Date: 30/8/25 Importing and Creating Python modules and Packages in Python Program.

Task-3:

Aim:

To implement and demonstrate the process of importing built-in modules, creating user-defined modules, and organizing code into Packages in python, thereby promoting code reusability, modularity, and maintainability.

3.1

1) Perform Common math and random operations.
2) Work with the operating system and read the Python version.
3) Compute basic statistics.

Algorithm:

1) Import required modules: math, random, os, sys, statistics, Pathlib.
2) Math & random:
- Compute sqrt(5), radians(30), a random float in [0.0, 1.0), a random 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, log10(a) for a=100, and check NaN/Infinity.
3) OS & Sys:
- Create C:\pythonlab if not present and print the current working directory.
- Create C:\Pythonlab\s2491 if not present and change the current working directory to it
- List all files / directories in the new current directory.
- Print Python interpreter version.
4) Statistics:
- On lists: [5,6,8,10] and [8,5,3,2,8,3,9,4,2,5,6], Compute mean, median, mode, stdev.
5) Print neatly formatted results.

output:

--- Math & Random ---

sqrt (5) = 2.23606797749979

factorial (30) = 0.5235987755982988

random() in [0,1) = 0.37444887175564646 <-- (will vary)

randint (2,6) = 6

Pi = 3.14

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 = 0.6309297 5

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:\Python310+32\Lab1\[]

Directory contents of C:\Python310+32\Lab1: []

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

--- 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]) = 2.2715633

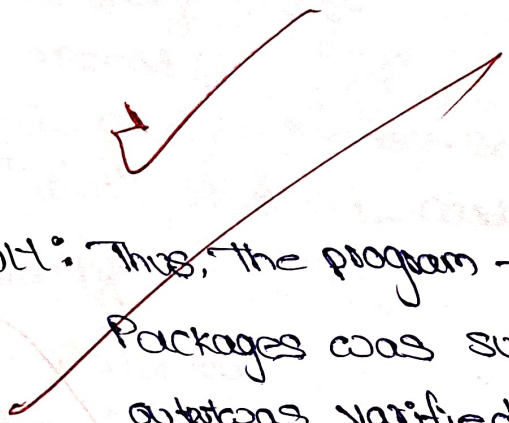
Program:

```
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("radiant(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'log10({a_val}) =", math.10(a_val))
in_f_val = float('inf)
nan_val = float('nan')
Print(f'isinf(oo)= {math.isinf(inf_val)}, isnan(NaN) =
                                {math.isnan(nan_val)}")

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 S2L4")
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]
data 2 = [2.5,3,2,8,3,9,4,2,5,6]
Print(f" mean ({data1}) =", Stats.mean(data1))
Print(f" median ({data2}) =", Stats.median(data1))
Print(f" mode ({data2}) = ", Stats.mode(data2))
Print(f"Stdev({data2}) = ", Stats.stdev(data2))
```



RESULT: Thus, the program for importing python modules and packages was successfully executed and the output was verified.

Aim: To implement and demonstrate the process of importing built-in modules, creating user-defined modules, and organizing code into packages in python.

3.2

Create a python package named cardpack containing a module Card-fun that imports the random module. Assign a range of cards, call a function from the module, and display a random sample of cards.

Algorithm:

1) Start
2) To create a package cardpack
3) To create a module card-fun and import random function
4) Assign a card range.
5) Call a module function
6) Display the random sample cards
7) Stop.

Program:

Card fun

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

My mod.py

```
import Card fun
Cardfun.func()
```

Output:

```
RESTART:
C:\Users\student\MAT2X6833\AppData\Local\Programs\Python\Python311\Lib\site-packages\cardpack\mymod.py
[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]
```

Result:

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

Aim: To implement and demonstrate the process of importing built-in modules, creating user-defined modules and organizing code into packages in Python

3.3

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

Algorithm:

1. Define functions for addition, subtraction, multiplication and division.
2. Handle division by zero by raising an error if the divisor is zero.
3. Import the module containing these functions.
4. Initialize two numbers (a=10, b=5).
5. Call each function using Mymath.<function_name>(a, b).
6. Print 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
Print("Addition:", mymath.add(a,b))
```

output

Addition : 15

Subtraction : 5

Multiplication : 50

Division : 2.0

```
Print("Subtractions:", mymath.subtract(a,b))
Print("Multiplication:", mymath.multiply(a,b))
Print("Division:", mymath.divide(a,b))
```

Result: Thus, The program for importing python modules and Packages was successfully executed and the output was verified.

Aim: To implement and demonstrate the process of importing built-in modules, creating user-defined modules and organizing code into package in Python.

3.4

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 create a package name mypackage which includes sub packages Pack1 and Pack2 with two modules: math functions and area functions Demonstrate the use of function by performing a few calculations and printing the results.

Algorithm:

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

Program

1. Create the mathfunctions.py module.

```
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:
return "Error! Division by zero"
return a/b
```

2. Create the area functions.py module

```
import math
def circle_area(radius):
```

Output:

Addition: 15.0

Subtraction: 5

Multiplication: 50

Division: 2.0

Circle area: 153.93

Rectangle area: 50

Triangle area: 24.0

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

3. Create the main.py file

```
import math_functions
import areafunctions

# using area functions
Print("Circle Area(radius=7):", area_function.circle_area(7))
Print("Rectangle Area(5x10):", area_function.rectangle_area(5,10))
Print("Triangle Area(base=6,height=7):", areafunctions.triangle_area(6,8)).
```

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

Result: Thus, the program for importing python modules and packages was successfully executed and the output was verified