20/8/25

Task3: Importing and creating python modules and package in python program

Aim: To implement and demonstrate the process of importing built, thereby promoting code reusability modularity, and maintainability

3.1

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

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), ceil(2.3), floor(2.3) factorial(5), gcd(5, 15), fabs(-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 and print the current working directory
- Create C:\pythonslot\2L4 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 list [5, 6, 8, 10] and [2.5, 3, 2.8, 3.4, 4, 2.5, 6] compute mean, median, mode stdev,

5) print neatly formatted results.

```
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"log10({a_val}) =", math.log10(a_val))
inf_val = float('inf')
nan_val = float('nan')
print(f"isinf(∞) = {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("created/ensured:", path_pythonlab)
print("Current working directory:", os.getcwd())
target_dir = Path(r"E:\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]
data2 = [2,5,3,2,8,3,9,4,2,5,6]

print(f"mean({data1}) =", stats.mean(data1))
print(f"median({data1}) =", stats.median(data1))
print(f"mode({data2}) =", stats.mode(data2))
print(f"stdev({data2}) =", stats.stdev(data2))
```

Sample output:

sqrt(5) = 2.236 067 977 499 79

radians(30) = 0.523 598 775 598 298 8

random() in [0,1] = 0.374 484 771 756 488

radiant(216) = 6

pi = 3.141 592 653 589 793

ceil(2.3) = 3

floor(2.3) = 2

factorial(5) = 120

gcd(5,15) = 5

abs(-10) = 10

pow(3,5) = 243

log base 30 of 2 = 0.630 929 753 571 457 4

log 10 (100) = 2.0

isinf(00) = True, isnan(Nan) = True

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:

Step1: Start
Step2: To create a package Card pack
Step3: To create a module Card fun and import random function.
step4: Assign a cards range
step 5: Call a module function
step6: Display the random sample cards.
Step7: Stop

Program:

```
Card fun
import random
def fun():
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
card.fun.fun()
```

output:

Restart:

[5, 29, 13, 22, 20, 41, 38, 51, 47, 34, 49, 14, 50, 37, 40,
15, 35, 17, 18, 33, 39, 36, 42, 12, 16, 19, 48,
29, 2, 27, 11, 31, 46, 28, 21, 32, 8, 25, 30, 23, 26,
10, 43, 47, 3, 44, 52, 1, 4, 5, 9]

3.3 You are tasked with developing a modular calculator application in Python. The calculator should support basic arithmetic operations: addition, subtraction, multiplications, and division. Each operation should be implemented in a separate module. Additionally, you should create a main program to handle user input, call the appropriate module, and display the results.

Algorithm:

1) Define functions for addition, subtraction, multiplications, and division.
2) Handle divison by zero by raising an error if the divison is zero.
3) Import the module (mymath) containing the 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 ValueError("cannot divide by zero")
    return a/b
import mymath
a=10, b=5
print("Addition", mymath.add(a,b))
Print("subtraction", mymath.subtract(a,b))
Print("Multiplication", mymath.multiply(a,b))
Print("Division", mymath.divide(a,b))
```

8.4 Youare 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 named mypacked which includes sub packages pack1 and pack2 with two modules: math functions and area functions Demonstrate the use of the functions by performing a few calculation and printing the results.

Algorithm:

1) Create math functions.py module.
2) Create area functions.py module
3) Create main.py:
4) Print the output as expected.

Program:

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

output:

Addition : 15

Subtraction : 5

Multiplication : 50

Division : 2.0

Circle Area (radius = 7) : 153.93804002589985

Rectangle Area (5 x 10) : 50

Triangle Area (base = 6, height = 8) : 24.0

```
import math functions
import area functions
#using math functions
Print(" Addition:" math functions.add(10,5))
Print("subtration", mathfunction,subtract (10,5))
Print("multiplication", mathfunctions.multiply(10,5))
Print(" Division", math functions.divide(10,5))
#using area functions
Print("Circle Area (radius = 7): ", areafunctions.circle_area(7))
Print("Rectangle Area (5x10:" area functions.retangle_are(5,10))
Print("Triangle Area (base=6, height=8);", area function :trangle_
                                                area (6,8))
```

VEL TECH	
EX NO.	3
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	
SIGNATURE WITH DATE	20

Result: Thus the program for importing python Modules and packages was successfully executed and the output was verified.