True)
print(f" Created / ensured : {path_pythonlab}")
print (" Current working directory:", os.getcwd())
target_dir = Path (r"C:\Pythonslot$2L4")
target_dir.mkdir ( parents = True, exist_ok = True)
os.chdir (target_dir)
print(f "changed into : {target_dir}")
print (" Directory contents:", os.listdir())
print (" Python version:", sys.version)

print ("\n--- STATISTICS---")
data1 = [5,6,8,10]
data2 = [2,5,3,2,8,3,9,4,2,5,6]
print(f "mean({data}) =", stats.mean (data1))
print(f "median({data1}) =", stats.median (data1))
print(f "mode ({data2}) =", stats.mode(data2))
print (f "stdev ({data2}) =", stats.stdev (data2))
```

Result:- To implement and demonstrate the process of importing built in modules is verified successfully

Task : 13.2

Create a Python package named Cardpack containing a module CardFun that imports the random module. Assign a range of cards, call a function from the module

Aim :- To create a python package named cardpack containing a module cardfun that imports the random module.

Algorithm :-

Step 1 :- Start

Step 2 :- To create a package Cardpack

Step 3 :- To create CardFun and import random function

Step 4 :- Assign a Cards range.

Step 5 :- Call a module function.

Step 6 :- Display the random sample cards

Step 7 :- Stop

Program :

Card Fun

```
import random
def func():
Cards=[]
for i in range (1,53):
 Cards. append(i)
Shuffle_cards = random. sample (cards, k=52)
print ("\n\n", shuffled_cards, "\n\n")
```

Mymod. Py

```
import Card Fun
CardFun. func()
```

Output

```
RESTART:
C:\Users\student.MAT2VC6833\AppData\Local\Programs\
Python\Python31\Lib\sitepackages\Cardpack\MyModfy
[5,24,13,22,20,41,38,51,4,7,34,49,14,50,37,40,
15,35,17,18,33,39,36,42,12,6,16,19,48,29,2,27,
11,31,46,28,21,32,8,25,30,23,26,10,43,47,3,
44,52,1,45,9]
```

Result:- To create a python package named cordpack containing a module cordfun that imports & works successfully.

Task 3.3

You are tasked with developing a modular calculator application in Python. The calculator should support basic arithmetic operation: addition, subtraction, multiplication, and division. Each operation should be implemented in a seperate module.

Aim:- To create a main program to handle user input call appropriate module, and diplay results

Algorithm:-

1. Define function for addition, subtraction, multiplication, and division
2. Handle division by zero by raising an error if the divisor is 0.
3. Imports the module (mymath) containing these function
4 Intidize two numbers (a=10, b=5).
5. Call each function using mymath. <function_name>(a,b).
6. Rint the results of all operations.

Program: (my) math

```
def add (a,b):
return a+b
def subtract (a,b):
return a-b
def multiply (a,b):
 return a*b
def divide (a,b):
if b==0:
raise Value Error ("Cannot divide by zero")
return a/b
```

```
import mymath
a=10
b=5
```

Output

Addition : 15

Subtraction : 5

Multiplication : 50

Division : 2.0

```
print (" Addition : ", mymath . Add (a,b))
print (" Subtraction : " mymath . subtract (a,b))
print (" Multiplication : ", mymath . multiply (a,b))
print (" Division : ", mymath . divide (a,b))
```

Result :- Thus to create a main program to handle user input call appropriate module & verified successfully.

Task 34

You are working on a Python project that requires you to perform various mathematical operations and geometric area calculations. To organize your code better, you decide to two modules: mathfunction and areafunction
Demonstrate the use of function by performing a few

Aim:- To organize your code better, ya decide to two modules i math function and area function
Demonstrate the use of functio.

Algorithm:-

1. Create mathfunctions.py module:
2. Create areafunctions.py module:
3. Create main.py:
4. Print the output as expected.

Program:

1. Crede the mathfunctions.py module

```
defadd (a,b):
return a+b
defsubtract(a,b):
return a-b
def multiply (a,b):
return a*b
def divide(a,b):
if b ==0 :
return "Error! Division by zero"
return a/b
```

Output

```
Addition : 15
Subtraction : 5
Multiplication : 50
Division : 2.0
Circle area (radius = 7) : 153.93804002589985
Rectangle Area (5x10) : 50
Triangle Area (base = 6, height = 8) : 24.0
```

2. Create the areafunction.py module

```
import math
def Circle_area(radius):
return math.pi * radius * radius
def rectangle_area(length,width):
return length * width
def triangle_area(base,height):
return 0.5*base*height
```

VEL TECH [illegible]	
EX NO.	3
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	3
VIVA VOCE (3)	3
RECORD (4)	4
TOTAL (15)	15
SIGN WITH DATE	

3. Create the main.py file

```
import mathfunctions
import areafunctions
# Using math functions
print("Addition:", mathfunction.add(10,5))
print("Substration:", mathfunctions.subtrat(10,5))
print("Multiplication", mathfunctions.multiply(10,5))
print("Division!", math function.divide(10,5))
# Using area functions
print("Circle Area(radius=7):", areafunction.circle_area(
print("Rectangle Area 5x10):", areafunctions.rectangle_area(5,10))
print("Triangle Area(base=6, height=8):"areafunction.triangle_area(6,8)
```

Result:-

Thus, the program for importing Python modules and packages was successfully executed and the output was verified.