Exp 3: 3a:

```
# Program on Inheritance

class Person:

  def __init__(self, name, age):

    self.name = name

    self.age = age

  def display(self):

    print("Name:", self.name)

    print("Age:", self.age)

class Teacher(Person):

  def __init__(self, name, age, exp, r_area):

    super().__init__(name, age)

    self.exp = exp

    self.r_area = r_area

  def displayData(self):

    self.display()

    print("Experience:", self.exp)

    print("Research Area:", self.r_area)

class Student(Person):

  def __init__(self, name, age, course, marks):

    super().__init__(name, age)

    self.course = course

    self.marks = marks

  def displayData(self):

    self.display()

    print("Course:", self.course)

    print("Marks:", self.marks)

# Creating objects and displaying data
```

print("***** TEACHER *****")

T = Teacher("Ashfaque", 40, 10, "python")

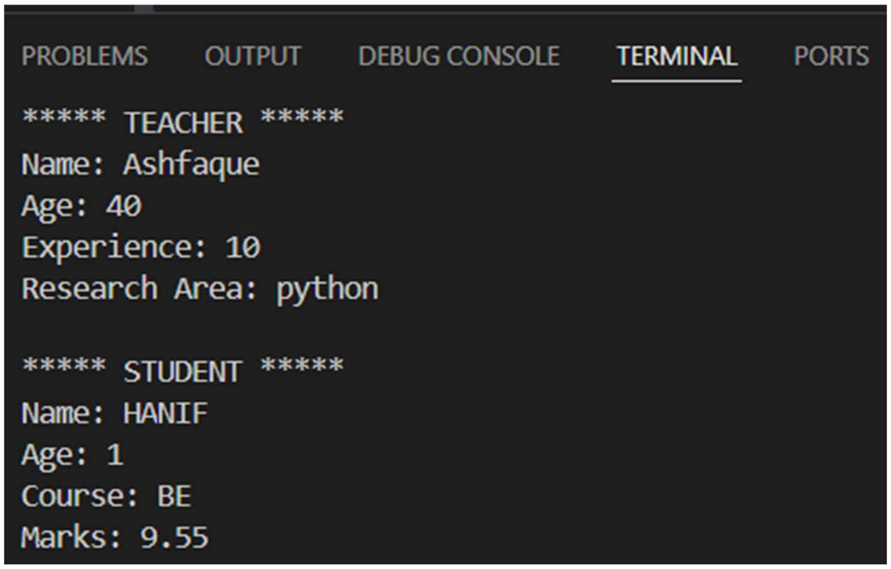
T.displayData()

print("\n***** STUDENT *****")

S = Student("HANIF", 1, "BE", 9.55)

S.displayData()

OUTPUT

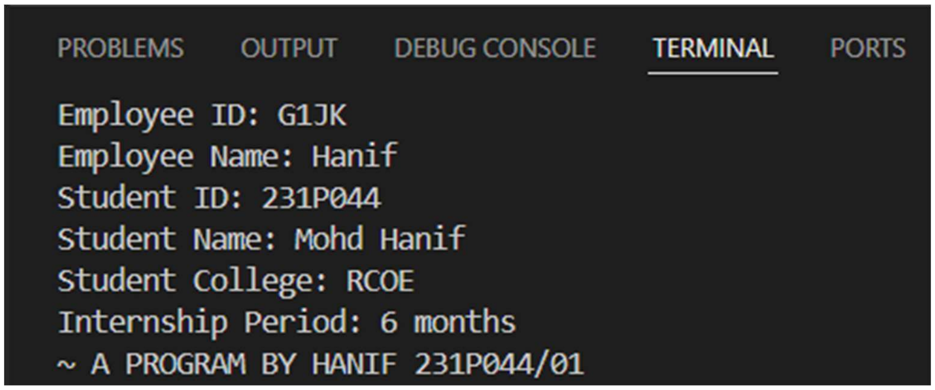

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
***** TEACHER *****
Name: Ashfaque
Age: 40
Experience: 10
Research Area: python

***** STUDENT *****
Name: HANIF
Age: 1
Course: BE
Marks: 9.55
```

3b.

```
"""

Aim: Write a Program in Python to implement Multiple Inheritance.

NAME:  HANIF 231P044/01

"""

class Employee:

  def __init__(self, emp_id, emp_name):

    self.emp_id = emp_id

    self.emp_name = emp_name

  def set_emp_id(self, emp_id):

    self.emp_id = emp_id

  def get_emp_id(self):
        return self.emp_id

    def set_emp_name(self, emp_name):

        self.emp_name = emp_name

    def get_emp_name(self):

        return self.emp_name

class Student:

    def __init__(self, student_id, student_name, student_college):

        self.student_id = student_id

        self.student_name = student_name

        self.student_college = student_college

    def set_student_id(self, student_id):

        self.student_id = student_id

    def get_student_id(self):

        return self.student_id

    def set_student_name(self, student_name):

        self.student_name = student_name

    def get_student_name(self):

        return self.student_name

    def set_student_college(self, student_college):

        self.student_college = student_college

    def get_student_college(self):

        return self.student_college

class Intern(Employee, Student):

    def __init__(self, emp_id, emp_name, student_id, student_name, student_college,
period):

        Employee.__init__(self, emp_id, emp_name)

        Student.__init__(self, student_id, student_name, student_college)

        self.period = period
    def set_period(self, period):
        self.period = period
    def get_period(self):
        return self.period
    def display_intern_details(self):
        print(f"Employee ID: {self.get_emp_id()}")
        print(f"Employee Name: {self.get_emp_name()}")
        print(f"Student ID: {self.get_student_id()}")
        print(f"Student Name: {self.get_student_name()}")
        print(f"Student College: {self.get_student_college()}")
        print(f"Internship Period: {self.get_period()} months")
# Creating an Intern object
intern = Intern(emp_id="G0JK", emp_name="QAYAM", student_id="231P038",
            student_name="Mohd Qayam", student_college="RCOE", period=6)
# Displaying Intern details
intern.display_intern_details()
print("~ A PROGRAM BY HANIF 231P044/01")
```

Output:



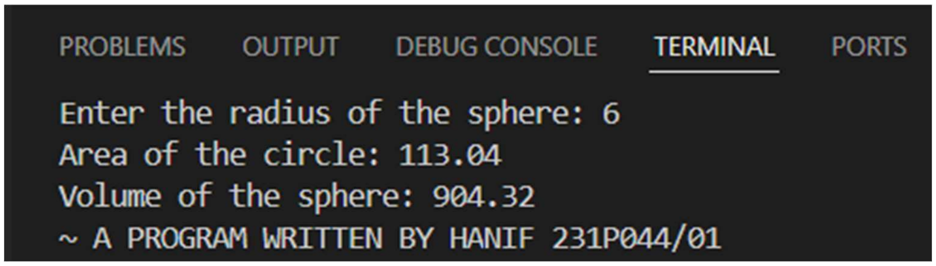
```
Employee ID: G1JK
Employee Name: Hanif
Student ID: 231P044
Student Name: Mohd Hanif
Student College: RCOE
Internship Period: 6 months
~ A PROGRAM BY HANIF 231P044/01
```

3c.

```
"""
Aim: Write a program in Python to calculate the volume of a sphere using multilevel inheritance.
NAME: HANIF 231P044/01
"""
class Circle:
  def __init__(self):
    self.radius = 0
  def accept_radius(self):
    self.radius = float(input("Enter the radius of the sphere: "))
class Area(Circle):
  def __init__(self):
    super().__init__()
  def calculate_area(self):
    area = 3.14 * (self.radius ** 2)
    print(f"Area of the circle: {area:.2f}")
class Volume(Area):
  def __init__(self):
    super().__init__()
  def calculate_volume(self):
    volume = (4/3) * 3.14 * (self.radius ** 3)
    print(f"Volume of the sphere: {volume:.2f}")
# Creating an object of Volume class (which inherits from Area -> Circle)
sphere = Volume()
sphere.accept_radius()
sphere.calculate_area()
sphere.calculate_volume()
```

print("~ A PROGRAM WRITTEN BY HANIF 231P044/01")

Output:



```
Enter the radius of the sphere: 6
Area of the circle: 113.04
Volume of the sphere: 904.32
~ A PROGRAM WRITTEN BY HANIF 231P044/01
```

3d.

```
"""
Aim: Write a program in Python to calculate the volume of a sphere using multilevel
inheritance demonstrating method overriding.
NAME: HANIF 231P044/01
"""
class Circle:
  def __init__(self):
    self.radius = 0
  def accept_radius(self):
    self.radius = float(input("Enter the radius of the sphere: "))


class Area(Circle):
  def __init__(self):
    super().__init__()
  def accept_radius(self):
    print("In Area class: Calculating area of the circle.")
    super().accept_radius()  # Calling parent class method
  def calculate_area(self):
    area = 3.14159 * (self.radius ** 2)
    print(f"Area of the circle: {area:.2f}")
```