

SE 1108 PROGRAMMING AND PROBLEM SOLVING II

Task: Car Park Management System

A car park needs an efficient way to manage various types of vehicles, each requiring different amounts of space and incurring different parking fees based on size and parking duration. Your task is to create a system that can add vehicles to the car park, ensuring that it does not exceed its capacity and calculates parking fees dynamically.

Requirements:

1. Abstract Class – Vehicle:

- Define the **Vehicle** class with the following attributes:
 - licensePlate (String)
 - vehicleSize (VehicleSize)
- Create a constructor to initialize licensePlate attribute.
- Getters for attributes
- Abstract Method: **calculateParkingFee** – that takes a parking duration parameter and returns the fee based on vehicle size and parking duration.

2. Enum - VehicleSize:

- Constants: SMALL, MEDIUM, LARGE.
- SMALL vehicles occupy 1 space and pays \$1 per hour.
- MEDIUM vehicles occupy 2 spaces and pays \$2 per hour.
- LARGE vehicles occupy 3 spaces and pays \$3 per hour.
- You might consider adding methods to the Enum if necessary.

3. Vehicle Subclasses:

- Define specific vehicle classes (Motorcycle, Car, Truck) that extend the Vehicle class.
- Each class should have a constructor that specifies the vehicleSize.
- A toString method that uses all attributes.

4. CarPark Class:

- Define the **CarPark** class with the following attributes:
 - parkedVehicles (ArrayList<Vehicle>): To hold parked vehicles.
 - capacity (int): Maximum number of spaces the car park can accommodate.
 - totalEarnings (int): Total earnings from parked vehicles
- Create a constructor to initialize attributes.
- Implement **parkVehicle** method, should check for available space that fit the vehicle size and park the vehicle accordingly.
- The **releaseVehicle** method handles vehicle departure and fee processing according to the given parking duration.
- A toString method