

Program:

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

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(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({a_val}) = ", math.log10(a_val))
inf_val = float('inf')
nan_val = float('nan')
printf(f"isnan(inf) = {math.isnan(math.inf)}, isnan(nan) = {math.isnan(nan)}")
printf(f"isnan(nan_val) = {math.isnan(nan_val)})")

print("In -- OS & SYS --")
path_pythonlab = Path(r"C:\Pythonlab")
path_pythonlab.mkdir(parents=True, exist_ok=True)
printf("Created/ensured: {path_pythonlab}")
printf("Current working directory: ", os.getcwd())
target_dir = Path(r"C:\PythonSolutions")
target_dir.mkdir(parents=True, exist_ok=True)
```

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

- 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.

- Algorithm:

1) Import required modules: math, random, os, sys, statistics, pathlib.

2) Math & random:

- Computer sqrt(5), radians(30), a: random float in [0.0, 1.0], 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:\PythonLabSQL 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 [2, 5, 3, 2, 8, 3, 9, 4, 2, 5, 6], compute mean, median, mode, stdev.

5) Print neatly - formatted results.

```

os.chdir(target_dir)
print(f"Changed into: {target_dir}")
print("Directory Contents:", os.listdir())
print("Python version:", sys.version)

print("-- 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"stddev({data2})={stats.stdev(data2)}")

```

Output:-

- MATH & Random -

$\sqrt{5} = 2.23606797749979$

$\text{rad} \text{ and}(30) = 0.5235987755982988$

$\text{random() int}[0:1] = 0.37444887175646646$

$\text{radian}(2, 6) = 6$

$\pi = 3.141592653589793$

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

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

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

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

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

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

$\log \text{base } 3 \text{ of } 2 = 0.6309297535714574$

$\log 10(100) = 2.0$

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

-- OS & SYS -

Created/ensured: c:\python\lab

current working directory: c:\ -- (your current path)

Created/ensured & changed into: c:\python\lab s2L4

Directory contents of c:\python\lab s2L4: []

python version: 3.10.0 (details)

-- STATISTICS --

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

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

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

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

Program :

Cardfun

```
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")
```

Mymod.py

```
import CardFun
CardFun.func()
```

Output :

RESTART:

C:\Users\student.MAT2NC6833\AppData\Local\Programs\Python\Python31\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, 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]
```

3.2 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, and display a random sample of cards.

-Algorithm:-

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

(*Macrorhynchus* "candidus") 1919
(*Macrorhynchus* "candidus") 1919
(*Macrorhynchus* "candidus") 1919
(*Macrorhynchus* "candidus") 1919

May 2003

```
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
```

3.3 You are tasked with developing 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.

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, multiplication, and division.
2. Handle division by zero by raising an error if the divisor is zero.
3. Import the module (mymath) 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.

Output:

After execution of mymath.py

addition from mymath

subtraction from mymath

multiplication from mymath

division from mymath

addition from mymath

subtraction from mymath

multiplication from mymath

division from mymath

addition from mymath

subtraction from mymath

multiplication from mymath

division from mymath

addition from mymath

Programs:

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 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
```

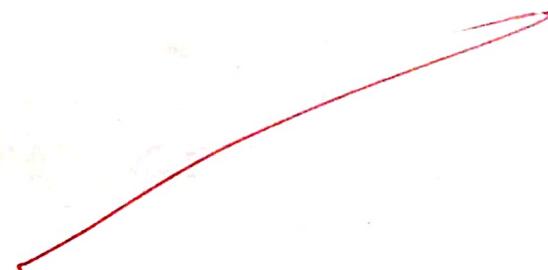
3. Create the main.py file

```
import mathfunctions  
import areafunctions  
  
# Using math functions  
print("Addition:", mathfunctions.add(10,5))  
print("Subtraction:", mathfunctions.subtract(10,5))  
print("Multiplication:", mathfunctions.multiplication(10,5))  
print("Division:", mathfunctions.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))
```

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 named mypackage 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 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.



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

1. ~~Program to print the first 10 prime numbers~~
 2. ~~Program to print the first 10 even numbers~~
 3. ~~Program to print the first 10 odd numbers~~
 4. ~~Program to print the first 10 perfect numbers~~
 5. ~~Program to print the first 10 amicable numbers~~
 6. ~~Program to print the first 10 happy numbers~~
 7. ~~Program to print the first 10 deficient numbers~~
 8. ~~Program to print the first 10 abundant numbers~~
 9. ~~Program to print the first 10 narcissistic numbers~~
 10. ~~Program to print the first 10 palindromic numbers~~
Project :- Python Application for Data Science

Project Name : **PyData**
 Project Leader : **Ramya Sankar**
 Project Members :
 1. **Abhishek**
 2. **Pranav**
 3. **Sai**
 4. **Shivani**
 5. **Vishal**

VELTECH	
EXPO.	3
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	1
TOTAL (20)	15
SIGN WITH DATE	

Result:- Thus, the program for importing Python modules and packages was successfully created and the output was verified