

RIPHAH INTERNATIONAL **UNIVERSITY, ISLAMABAD**



Lab#5

Bachelors of Computer Science – 6th Semester

Course: Artificial Intelligence

Submitted to: Ms. Ayesha

Submitted by: Tabinda Hassan

SAP-46374

Date of Submission: 04-03-2025

- present or absent.**

Student with roll number 1 is Present
Student with roll number 2 is Absent
Student with roll number 3 is Present
Student with roll number 4 is Absent
Student with roll number 5 is Present
Student with roll number 6 is Absent
Student with roll number 7 is Present
Student with roll number 8 is Absent
Student with roll number 9 is Absent
Student with roll number 10 is Present
Student with roll number 11 is Absent
Student with roll number 12 is Present
Student with roll number 13 is Absent
Student with roll number 14 is Absent
Student with roll number 15 is Present
Student with roll number 16 is Absent
Student with roll number 17 is Absent
Student with roll number 18 is Present
Student with roll number 19 is Present
Student with roll number 20 is Absent

2. Define a class and create multiple object of class, access attributes and assign new values.

```
class MyClass: 2 usages
    def __init__(self, attribute):
        self.attribute = attribute

obj1 = MyClass(10)
obj2 = MyClass(20)

print(obj1.attribute)
print(obj2.attribute)

obj1.attribute = 30
obj2.attribute = 40

print(obj1.attribute)
print(obj2.attribute)
```

```
"C:\Users\FAR00Q HASSAN
10
20
30
40
```

3. Create a student class with attributes name, age, and grades (list). Add a method average grade that calculates and returns the average of the grades.

```
class Student:
    def __init__(self, name, age, grades):
        self.name = name
        self.age = age
        self.grades = grades

    def average_grade(self):
        return sum(self.grades) / len(self.grades)

student1 = Student(name="Alice", age=18, grades=[85, 90, 92, 88])
print(student1.average_grade())
```

```
"C:\Users\FA
88.75"
```

4. Create a base class Employee with:

- name
- salary
- Method `display_details()` to show employee info.

Create two subclasses:

1. Manager (inherits Employee) and has an additional attribute `department`

2. Developer (inherits Employee) and has an additional attribute `programming_language`

Override the `display_details()` method in both subclasses to include their specific attributes.

```
class Employee: 2 usages
    def __init__(self, name, salary):
        self.name = name
        self.salary = salary

    def display_details(self): 2 usages
        print(f"Name: {self.name}")
        print(f"Salary: {self.salary}")

class Manager(Employee): 1 usage
    def __init__(self, name, salary, department):
        super().__init__(name, salary)
        self.department = department

    def display_details(self): 1 usage
        super().display_details()
        print(f"Department: {self.department}")

class Developer(Employee): 1 usage
    def __init__(self, name, salary, programming_language):
        super().__init__(name, salary)
        self.programming_language = programming_language

    def display_details(self): 1 usage
        super().display_details()
        print(f"Programming Language: {self.programming_language}")

manager1 = Manager( name: "Bob", salary: 80000, department: "Engineering")
developer1 = Developer( name: "Alice", salary: 70000, programming_language: "Python")

manager1.display_details()
print()
developer1.display_details()
```

```
"C:\Users\FAR00Q HASSAN\AppData\Local
Name: Bob
Salary: 80000
Department: Engineering

Name: Alice
Salary: 70000
Programming Language: Python

Process finished with exit code 0
```

5. Create a base class Shape with a method area().

Create the following subclasses:

- **Circle** (takes radius and implements area() as $\pi * r^2$)
- **Rectangle** (takes length and width and implements area() as $\text{length} \times \text{width}$)
- **Triangle** (takes base and height and implements area() as $0.5 \times \text{base} \times \text{height}$)

Use polymorphism to calculate the area of different shapes.

```
class Circle: 1usage
    def __init__(self, radius):
        self.radius = radius

    def area(self): 1usage
    return 3.14 * self.radius ** 2

class Rectangle: 1usage
    def __init__(self, length, width):
        self.length = length
        self.width = width

    def area(self): 1usage
    return self.length * self.width

class Triangle: 1usage
    def __init__(self, base, height):
        self.base = base
        self.height = height

    def area(self): 1usage
    return 0.5 * self.base * self.height

print("Area of Circle:", Circle(5).area())
print("Area of Rectangle:", Rectangle( length: 3, width: 4).area())
print("Area of Triangle:", Triangle( base: 6, height: 2).area())
```

```
"C:\Users\FAR00Q HASSAN\AppData
Area of Circle: 78.5
Area of Rectangle: 12
Area of Triangle: 6.0
```