CHANDIGARH UNIVERSITY
Discover. Learn. Empower.

EXPERIMENT NUMBER – 3.2

STUDENT'S NAME- Sameer Saifi	**STUDENT'S UID- 20BCS7619**
CLASS AND GROUP- PH20BCS719-A	
SUBJECT- PYTHON PROGRAMMING LAB	
SEMESTER- 4TH	

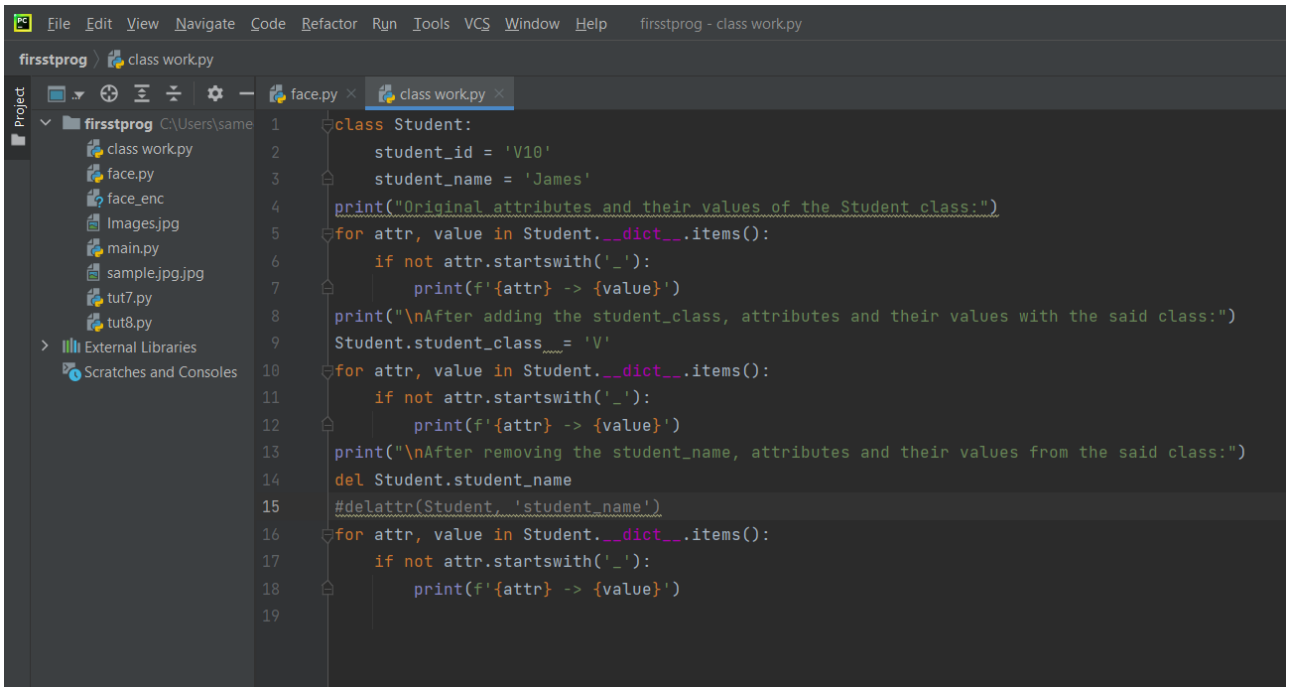
Question 1: **Write a Python class named Student with two attributes student_id, student_name. Add a new attribute student_class and display the entire attribute and their values of the said class. Now remove the student_name attribute and display the entire attribute with values**

CODE IN TEXT FORM

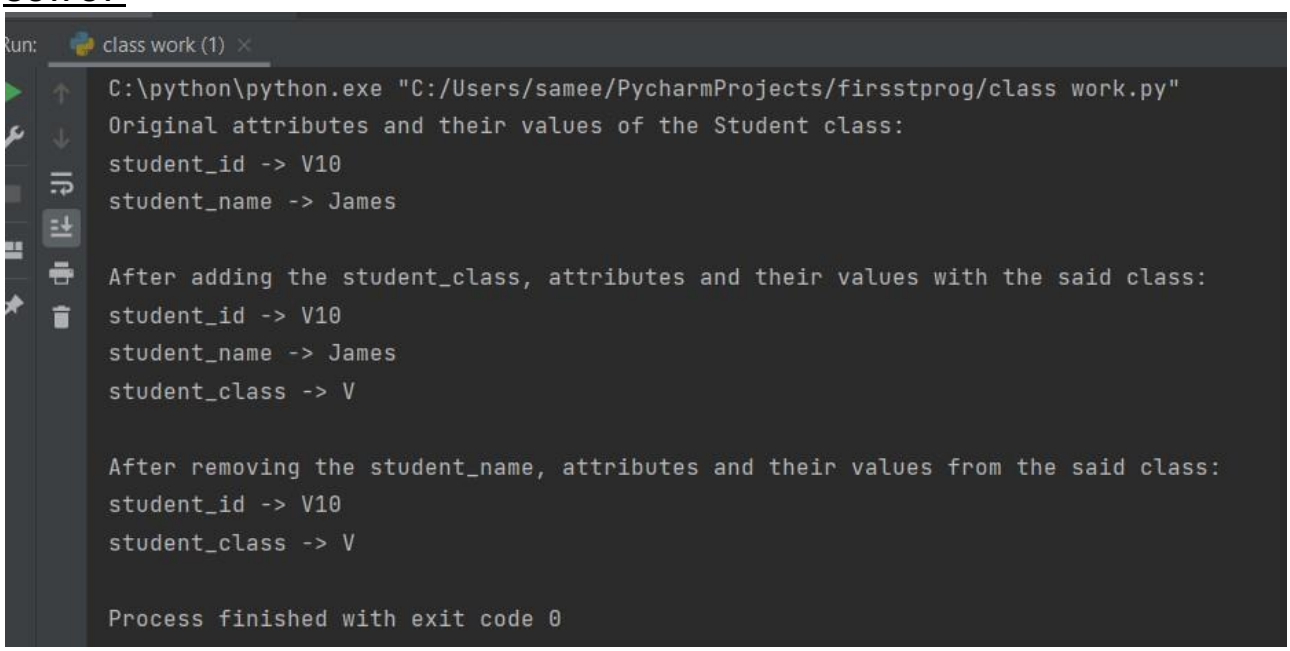
```
class Student:
    student_id = 'V10'
    student_name = 'James'
print("Original attributes and their values of the Student class:")
for attr, value in Student.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
print("\nAfter adding the student_class, attributes and their values with the said class:")
Student.student_class  = 'V'
for attr, value in Student.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
print("\nAfter removing the student_name, attributes and their values from the said class:")
del Student.student_name
#delattr(Student, 'student_name')
for attr, value in Student.__dict__.items():
```

```
if not attr.startswith('_'):
    print(f'{attr} -> {value}')
```

CODE IN COMPILER



OUTPUT-

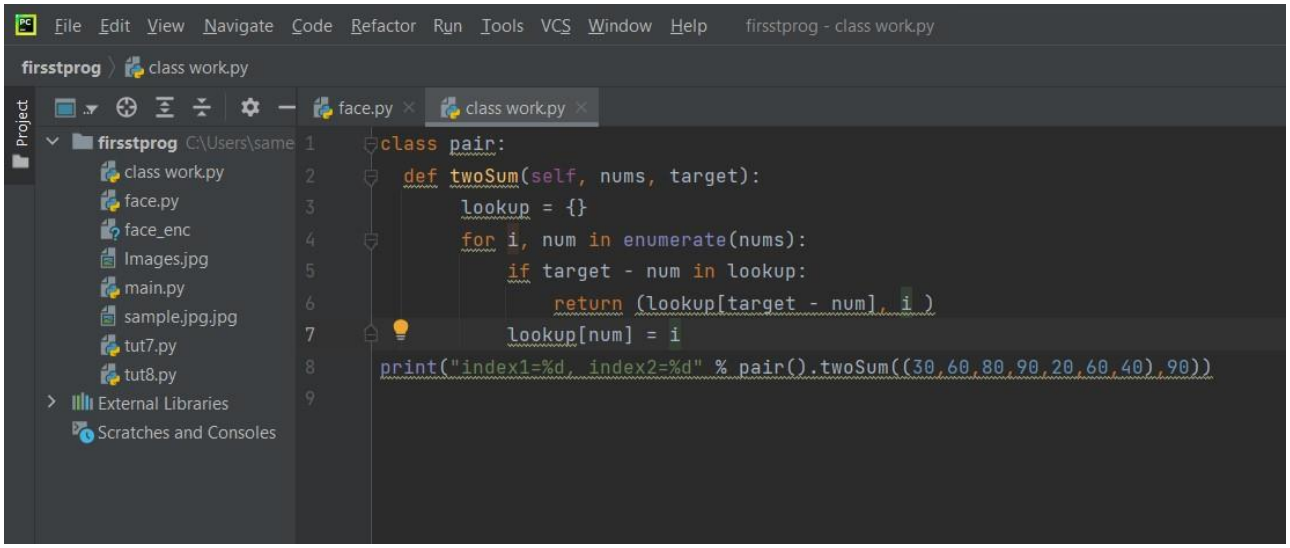


Question 2: Write a Python class to find a pair of elements (indices of the two numbers) from a given array whose sum equals a specific target number.

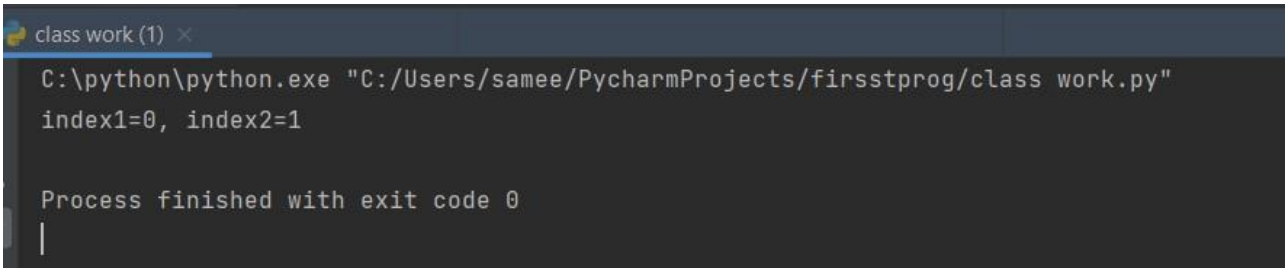
CODE IN TEXT FORM

```
class pair:
  def twoSum(self, nums, target):
      lookup = {}
      for i, num in enumerate(nums):
          if target - num in lookup:
              return (lookup[target - num], i )
          lookup[num] = i
print("index1=%d, index2=%d" %
pair().twoSum((30,60,80,90,20,60,40),90))
```

CODE IN COMPILER –



OUTPUT –

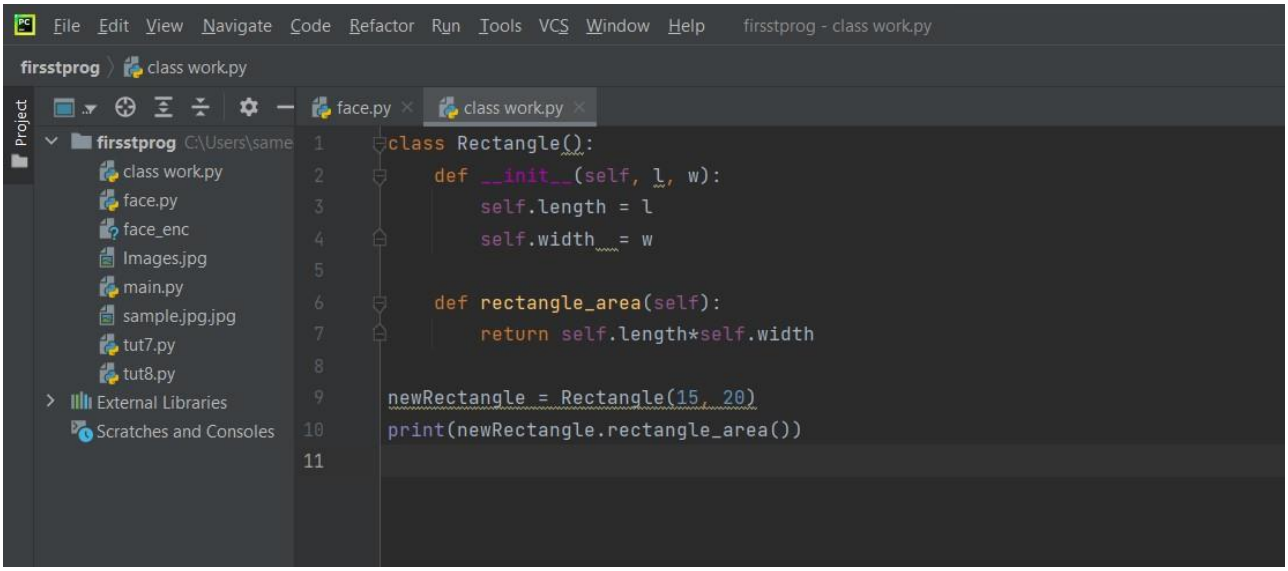

class work (1)
C:\python\python.exe "C:/Users/samee/PycharmProjects/firsstprog/class work.py"
index1=0, index2=1
Process finished with exit code 0

Question 3: Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle

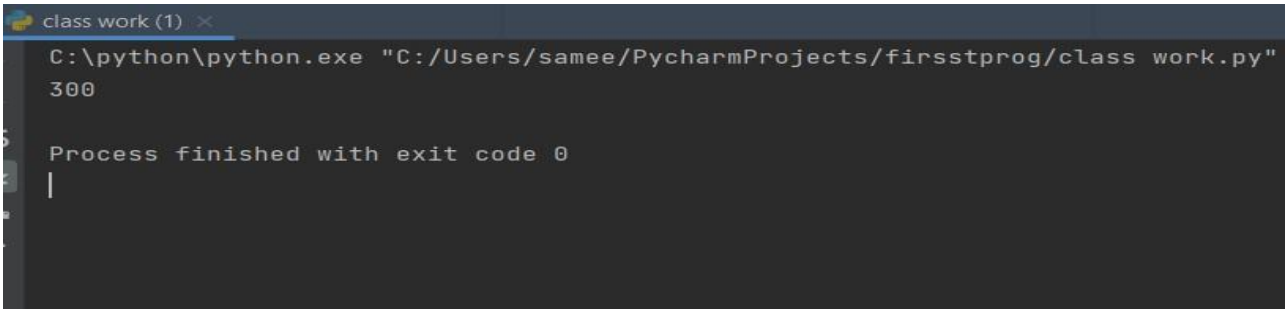
CODE IN TEXT FORM –

```
class Rectangle():
    def __init__(self, l, w):
        self.length = l
        self.width  = w

    def rectangle_area(self):
        return self.length*self.width

newRectangle = Rectangle(12, 10)
print(newRectangle.rectangle_area())
```

CODE IN COMPILER-



OUTPUT –

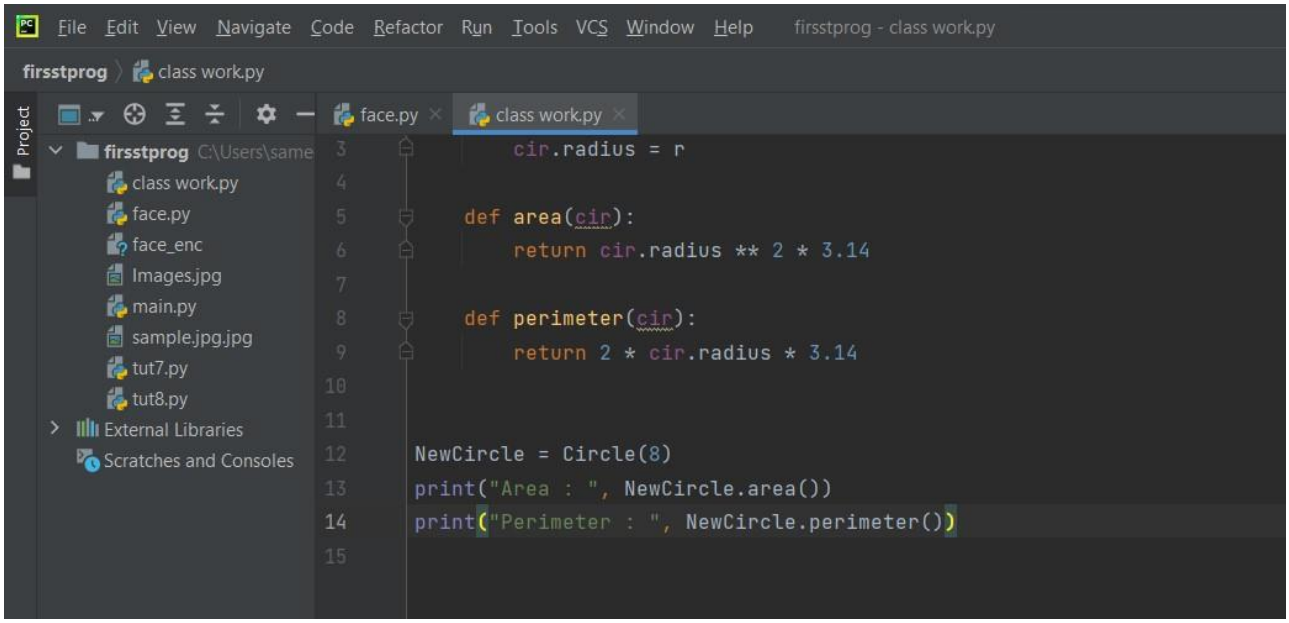


Question 4: Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle

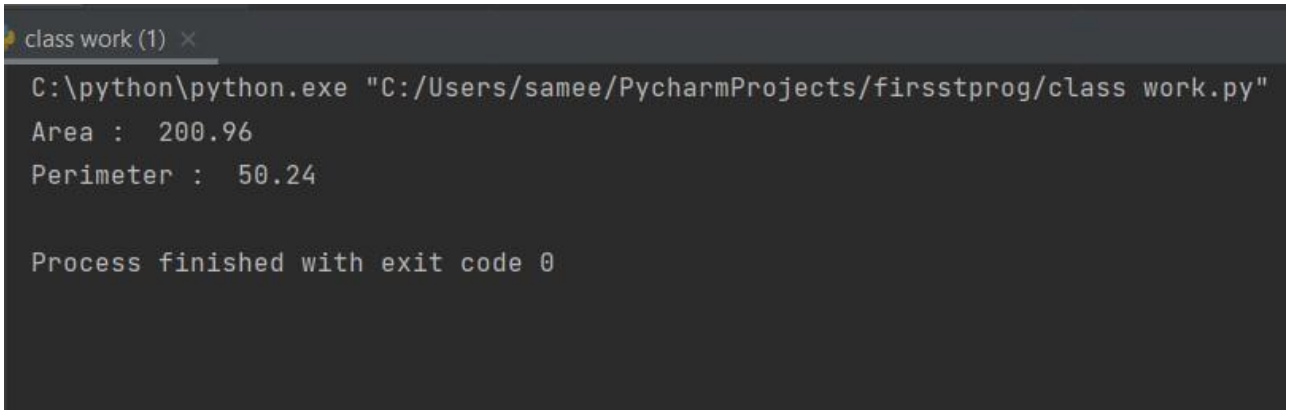
CODE IN TEXT –

```
class Circle():
    def __init__(cir, r):
        cir.radius = r

    def area(cir):
        return cir.radius**2*3.14

    def perimeter(cir):
        return 2*cir.radius*3.14

NewCircle = Circle(8)
print("Area : ",NewCircle.area())
print("Perimeter : ",NewCircle.perimeter())
```

CODE IN COMPILER –



OUTPUT –

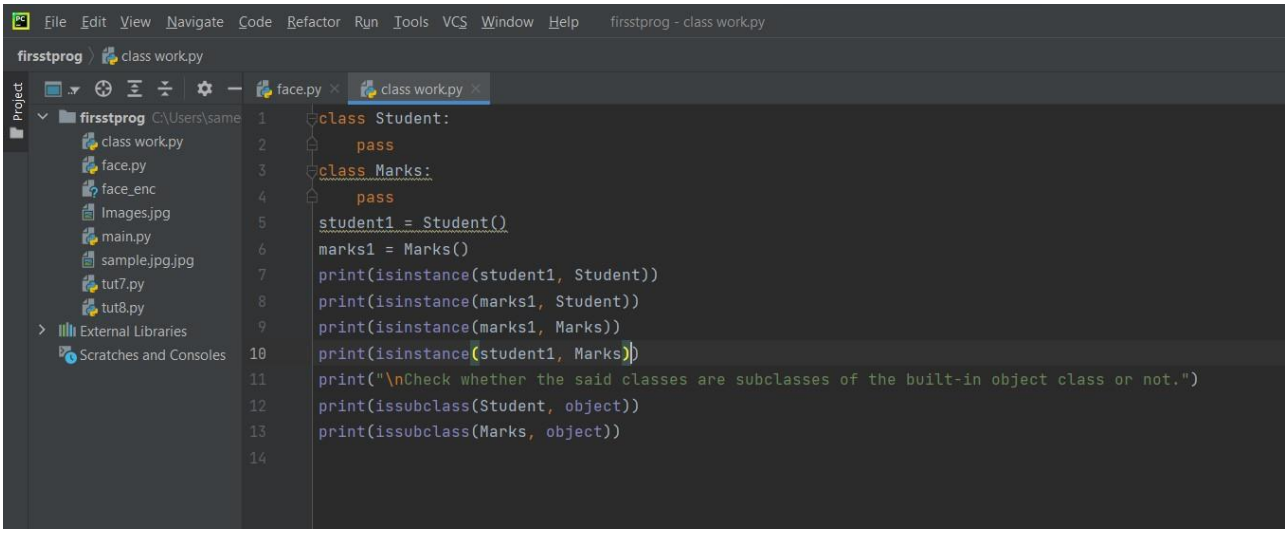


Question 5: Write a Python program to crate two empty classes, Student and Marks. Now create some instances and check whether they are instances of the said classes or not. Also, check whether the said classes are subclasses of the built-in object class or not

CODE IN TEXT –

```
class Student:
    pass
class Marks:
    pass
student1 = Student()
marks1 = Marks()
print(isinstance(student1, Student))
print(isinstance(marks1, Student))
print(isinstance(marks1, Marks))
print(isinstance(student1, Marks))
print("\nCheck whether the said classes are subclasses of the built-in object class or not.")
print(issubclass(Student, object))
print(issubclass(Marks, object))
```

CODE IN COMPILER –



OUTPUT –