Assignment No. 09

Name -: Parth Manoj Poriya

Roll No-: 34

Branch-: SE IT(B2)

Subject-: Object Oriented Programming

TITLE

Write a program to Implement Factory Design Pattern for the given context

INPUT

```
abstract class Car {
   public abstract void allocateAccessories();
   public abstract void assemble();
   public abstract void paint();
   public abstract void test();

   public final void buildCar() {
      allocateAccessories();
      assemble();
      paint();
      test();
   }
```

```
}
class Hatchback extends Car {
   public void allocateAccessories() {
      System.out.println("Allocating accessories for Hatchback.");
   }

   public void assemble() {
      System.out.println("Assembling Hatchback.");
   }

   public void paint() {
      System.out.println("Painting Hatchback.");
   }

   public void test() {
      System.out.println("Testing Hatchback.");
   }
}
class Sedan extends Car {
   public void allocateAccessories() {
      System.out.println("Allocating accessories for Sedan.");
   }

   @Override
   public void assemble() {
      System.out.println("Assembling Sedan.");
   }

   public void paint() {
      System.out.println("Painting Sedan.");
   }

   public void test() {
      System.out.println("Testing Sedan.");
   }
```

```
}
class SUV extends Car {
    public void allocateAccessories() {
        System.out.println("Allocating accessories for SUV.");
    }

    public void assemble() {
        System.out.println("Assembling SUV.");
    }

    public void paint() {
        System.out.println("Painting SUV.");
    }

    public void test() {
        System.out.println("Testing SUV.");
    }
}
class CarFactory {
    public static Car createCar(String type) {
        switch (type.toLowerCase()) {
            case "hatchback":
                return new Hatchback();
            case "sedan":
                return new Sedan();
            case "suv":
                return new SUV();
            default:
                throw new IllegalArgumentException("Unknown car type: "
+ type);
        }
    }
}
public class Assign9 {
    public static void main(String[] args) {
        Car hatchback = CarFactory.createCar("hatchback");
        hatchback.buildCar();

        System.out.println();

        Car sedan = CarFactory.createCar("sedan");
        sedan.buildCar();

        System.out.println();

        Car suv = CarFactory.createCar("suv");
        suv.buildCar();
    }
}
```

OUTPUT

```
PS C:\Users\Acer\OneDrive\Desktop\Java> javac Assign9.java
PS C:\Users\Acer\OneDrive\Desktop\Java> java Assign9
Allocating accessories for Hatchback.
Assembling Hatchback.
Painting Hatchback.
Testing Hatchback.

Allocating accessories for Sedan.
Assembling Sedan.
Painting Sedan.
Testing Sedan.

Allocating accessories for SUV.
Assembling SUV.
Painting SUV.
Testing SUV.
```

-------------------------------X----------------------X--------------------------X-----------------------