ISE –III Answer Sheet

# NAME: SAUBAAN M. SHAIKH

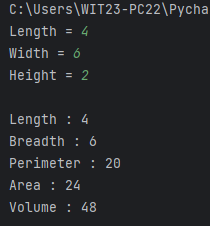
# ROLL NO.: C- 38

# Q1. Create a Rectangle class in Python that enables the construction of a rectangle with length and width attributes. The class contains a find\_perimeter() method to calculate the perimeter of the rectangle and find\_area() method to calculate the area of the rectangle..Additionally, it contains display() method that displays the length, width, perimeter, area of rectangle. Then, create a Parallelepiped child class that inherits from the Rectangle class, including a height attribute and a find\_volume() method to calculate the volume of the Parallelepiped.(volume of parallelepiped=length \* width \* height ) and display\_volume() method that displays volume of an object created using an instantiation of the Parallelepiped class.

## PROGRAM:

class Rectangle:  
 def \_\_init\_\_(self, length, width):  
 self.\_length = length  
 self.\_width = width  
  
 def find\_perimeter(self):  
 peri = 2 \* (self.\_length + self.\_width)  
 return peri  
  
 def find\_area(self):  
 area = self.\_width \* self.\_length  
 return area  
  
 def display(self):  
 print(f"\nLength : {self.\_length}\nBreadth : {self.\_width}")  
 print(f"Perimeter : {self.find\_perimeter()}\nArea : {self.find\_area()}")  
  
  
class ParallelPiped(Rectangle):  
 def \_\_init\_\_(self, height, length, width):  
 super().\_\_init\_\_(length, width)  
 self.\_height = height  
  
 def find\_volume(self):  
 volume = self.find\_area() \* self.\_height  
 return volume  
  
 def display(self):  
 super().display()  
 print(f"Volume : {self.find\_volume()}")  
  
  
ln = int(input('Length = '))  
wd = int(input('Width = '))  
ht = int(input('Height = '))  
  
para = ParallelPiped(ht, ln, wd)  
para.display()

## OUTPUT:



# Q2. Create a Python program for a credit card validation using re module that verifies whether a given credit card number is valid or not. The program should check the credit card number for the following conditions:

# - The credit card number must be 16 digits long.

# - The first digit must be in the range of 4 to 6 i.e. It must start with a 4,5 or 6.

# - All characters must be digits (0-9) only.

# - It may have digits in groups of 4 separated by one hyphen "-".

# Program:

import re  
  
  
def validate(card\_no):  
 nospace = re.sub(r'-', '', card\_no)  
 if not re.match(r'^\d{16}$', nospace):  
 return "The credit card number is not valid."  
  
 if not re.match(r'^[4-6]', card\_no):  
 return "The credit card number is not valid."  
  
 if not re.match(r'^\d+$', nospace):  
 return "The credit card number is not valid."  
  
 return "The credit card number is valid."  
  
  
card\_num = 0  
while card\_num != 'e':  
 card\_num = input("credit\_card\_number = ")  
 result = validate(card\_num)  
 print(result)

# Output:

