**Name: Usman Ul haq**

**Roll number: SU92-BSAIM-S24-035**

**Section: 2A**

**Semester: 2nd**

**Task = 07 (1 & 2)**

**Task 1 Create a Python program to model a vehicle rental system:**

1. Implement a parent class called Vehicle with attributes make and model. Include a

method display\_info to display the make and model of the vehicle.

2. Create a child class Car inheriting from Vehicle. The Car class should have an

additional attribute num\_doors and a method additional\_info to display the number

of doors.

3. Create another child class LuxuryCar inheriting from Car. The LuxuryCar class

should have an additional attribute features and a method additional\_info to display

the luxury features.

class Transportation:

def \_\_init\_\_(self, make, model):

self.make = make

self.model = model

def display\_info(self):

print(f"Make: {self.make}")

print(f"Model: {self.model}")

class Sedan(Transportation):

def \_\_init\_\_(self, make, model, num\_doors):

super().\_\_init\_\_(make, model)

self.num\_doors = num\_doors

def additional\_info(self):

print(f"Number of Doors: {self.num\_doors}")

class PremiumSedan(Sedan):

def \_\_init\_\_(self, make, model, num\_doors, features):

super().\_\_init\_\_(make, model, num\_doors)

self.features = features

def additional\_info(self):

super().additional\_info()

print(f"Luxury Features: {', '.join(self.features)}")

transportation = Transportation("Toyota", "Corolla")

transportation.display\_info()

sedan = Sedan("Honda", "Civic", 4)

sedan.display\_info()

sedan.additional\_info()

premium\_sedan = PremiumSedan("BMW", "7 Series", 4, ["Leather Seats", "Sunroof", "Advanced Navigation"])

premium\_sedan.display\_info()

premium\_sedan.additional\_info()

**Task 2;** **Create a Python program to model a company's employee hierarchy:**

1. Implement a parent class called Employee with attributes name and position.

Include a method display\_info to display the employee's name and position.

2. Create two child classes Manager and Worker inheriting from Employee. Each

should have an additional attribute (department for Manager and hours\_worked for

Worker) and a method additional\_info to display the additional attributes.

class Staff:

    def \_\_init\_\_(self, name, position):

        self.name = name

        self.position = position

    def display\_info(self):

        print(f"Name: {self.name}")

        print(f"Position: {self.position}")

class TeamLead(Staff):

    def \_\_init\_\_(self, name, position, department):

        super().\_\_init\_\_(name, position)

        self.department = department

    def additional\_info(self):

        print(f"Department: {self.department}")

class Laborer(Staff):

    def \_\_init\_\_(self, name, position, hours\_worked):

        super().\_\_init\_\_(name, position)

        self.hours\_worked = hours\_worked

    def additional\_info(self):

        print(f"Hours Worked: {self.hours\_worked}")

staff = Staff("Alice", "General Staff")

staff.display\_info()

team\_lead = TeamLead("Bob", "Team Lead", "Sales")

team\_lead.display\_info()

team\_lead.additional\_info()

laborer = Laborer("Charlie", "Laborer", 40)

laborer.display\_info()

laborer.additional\_info()