

Task-3:- Importing and creating Python modules and packages in python program.

Date: 18/8/25

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

3.1.1.

- 1) Perform common math and random operations.
- 2) Work with 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, pathlib.
- 2) math & random:
 - compute sqrt(5), radians(30), a random float in [0, 0.1], a random integer in [2, 6] (inclusive), pi, 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\plots2u if not present and change the current working directory to it.
 - print python interpreter version.
- 4) statistics:
 - on lists [5, 6, 8, 10] and [2, 5, 3, 2, 8, 3, 9, 4, 2, 5, 6] compute mean, median, mode, stdDev.
- 5) print the neatly formatted results.

Program:-

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

Expected sample output:

-- MATH & RANDOM --

$\text{sqrt}(5) = 2.23606797749979$

$\text{radians}(30) = 0.5235987755982988$

$\text{random}() \in [0, 1] = 0.57444871$ \leftarrow will be very

$\text{randint}(2, 6) = 6$ \leftarrow inclusive, will vary

$\pi = 3.141592653538471$

$\text{ceil}(2.3) = 3$

$\text{floor}(2.3) = 2$

$\text{factorial}(5) = 120$

$\text{gcd}(5, 15) = 5$

$\text{abs}(-10) = 10$

$\text{pow}(3, 5) = 243$

$\log_{\text{base } 3 \text{ of } 2} = 0.6309297535371451$

$\log_{10}(100) = 2.0$

$\text{isinf}(\infty) = \text{True}, \text{isnan}(NaN) = \text{True}$

-- OS & SYS --

Created /ensured: C:\pythonlab

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

Created /ensured & changed into: C:\python\\$lot\\$214

Directory contents of C:\python\\$lot\\$214: []

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

-- STATISTICS --

$\text{mean}([5, 6, 8, 10]) = 7.25$

$\text{median}([5, 6, 8, 10]) = 7.0$

$\text{mode}([3, 3, 3, 8, 3, 9, 9, 2, 5, 6]) = 3$

$\text{stdev}([2, 5, 3, 3, 8, 3, 9, 4, 3, 5, 6]) = 2.2715633383201093$

Expected sample outputs have nothing to do with the

-- MATH & RANDOM --

$\sqrt{5} = 2.236067977$

$\text{radians}(30) = 0.52359877559872$

$\text{random}() \text{ in } [0, 1] = 0.374448871756$ <- will vary

$\text{randint}(3, 6) = 6.$ <- inclusive, will vary

$\pi = 3.14$

$\text{ceil}(2.3) = 3$

$\text{floor}(2.3) = 2$

$\text{gcd}(5, 15) = 5$

$\text{abs}(-10) = 10$

$\text{pow}(3, 5) = 243$

$\log \text{ base } 3 \text{ of } 2 = 0.63092975$

$\log_{10}(100) = 2.0$

in $\text{isinf}(\infty) = \text{True}$, $\text{isnan}(\text{NaN}) = \text{True}$

(10.0) \rightarrow OS & sys

current working directory: C:\Python\lab

current working directory: C:\ - (your current Path)

created / ensured & changed into: C:\Python\slotS2L4

Directory contents of C:\Python\slotS2L4:[C]

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

-- STATISTICS --

$\text{mean}([5, 6, 8, 10]) = 7.25$

$\text{median}([5, 6, 8, 10]) = 7.0$

$\text{mode}([3, 5, 3, 3, 8, 3, 9, 4, 3, 5, 6]) = 3$

$\text{stdev}([3, 5, 3, 3, 8, 3, 9, 4, 3, 5, 6]) = 2.271563$

- emerging

alarm trigger

motor trigger

co trigger

rc trigger

state is set to trigger

trigger dictionary most

```
print("In -- MATH & RANDOM --")
print("sqrt(5) = ", math.sqrt(5))
print("radians(30) = ", math.radians(30))
print("random() in [0,1] = " random.random())
print("randint(3,6) = " random.randint(3,6)) # inclusive
print("pi = " math.pi)
Print("ceil(2.3) = " math.ceil(2.3))
Print("floor(2.3) = " math.floor(2.3))
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.log10(a-val))
inf_val = float('inf')
nan_val = float('nan')
print(f"isnan(x) = {math.isnan(inf_val)}, isnan(Nan) = {math.isnan(nan_val)}".)

```

```
Print("In -- 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.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("In -- STATISTICS --")~~

data1 = [5, 6, 8, 10]

data12 = [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({data1}) = " stats.mode(data1))

print(f"stddev({data1}) = " stats.stdev(data1))

Output:-

RESTART:

c:\users\student.MATZVC833\appdata\local\program\python3.11
packages\card pack\my mod.py

[5, 29, 13, 22, 29, 41, 38, 51, 4, 7, 34, 39, 40, 15, 35, 17, 18, 33, 36, 42
42, 12, 6, 16, 14, 29, 2, 27, 11, 46, 32, 8, 25, 30, 44, 52, 145, 9]

((21,2) bsp. dictone.) = ((21,2) bsp.) taking

((60,2) dictone.) = ((60,2) dictone) taking

((2,2) work.) = ((2,2) work) taking

((6,2) pol. others.) = 5 to e and pol) taking

((10,2) set pol. others.) = ((10,2) set pol) taking

((10,2) twell) = 10 - 10

((10,2) twell) = 10 - 10

((10,2) twell) = ((10,2) twell) taking

((10,2) twell) = ((10,2) twell)

((10,2) twell) = ((10,2) twell) taking

(3.2) Task

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, and display.

Algorithm:-

Step 1: Start

Step 2: To create a package cardpack

Step 3: To create module cardFun

Step 4: Assign a cards range

Step 5: call a module function

Step 6: Display the random sample cards.

Step 7: Stop

Program:-

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

Mymod. Py

```
import cardFun  
cardFun.func()
```

(ministries) have been formed especially central & regional, sub-area members are formed but not but system is Output:- when too many members in this case for Spry a problem

Addition : 13

Subtraction : 5

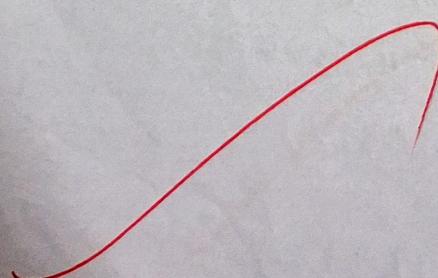
multiplication : 50

Division : 2.0

members especially, others of : 50
members sub-area others of : 50
especially others & regional 100
central, sub-area & others : 200
others sub-area members are placed : 100
others : 50

- (members)
out but
members together
- (members) who
() others
: (50) especially others
() members others
(50+200) members - different
("out" others - different, "out") same

eg. members
members together
() out, others



(33) Task:- You are tasked with developing a modular calculator application in Python.

Algorithm:-

1. Define function for addition, subtraction, multiplication, division.
2. Handle division by zero by raising division is zero.
3. Import the module (mymath) containing these functions.
4. Call each function using mymath. <function name>(a,b)
5. 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))
```

Output:

Addition : 15

Subtraction : 5

multiplication : 50

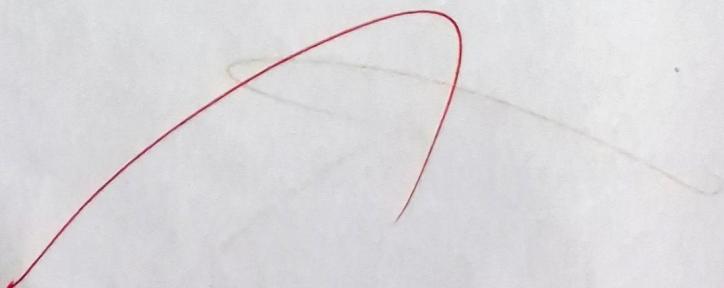
Division : 2.0

Circle Area (radius=7) : 153.9380433

Rectangle Area (5x10) : 50

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

((d, e) takes Δ area for "width/height") true
 (d, e) takes Δ area for "midpoint" true
 (d, e) plithus Δ area for "center/center" true
 (d, e) stevins Δ area for "midline" true



(3.4) Task:- you are working project the requires to perform various mathematical operation and geometric area calculations. To organize your code better, you decide to create a package named mypackage which includes sub package pack 1 and pack 2 with two modules a few calculation 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 sub(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 areafunctions .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.
```

3. Create the main.py file

```
# import mathfunctions  
# import areafunctions  
# Using math functions  
print("Addition:", mathfunction.add(10, 5))  
print("Subtraction:", mathfunction.sub(10, 5))
```

```

print("Division:", mathfunction.divide(10, 5))

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

```

(using loops, while loops, for loops)

Algorithm:

1. Start
2. Create an empty list sales
3. For 7 days input integer value to sales
4. Compute total = sum(sales) and calculate E: per day
5. Find max and min(sales) and calculate R: range
6. Find corresponding days with max and min sales
7. Stop

program (using append(), index(), count(),

List scenario

days = 7

sales = []

for i in range(7):

Results:-

thus, the program for importing python modules and package successfully executed and the output was verified.

VEL TECH - CSE	
EX NO.	3
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	
GN WITH DATE	15/05