

Object Oriented Programming

Lab 7

Task 1:

- ◆ Creating a Python program to model a vehicle rental system:

```
class Vehicle:
    def __init__(self, make, model):
        self.make = make
        self.model = model

    def display_info(self):
        print(f"Make: {self.make}, Model: {self.model}")

class Car(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 LuxuryCar(Car):
    def __init__(self, make, model, num_doors, features):
        super().__init__(make, model, num_doors)
        self.features = features

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

if __name__ == "__main__":
    vehicle = Vehicle("Toyota", "Corolla")
    vehicle.display_info()

    car = Car("Honda", "Civic", 4)
    car.display_info()
    car.additional_info()

    luxury_car = LuxuryCar("Mercedes", "S-Class", 4, ["Leather seats", "Sunroof", "Premium sound system"])
    luxury_car.display_info()
    luxury_car.additional_info()
```

```
Make: Toyota, Model: Corolla
Make: Honda, Model: Civic
Number of doors: 4
Make: Mercedes, Model: S-Class
Luxury Features: Leather seats, Sunroof, Premium sound system
```

Task 2:

- ◆ Creating a Python program to model a company's employee hierarchy:

```
class Employee:
    def __init__(self, name, position):
        self.name = name
        self.position = position

    def display_info(self):
        print(f"Name: {self.name}, Position: {self.position}")

class Manager(Employee):
    def __init__(self, name, position, department):
        super().__init__(name, position)
        self.department = department

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

class Worker(Employee):
    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}")

if __name__ == "__main__":
    employee = Employee("Ayan", "Founder")
    employee.display_info()

    manager = Manager("Ali", "Sales Manager", "Sales")
    manager.display_info()
    manager.additional_info()

    worker = Worker("Abdullah", "Developer", 20)
    worker.display_info()
    worker.additional_info()
```

```
Name: Ayan, Position: Founder
Name: Ali, Position: Sales Manager
Department: Sales
Name: Abdullah, Position: Developer
Hours Worked: 20
```

