**Mini Project Assignment: Vehicle Rental System**

**Problem Statement:**

Design and implement a **Vehicle Rental System** that allows customers to rent different types of vehicles (cars, bikes, trucks) for specific durations. The system should manage vehicle inventory, customer information, rental records, and generate rental bills.

**Features to Implement:**

**1. Classes and Attributes**

* **Vehicle** (Base Class)
  + Vehicle ID
  + Brand
  + Model
  + Type (Car, Bike, Truck)
  + Rent per day
  + Availability status
* **Car / Bike / Truck** (Derived Classes)
  + Additional attributes like number of seats (Car), engine capacity (Bike), load capacity (Truck)
* **Customer**
  + Customer ID
  + Name
  + Contact information
* **Rental**
  + Rental ID
  + Customer (object)
  + Vehicle (object)
  + Rental start date
  + Rental duration (in days)
  + Total rent
* **RentalSystem**
  + Manages all vehicle rentals, returns, and inventory status

**Functional Requirements**

* Add vehicles to the system
* Display available vehicles by type
* Register new customers
* Rent a vehicle to a customer
* Return a vehicle and mark it available
* Calculate total rent (including optional insurance)
* Generate rental receipt/report

**OOP Concepts to Use**

|  |  |
| --- | --- |
| **Concept** | **Where It's Applied** |
| **Encapsulation** | Use private attributes and public accessors |
| **Inheritance** | Derive Car, Bike, Truck from Vehicle class |
| **Polymorphism** | Override display methods for vehicle types |
| **Abstraction** | Optional: Abstract base class Vehicle |
| **Association** | A Rental has a Customer and a Vehicle |

**Optional Enhancements**

* Use file handling for storing customer or rental data
* Calculate late fees if vehicle is returned after due date
* Include insurance charges per day
* Categorize vehicles by fuel type or transmission
* Implement a menu-driven console interface

**Sample Console Menu**

--- Vehicle Rental System ---

1. Add Vehicle

2. View Available Vehicles

3. Register Customer

4. Rent a Vehicle

5. Return a Vehicle

6. View Rental History

7. Exit

**Deliverables**

* Source code (Java or C++)
* Sample run/output
* Class-wise explanation of code
* Report or readme file