**C# Inheritance (Real-Life Scenario: Vehicle Management System)**

**Scenario:**

A **vehicle management system** needs to manage different types of vehicles like **Cars**, **Bikes**, and **Trucks**. All vehicles have common properties like **Brand**, **Model**, and **Speed**, but different types of vehicles have specific features.

The goal of this assessment is to build a hierarchy using **inheritance** to manage these vehicle types.

**Part 1: Base Class and Derived Classes**

**Task: Create a Base Class Vehicle and Derived Classes Car, Bike, and Truck.**

using System;

namespace VehicleManagementSystem

{

// Base class

public class Vehicle

{

public string Brand { get; set; }

public string Model { get; set; }

public int Speed { get; set; }

public Vehicle(string brand, string model, int speed)

{

Brand = brand;

Model = model;

Speed = speed;

}

public virtual void DisplayInfo()

{

Console.WriteLine($"Brand: {Brand}, Model: {Model}, Speed: {Speed} km/h");

}

}

// Derived class: Car

public class Car : Vehicle

{

public int NumberOfDoors { get; set; }

public Car(string brand, string model, int speed, int numberOfDoors)

: base(brand, model, speed)

{

NumberOfDoors = numberOfDoors;

}

public override void DisplayInfo()

{

base.DisplayInfo();

Console.WriteLine($"Number of Doors: {NumberOfDoors}");

}

}

// Derived class: Bike

public class Bike : Vehicle

{

public bool HasCarrier { get; set; }

public Bike(string brand, string model, int speed, bool hasCarrier)

: base(brand, model, speed)

{

HasCarrier = hasCarrier;

}

public override void DisplayInfo()

{

base.DisplayInfo();

Console.WriteLine($"Has Carrier: {HasCarrier}");

}

}

// Derived class: Truck

public class Truck : Vehicle

{

public int LoadCapacity { get; set; } // in tons

public Truck(string brand, string model, int speed, int loadCapacity)

: base(brand, model, speed)

{

LoadCapacity = loadCapacity;

}

public override void DisplayInfo()

{

base.DisplayInfo();

Console.WriteLine($"Load Capacity: {LoadCapacity} tons");

}

}

}

**Part 2: Creating Objects and Testing Inheritance**

**Task: Write a Main method to create objects of Car, Bike, and Truck and call the DisplayInfo() method for each.**

class Program

{

static void Main(string[] args)

{

// Creating Car object

Car car = new Car("Tesla", "Model S", 250, 4);

car.DisplayInfo();

Console.WriteLine();

// Creating Bike object

Bike bike = new Bike("Yamaha", "YZF R15", 150, true);

bike.DisplayInfo();

Console.WriteLine();

// Creating Truck object

Truck truck = new Truck("Volvo", "FH16", 120, 30);

truck.DisplayInfo();

}

}