Who is what?

The classes, interfaces, and objects in the above class diagram can be identified as follows:

1. **IVehicleBuilder** - Builder interface
2. **HeroBuilder & HondaBuilder**- Concrete Builder
3. **Vehicle**- Product
4. **Vehicle Creator**- Director

/// <summary>

/// The 'Builder' interface

/// </summary>

public interface IVehicleBuilder

{

void SetModel();

void SetEngine();

void SetTransmission();

void SetBody();

void SetAccessories();

Vehicle GetVehicle();

}

/// <summary>

/// The 'ConcreteBuilder1' class

/// </summary>

public class HeroBuilder : IVehicleBuilder

{

Vehicle objVehicle = new Vehicle();

public void SetModel()

{

objVehicle.Model = "Hero";

}

public void SetEngine()

{

objVehicle.Engine = "4 Stroke";

}

public void SetTransmission()

{

objVehicle.Transmission = "120 km/hr";

}

public void SetBody()

{

objVehicle.Body = "Plastic";

}

public void SetAccessories()

{

objVehicle.Accessories.Add("Seat Cover");

objVehicle.Accessories.Add("Rear Mirror");

}

public Vehicle GetVehicle()

{

return objVehicle;

}

}

/// <summary>

/// The 'ConcreteBuilder2' class

/// </summary>

public class HondaBuilder : IVehicleBuilder

{

Vehicle objVehicle = new Vehicle();

public void SetModel()

{

objVehicle.Model = "Honda";

}

public void SetEngine()

{

objVehicle.Engine = "4 Stroke";

}

public void SetTransmission()

{

objVehicle.Transmission = "125 Km/hr";

}

public void SetBody()

{

objVehicle.Body = "Plastic";

}

public void SetAccessories()

{

objVehicle.Accessories.Add("Seat Cover");

objVehicle.Accessories.Add("Rear Mirror");

objVehicle.Accessories.Add("Helmet");

}

public Vehicle GetVehicle()

{

return objVehicle;

}

}

/// <summary>

/// The 'Product' class

/// </summary>

public class Vehicle

{

public string Model { get; set; }

public string Engine { get; set; }

public string Transmission { get; set; }

public string Body { get; set; }

public List<string> Accessories { get; set; }

public Vehicle()

{

Accessories = new List<string>();

}

public void ShowInfo()

{

Console.WriteLine("Model: {0}", Model);

Console.WriteLine("Engine: {0}", Engine);

Console.WriteLine("Body: {0}", Body);

Console.WriteLine("Transmission: {0}", Transmission);

Console.WriteLine("Accessories:");

foreach (var accessory in Accessories)

{

Console.WriteLine("\t{0}", accessory);

}

}

}

/// <summary>

/// The 'Director' class

/// </summary>

public class VehicleCreator

{

private readonly IVehicleBuilder objBuilder;

public VehicleCreator(IVehicleBuilder builder)

{

objBuilder = builder;

}

public void CreateVehicle()

{

objBuilder.SetModel();

objBuilder.SetEngine();

objBuilder.SetBody();

objBuilder.SetTransmission();

objBuilder.SetAccessories();

}

public Vehicle GetVehicle()

{

return objBuilder.GetVehicle();

}

}

/// <summary>

/// Builder Design Pattern Demo

/// </summary>

class Program

{

static void Main(string[] args)

{

var vehicleCreator = new VehicleCreator(new HeroBuilder());

vehicleCreator.CreateVehicle();

var vehicle = vehicleCreator.GetVehicle();

vehicle.ShowInfo();

Console.WriteLine("---------------------------------------------");

vehicleCreator = new VehicleCreator(new HondaBuilder());

vehicleCreator.CreateVehicle();

vehicle = vehicleCreator.GetVehicle();

vehicle.ShowInfo();

Console.ReadKey();

}

}