**Batch: A1 Roll No.: 1911004**

**Experiment / assignment / tutorial No.3**

**Grade: AA / AB / BB / BC / CC / CD /DD**

**Signature of the Staff In-charge with date**

|  |
| --- |
| **TITLE :** Modules and Packages |

**AIM:** To implement modules and Packages

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Expected OUTCOME of Experiment:**

CO2- Interpret different Decision Making statements, Functions, Object oriented programming in Python.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Books/ Journals/ Websites referred:**

1. PPT’s Provided & Lab Notes

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Pre Lab/ Prior Concepts:**

**Modules** – They refer to a file containing Python statements and definitions. A file containing Python code, for example: example.py, is called a module, and its module name would be example. We use modules to break down large programs into small manageable and organized files. Furthermore, modules provide reusability of code. We can define our most used functions in a module and import it, instead of copying their definitions into different programs. We can import the definitions inside a module to another module or the interactive interpreter in Python.

We use the import keyword to do this. Using the module name we can access the function using the dot . Operator.

Python has tons of standard modules. These files are in the Lib directory inside the location where you installed Python. Standard modules can be imported the same way as we import our user-defined modules.

**Packages** – We don't usually store all of our files on our computer in the same location. We use a well-organized hierarchy of directories for easier access. Similar files are kept in the same directory. Analogous to this, Python has packages for directories and modules for files.

As our application program grows larger in size with a lot of modules, we place similar modules in one package and different modules in different packages. This makes a project (program) easy to manage and conceptually clear. Similarly, as a directory can contain sub-directories and files, a Python package can have sub-packages and modules. A directory must contain a file named init .py in order for Python to consider it as a package. This file can be left empty but we generally place the initialization code for that package in this file.

**Implemented Program**:

#myMath.py in myPackage ====part1 Package & Module===

class myMath:

@staticmethod

def power(x,y):

return (x\*\*y)

@staticmethod

def factorial(n):

if (n==1 or n==0):

return 1

else:

return (n\*myMath.factorial(n-1))

#exp3.py with package & module

from myPackage.myMath import myMath

def Sin(x,n):

sin=0.0

for i in range(0,n+1):

sin+=((myMath.power(-1,i)\*(myMath.power(x,2\*i+1)/(myMath.factorial((2\*i+1))))))

return sin

def Cos(x,n):

cos=0.0

for i in range(0,n+1):

cos+=((myMath.power(-1,i)\*(myMath.power(x,2\*i)/(myMath.factorial((2\*i))))))

return cos

def Tan(x,n):

tan = Sin(x,n)/Cos(x,n)

return tan

print(‘With Module myMath in Package myPackage’)

x = float(input("Enter angle in degree "))

x=x\*3.14/180

n = int(input("Enter no of terms "))

print()

print('sin',Sin(x,n))

print('cos',Cos(x,n))

print('tan',Tan(x,n))

#myMath.py as module ====part2 Module====

class myMath:

@staticmethod

def power(x,y):

return (x\*\*y)

@staticmethod

def factorial(n):

if (n==1 or n==0):

return 1

else:

return (n\*myMath.factorial(n-1))

#exp3.py with module

from myPackage.myMath import myMath

def Sin(x,n):

sin=0.0

for i in range(0,n+1):

sin+=((myMath.power(-1,i)\*(myMath.power(x,2\*i+1)/(myMath.factorial((2\*i+1))))))

return sin

def Cos(x,n):

cos=0.0

for i in range(0,n+1):

cos+=((myMath.power(-1,i)\*(myMath.power(x,2\*i)/(myMath.factorial((2\*i))))))

return cos

def Tan(x,n):

tan = Sin(x,n)/Cos(x,n)

return tan

print(‘With Module myMath’)

x = float(input("Enter angle in degree "))

x=x\*3.14/180

n = int(input("Enter no of terms "))

print()

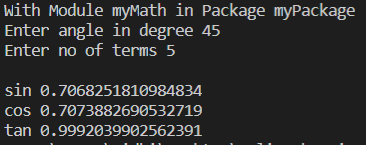
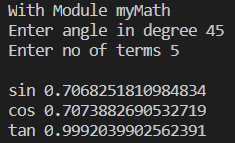
print('sin',Sin(x,n))

print('cos',Cos(x,n))

print('tan',Tan(x,n))

**Output:**

* + - 1. With Package & Module With Module

**Conclusion:**

We successfully implemented concept of Modules & Package in python ; thus getting correct necessary output

**Post Lab Descriptive Questions**

**what is the output of this code?**

Import random as r ; print(random.randomint(1,10)

1. **An error occurs**
2. 1
3. 10
4. Any random value

**2. Every package in Python is a directory which has a special file called** **\_\_init\_\_.py**

**3. Which variables are usually used to keep the count of number of objects created from a class**

1. **class**
2. object
3. ordinary
4. temporary

**4. which function return True if the object is an instance of the class or other classes derived from it?**

1. isobject()
2. issubclass()
3. **isinstance()**
4. issuperclass()

**5. When we add two objecs of class Complex, which functions are called when write print(C1+C2)**

1. **\_add\_(), \_str()\_**
2. \_str()\_, \_add\_()
3. \_sum()\_,\_str()\_
4. \_str()\_, \_sum\_()