**Practical No.8**

**A] Open a new file in IDLE (“New Window” in the “File” menu) and save it as geometry.py in the directory where you keep the files you create for this course. Then copy the functions you wrote for calculating volumes and areas in the “Control Flow and Functions” exercise into this file and save it. Now open a new file and save it in the same directory. You should now be able to import your own module like this: import geometry Try and add print dir(geometry) to the file and run it. 26 Now write a function pointyShapeVolume(x, y, squareBase) that calculates the volume of a square pyramid if squareBase is True and of a right circular cone if squareBase is False. x is the length of an edge on a square if squareBase is True and the radius of a circle when squareBase is False. y is the height of the object. First use squareBase to distinguish the cases. Use the circleArea and squareArea from the geometry module to calculate the base areas**

**Code:**

**Geometry.py**

import math

def SquareArea(sid\_len):

return sid\_len\*\*2

def CircleArea(radius):

return math.pi \*radius\*\*2

def SquarePyramidVolume(base\_edge,height):

basearea=SquareArea(base\_edge)

return basearea\*height/3

def ConeVolume(radius,height):

convol=CircleArea(radius)

return convol\*height/3

def PointyShapeVolume(x,y,squarebase):

if(squarebase):

print(f"The value of square pyramid : {SquarePyramidVolume(x,y)}")

else:

print(f"The value of cone : {ConeVolume(x,y)}")

**module.py**

import geometry

geometry.PointyShapeVolume(6,3,True)

geometry.PointyShapeVolume(6,3,False)

**Output:**



**B] . Write a program to implement exception handling**

**Code:**

class ValueTooHighError(Exception):

def \_\_init\_\_(self,msg):

self.msg=msg

super().\_\_init\_\_(self.msg)

def check\_age(age):

if age<18:

raise ValueTooHighError("Person is invalid")

try:

a=int(input("Enter your age:"))

check\_age(a)

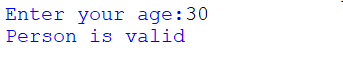
print("Person is valid")

except ValueTooHighError as e:

print(f"Error:{e}")

**Output:**

****

****