Name: Fatima Bint e Naseer

Roll number: SU92-BSAIM-S24-050

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 Vehicle:

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 CompactCar(Vehicle):

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 LuxuryCompactCar(CompactCar):

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)}")

**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 Personnel:

def \_\_init\_\_(self, employee\_name, job\_title):

self.employee\_name = employee\_name

self.job\_title = job\_title

def show\_info(self):

print(f"Employee Name: {self.employee\_name}")

print(f"Job Title: {self.job\_title}")

class TeamLeader(Personnel):

def \_\_init\_\_(self, employee\_name, job\_title, division):

super().\_\_init\_\_(employee\_name, job\_title)

self.division = division

def show\_additional\_info(self):

print(f"Division: {self.division}")

class Worker(Personnel):

def \_\_init\_\_(self, employee\_name, job\_title, hours\_logged):

super().\_\_init\_\_(employee\_name, job\_title)

self.hours\_logged = hours\_logged

def show\_additional\_info(self):

print(f"Hours Logged: {self.hours\_logged}")

personnel = Personnel("Alice", "General Staff")

personnel.show\_info()

team\_leader = TeamLeader("Bob", "Team Leader", "Sales")

team\_leader.show\_info()

team\_leader.show\_additional\_info()

worker = Worker("Charlie", "Laborer", 40)

worker.show\_info()

worker.show\_additional\_info()