

## Worksheet – 3.2

**Student Name:** Vivek Kumar

**UID:** 21BCS8129

**Branch:** BE-CSE (LEET)

**Section/Group:** 809/A

**Semester:** 4th

**Date of Performance:** 02/05/2022

**Subject Name:** Programming in Python Lab

**Subject Code:** 20CSP-259

### 1. Aim/Overview of the practical:

- I. 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.
- II. 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.
- III. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.
- IV. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.
- V. Write a Python program to create 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.

### 2. Task to be done/ Which logistics used:

- I. 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.
- II. 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.
- III. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.
- IV. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.
- V. Write a Python program to create 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.

### 3. Steps for experiment/practical/Code:

- I. 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.

```
class Student:
    student_id = '21BCS8129'
    student_name = 'Vivek Kaumar'

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 = 'Vnj'
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
for attr, value in Student.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
```

- II. 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.

```
def search(a, b):
    for d in b:
        if a == d:
            m = True
            break
    else:
        m = False
    return m

list1 = []
n = int(input('Enter the Size of the List: '))
for i in range(0, n):
    ele = int(input())
    list1.append(ele)

target = int(input("Enter the target: "))
```

```
for i in list1:
    if i < target:
        pair = int(target)-int(i)
        in2 = search(pair, list1)
        if in2 == True:
            print("the first number= %d the second number %d" % (i, pair))
            break
```

- III. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.

```
class Rectangle():
    def __init__(self, a, b):
        self.length = a
        self.breadth = b

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

length = int(input('Enter the Length: '))
breadth=int(input('Enter the Breadth: '))
newRectangle = Rectangle(length,breadth)
area = newRectangle.rectangle_area()
print("Area of Ractangle is: ", area)
```

- IV. Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.

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

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

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

r = int(input('Enter the Radius: '))
NewCircle = Circle(r)
print("Area : ", NewCircle.area())
print("Perimeter : ", NewCircle.perimeter())
```

- V. Write a Python program to create 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.

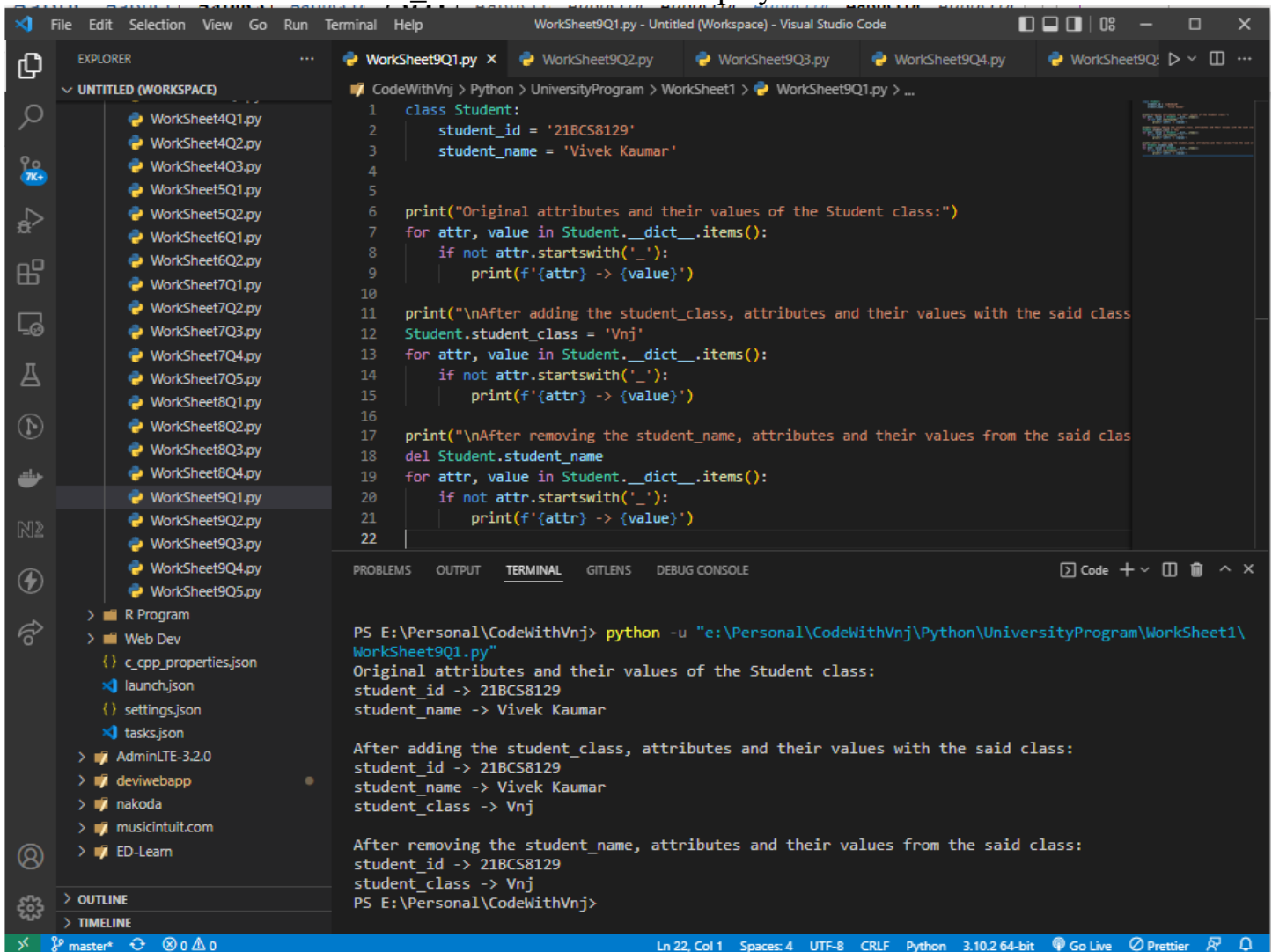
```
class Student:
    pass

class Marks:
    pass

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

## 4. Result/Output/Writing Summary:

- I. 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.



```

1 class Student:
2     student_id = '21BCS8129'
3     student_name = 'Vivek Kaumar'
4
5
6 print("Original attributes and their values of the Student class:")
7 for attr, value in Student.__dict__.items():
8     if not attr.startswith('_'):
9         print(f'{attr} -> {value}')
10
11 print("\nAfter adding the student_class, attributes and their values with the said class")
12 Student.student_class = 'Vnj'
13 for attr, value in Student.__dict__.items():
14     if not attr.startswith('_'):
15         print(f'{attr} -> {value}')
16
17 print("\nAfter removing the student_name, attributes and their values from the said class")
18 del Student.student_name
19 for attr, value in Student.__dict__.items():
20     if not attr.startswith('_'):
21         print(f'{attr} -> {value}')
22

```

```

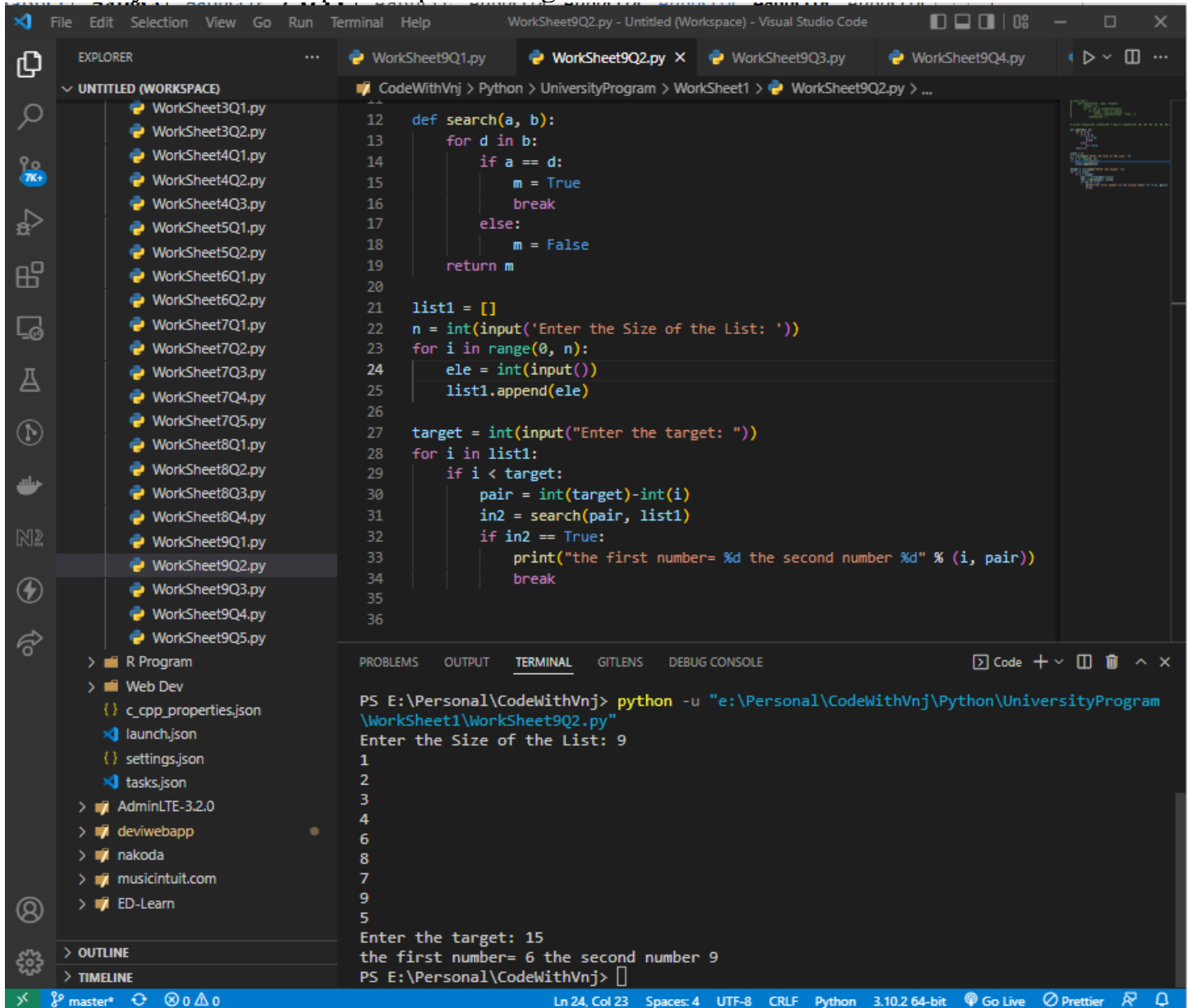
PS E:\Personal\CodeWithVnj> python -u "e:\Personal\CodeWithVnj\Python\UniversityProgram\WorkSheet1\
WorkSheet9Q1.py"
Original attributes and their values of the Student class:
student_id -> 21BCS8129
student_name -> Vivek Kaumar

After adding the student_class, attributes and their values with the said class:
student_id -> 21BCS8129
student_name -> Vivek Kaumar
student_class -> Vnj

After removing the student_name, attributes and their values from the said class:
student_id -> 21BCS8129
student_class -> Vnj
PS E:\Personal\CodeWithVnj>

```

II. 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.



```

11  def search(a, b):
12      for d in b:
13          if a == d:
14              m = True
15              break
16          else:
17              m = False
18      return m
19
20  list1 = []
21  n = int(input('Enter the Size of the List: '))
22  for i in range(0, n):
23      ele = int(input())
24      list1.append(ele)
25
26  target = int(input("Enter the target: "))
27  for i in list1:
28      if i < target:
29          pair = int(target)-int(i)
30          in2 = search(pair, list1)
31          if in2 == True:
32              print("the first number= %d the second number %d" % (i, pair))
33              break
34
35
36

```

Terminal Output:

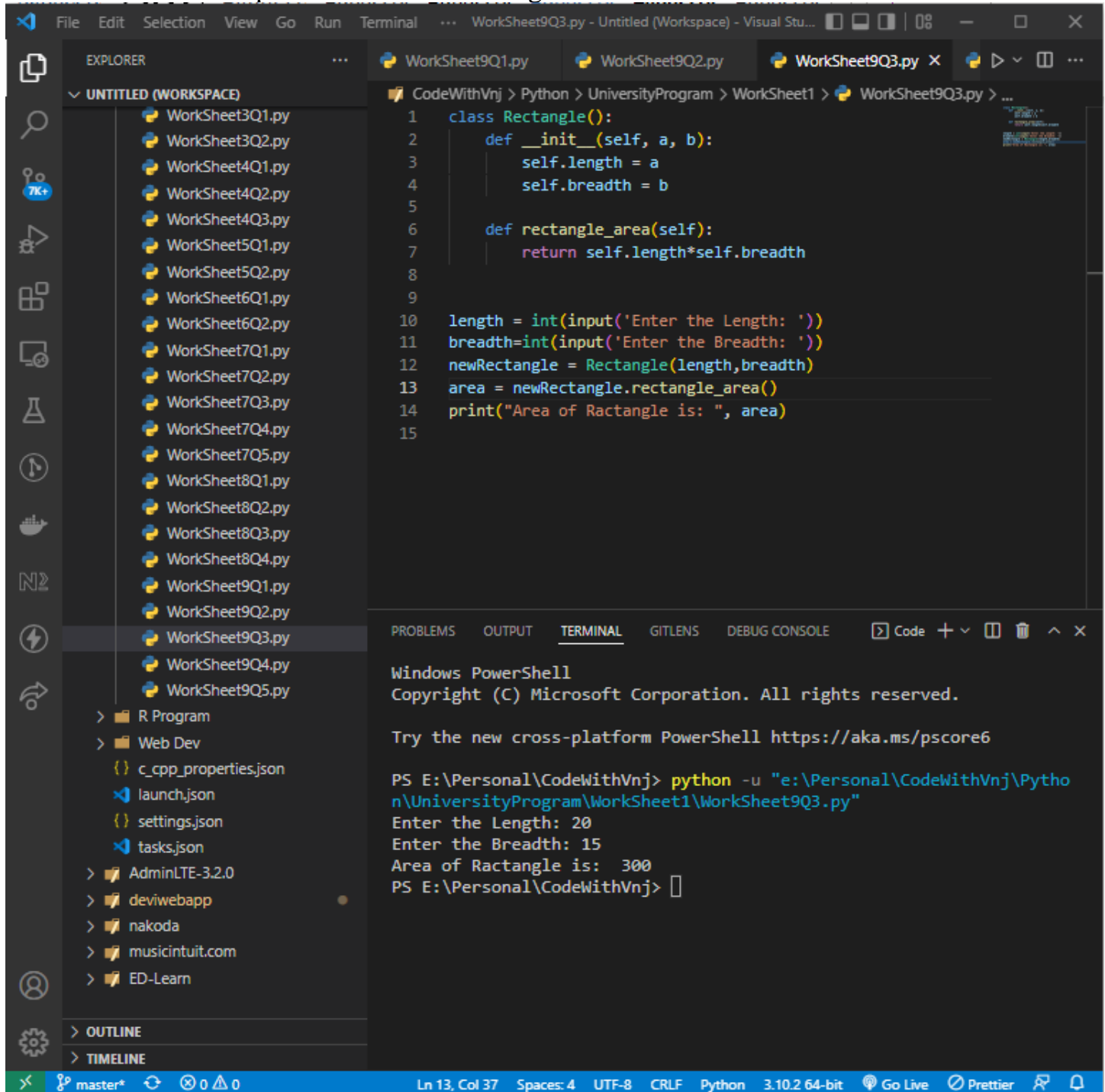
```

PS E:\Personal\CodeWithVnj> python -u "e:\Personal\CodeWithVnj\Python\UniversityProgram\WorkSheet1\WorkSheet9Q2.py"
Enter the Size of the List: 9
1
2
3
4
6
8
7
9
5
Enter the target: 15
the first number= 6 the second number 9
PS E:\Personal\CodeWithVnj>

```



III. Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left lists several Python files, with 'WorkSheet9Q3.py' selected. The main editor displays the code for 'WorkSheet9Q3.py'.

```

1  class Rectangle():
2      def __init__(self, a, b):
3          self.length = a
4          self.breadth = b
5
6      def rectangle_area(self):
7          return self.length*self.breadth
8
9
10 length = int(input('Enter the Length: '))
11 breadth=int(input('Enter the Breadth: '))
12 newRectangle = Rectangle(length,breadth)
13 area = newRectangle.rectangle_area()
14 print("Area of Ractangle is: ", area)
15

```

The TERMINAL pane at the bottom shows the execution of the script in a Windows PowerShell environment:

```

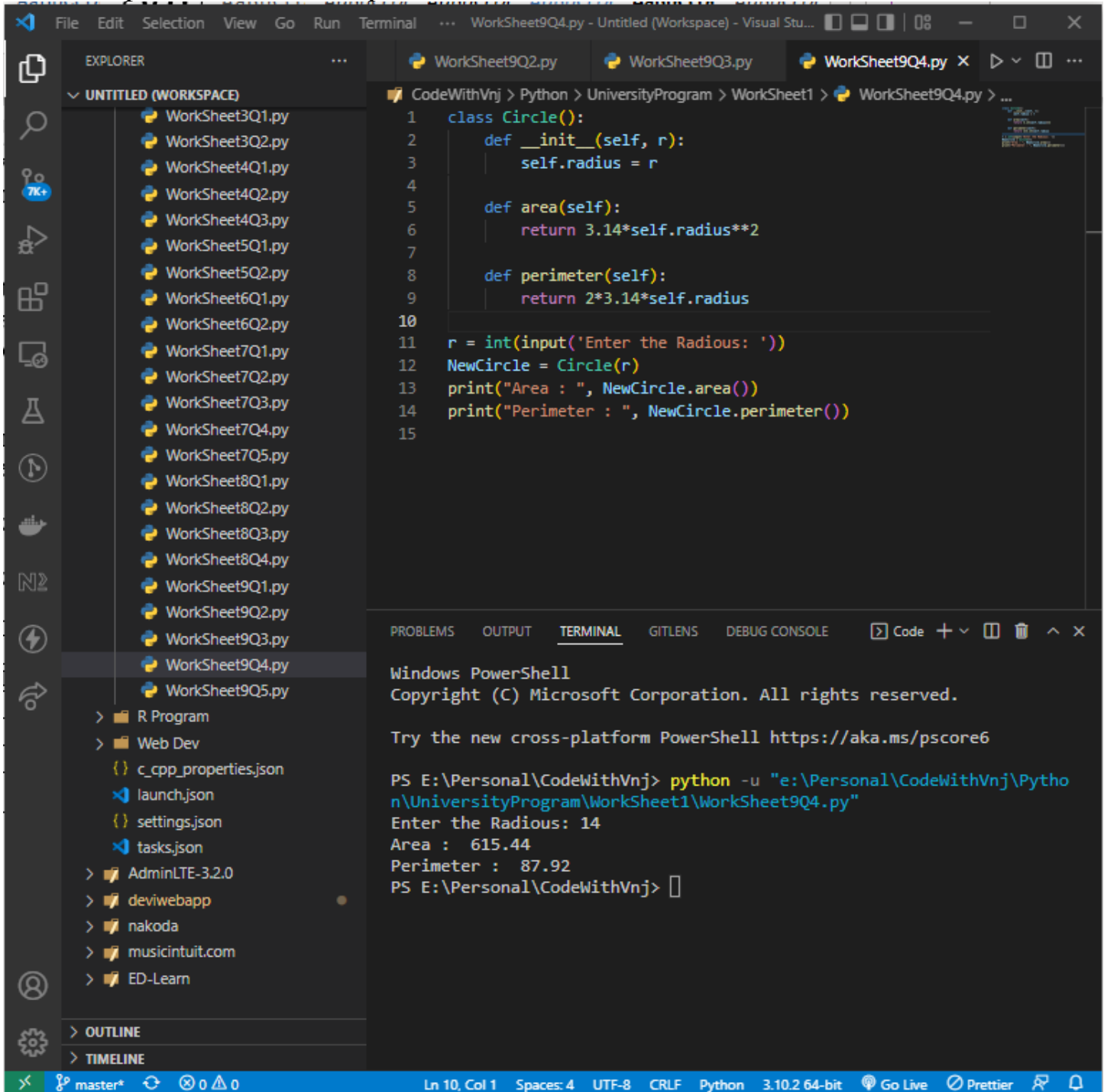
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\Personal\CodeWithVnj> python -u "e:\Personal\CodeWithVnj\Python\UniversityProgram\WorkSheet1\WorkSheet9Q3.py"
Enter the Length: 20
Enter the Breadth: 15
Area of Ractangle is: 300
PS E:\Personal\CodeWithVnj>

```

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



The screenshot shows the Visual Studio Code interface. The Explorer pane on the left lists several Python files, with 'WorkSheet9Q4.py' selected. The main editor displays the code for 'WorkSheet9Q4.py'.

```

1 class Circle():
2     def __init__(self, r):
3         self.radius = r
4
5     def area(self):
6         return 3.14*self.radius**2
7
8     def perimeter(self):
9         return 2*3.14*self.radius
10
11 r = int(input('Enter the Radius: '))
12 NewCircle = Circle(r)
13 print("Area : ", NewCircle.area())
14 print("Perimeter : ", NewCircle.perimeter())
15

```

The TERMINAL pane at the bottom shows the execution of the script in a Windows PowerShell environment:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

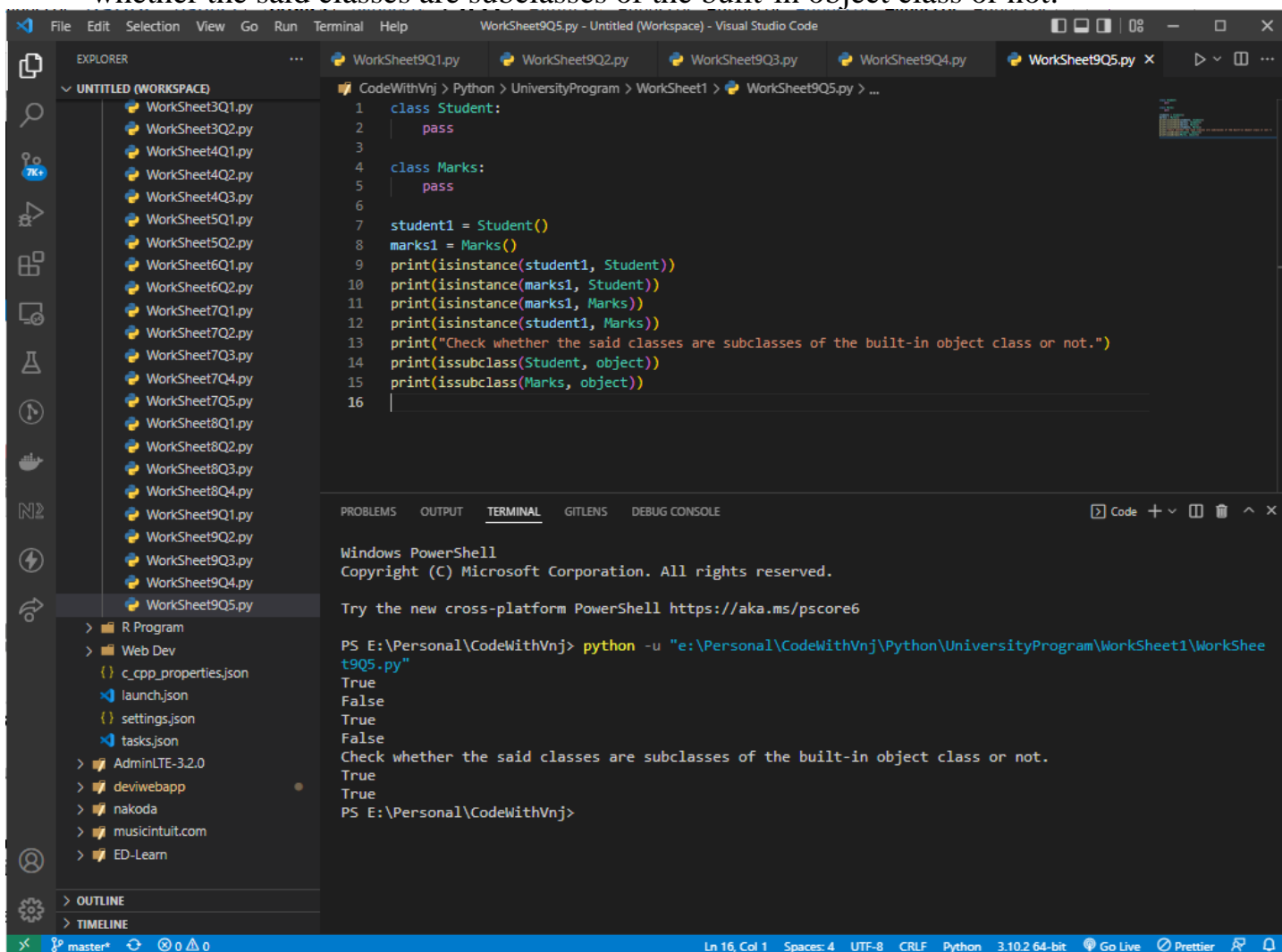
Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\Personal\CodeWithVnj> python -u "e:\Personal\CodeWithVnj\Python\UniversityProgram\WorkSheet1\WorkSheet9Q4.py"
Enter the Radius: 14
Area : 615.44
Perimeter : 87.92
PS E:\Personal\CodeWithVnj>

```



- V. Write a Python program to create 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.



```

1 class Student:
2     pass
3
4 class Marks:
5     pass
6
7 student1 = Student()
8 marks1 = Marks()
9 print(isinstance(student1, Student))
10 print(isinstance(marks1, Student))
11 print(isinstance(marks1, Marks))
12 print(isinstance(student1, Marks))
13 print("Check whether the said classes are subclasses of the built-in object class or not.")
14 print(issubclass(Student, object))
15 print(issubclass(Marks, object))
16

```

```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS E:\Personal\CodeWithVnJ> python -u "e:\Personal\CodeWithVnJ\Python\UniversityProgram\WorkSheet1\WorkSheet9Q5.py"
True
False
True
False
Check whether the said classes are subclasses of the built-in object class or not.
True
True
PS E:\Personal\CodeWithVnJ>

```

## Learning outcomes (What I have learnt):

1. I have learnt, how to create the and manipulate the Classes.

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			
4.			