



FACTORY METHOD

Design Pattern





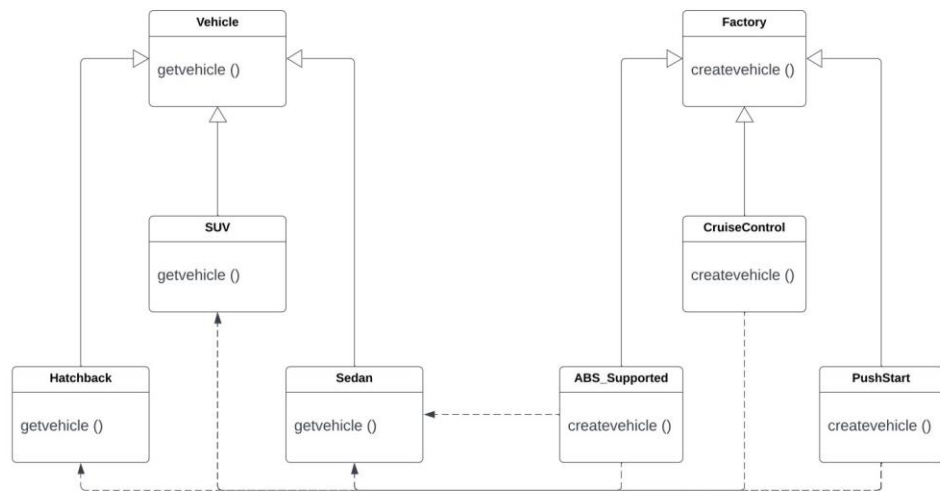
INTERNATIONAL ISLAMIC UNIVERSITY ISLAMABAD

Report

Name: Tamoor Ahmad
Registration No: 4125-FBAS/BSSE/F20
Course Code: SE322
Course Title: Software Design and Architecture

Submitted to: Dr. Muhammad Nasir

Class Diagram:



Consider the case of a vehicle manufacturing company that engineers' different types of vehicles, SUV, Sedan, Hatchback etc. The method (Factory method) to engineer each type of vehicle would be based on different features, ABS, Cruise Control, Engine Capacity, Hybrid, Echo Mode, Push-Start etc.

Source Code:

```
using System;
using System.Collections.Generic;

namespace factoryMethodAssignment
{
    class Program
    {
        static void Main(string[] args)
        {
```

```

        Factory[] Vehicles = new Factory[3];
        Vehicles[0] = new ABS_Supported();
        Vehicles[1] = new CruiseControl();
        Vehicles[2] = new PushStart();

        foreach (Factory vehicleFactory in Vehicles)
        {
            Console.WriteLine("\n" + vehicleFactory.GetType().Name + "--");
            foreach (Vehicle veh in vehicleFactory.vehicles)
            {
                Console.WriteLine(" " + veh.GetType().Name);
            }
        }
    }
}

abstract class Vehicle
{
}
class SUV : Vehicle
{
}
class Sedan : Vehicle
{
}
class hatchback : Vehicle
{
}
abstract class Factory
{
    private List<Vehicle> _vehicles = new List<Vehicle>();

    public Factory()
    {
        this.createVehicle();
    }

    public List<Vehicle> vehicles
    {
        get { return _vehicles; }
    }
    public abstract void createVehicle();
}
class ABS_Supported : Factory
{
    public override void createVehicle()
    {
        vehicles.Add(new SUV());
        vehicles.Add(new Sedan());
    }
}

class CruiseControl : Factory
{
    public override void createVehicle()
    {
        vehicles.Add(new Sedan());
        vehicles.Add(new hatchback());
    }
}

class PushStart : Factory

```

```
{  
    public override void createVehicle()  
    {  
        vehicles.Add(new hatchback());  
        vehicles.Add(new SUV());  
        vehicles.Add(new Sedan());  
    }  
}
```

Output:

Select C:\Users\ITS\source\repos\Factory Method\Factory Method\bin\Debug\Factory Method.exe

```
ABS_Supported--  
SUV  
Sedan  
  
CruiseControl--  
Sedan  
hatchback  
  
PushStart--  
hatchback  
SUV  
Sedan
```