# Assignment . 09

**Aim:** Design and implement Factory design pattern for the given context. Consider Car building process, which requires many steps from allocating accessories to final makeup. These steps should be written as methods and should be called while creating an instance of a specific car type. Hatchback, Sedan, SUV could be the subclasses of Car class. Car class and its subclasses, CarFactory and TestFactoryPattern should be implemented.

**Objectives:** To learn the concept of Design pattern

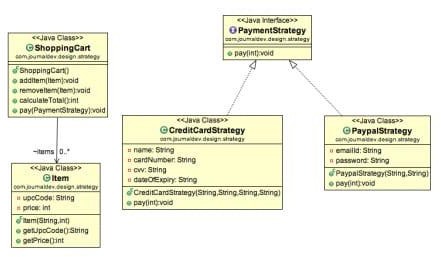
## Theory:

1. Design pattern
2. Factory design pattern diagram with example
3. Advantages of factory design pattern
4. Usage and the application where factory design patterns can be applied .

## Sample Code:

* Draw the class diagram for given context
* Crate classes such as **Car.java ,CarFactory.java ,**CarType .java
* LuxuryCar.java ,SedanCar.java ,SmallCar.java ,TestFactorypatern.java

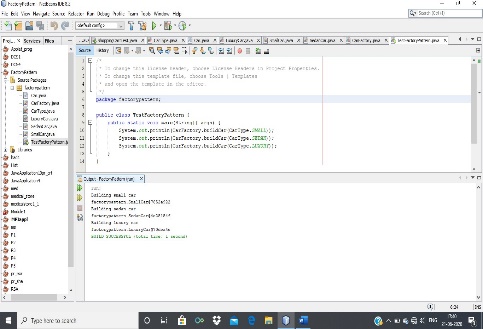
**Input:** Design and implement Factory design pattern for the given context. Consider Car building process, which requires many steps from allocating accessories to final makeup. These steps should be written as methods and should be called while creating an instance of a specific car type. Hatchback, Sedan, SUV could be the subclasses of Car class. Car class and its subclasses, CarFactory and TestFactoryPattern should be implemented.



**Output :**. Factory pattern –

Building small car factorypattern.SmallCar@7852e922 Building sedan car factorypattern.SedanCar@4e25154f Building luxury car factorypattern.LuxuryCar@70dea4e

BUILD SUCCESSFUL (total time: 1 second)



## Frequently Ask Question:

1. Give the applications where design patterns can be applied ?
2. Why factory pattern?
3. Explain factory pattern?
4. Draw the design Pattern with a context.
5. Give examples of creational design pattern .
6. Design application by applying the factory pattern .
7. Represent and implement a *Shape* interface which implements Circle ,Square, rectangle using Fatory pattern

Represent and implement for bill generation using GetPlanFactory to get a Plan object. Pass information (Domestic / commercial/ institutional) to get the type of object it needs.

ALGORTIHM:

1 Create an abstract class car factory

2 We have taken input and display function to take the budget and to show the car features .

3 Then we have made functions for hatch back ,luxury, and sedan car versions according to the budget

4 In the main class we have called all the functions.

5 ENDS

CODE:

import java.util.Scanner;

abstract class Car\_Factory{

String company,cn, col;

double budget;

abstract void getprice(double price);

abstract void detail(String company\_nane, String car\_name, String co);

abstract void access ();

void input (){

Scanner sc =new Scanner (System.in);

System.out.print("Company Name : ");

company = sc.next();

System.out.print("Car: ");

cn=sc.next();

System.out.print("Budget Planned (in lakhs): ");

budget = sc.nextDouble();

System.out.print ("Enter Color of Car : ");

col=sc.next();

}

void display(Car\_Factory c2){

c2.getprice(budget);

c2.detail(company,cn,col);

c2.access();

}

}

class Hatchback extends Car\_Factory{

String rep;

public void getprice(double price){

if(price<2)

System.out.println("your buget is insufficient, plz increase your Budget");

else if(price>2&&price<5)

rep = "No";

else

rep = "Yes";

}

public void detail(String company\_name, String car\_name, String Color) {

System.out.println("\n\*\*\*\*\*\*\*\*");

System.out.println("Company: "+company\_name);

System.out.println("Name of Car : "+car\_name);

System.out.println("Color : "+ Color);

System.out.println("Fule Type : Petrol");

System.out.println("Gears : Manual");

}

public void access(){

System.out.println("Type of wheels : Alloy Wheels");

System.out.println("Airbags : "+rep);

System.out.println("Back Wiper : "+rep);

System.out.println("Side Mirror : Two");

System.out.println("Touch Screen Music Player : "+rep);

}

}

class Sedan extends Car\_Factory{

String rep;

public void getprice(double price){

if(price<2)

System.out.println("your buget is insufficient, plz increase your Budget");

else if(price>6&&price<10)

rep = "No";

else

rep = "Yes";

}

public void detail(String company\_name, String car\_name, String Col) {

System.out.println("\n\*\*\*\*\*\*\*\*");

System.out.println("Company: "+company\_name);

System.out.println("Name of Car : "+car\_name);

System.out.println("Color : "+ Col);

System.out.println("Fule Type : Petrol/Diesel");

System.out.println("Gears :Automatic/ Manual");

}

public void access(){

System.out.println("Type of wheels : Alloy Wheels");

System.out.println("Airbags : YES");

System.out.println("Back Wiper : YES");

System.out.println("Side Mirror : Two");

System.out.println("Touch Screen Music Player : YES");

System.out.println("Roof Window : "+rep);

}

}

class Luxuray extends Car\_Factory{

String rep;

public void getprice(double price){

if(price<2)

System.out.println("your Budget is insufficient, pla increase your Budget");

else if(price>10&&price<14)

rep = "No";

else

rep = "Yes";

}

public void detail(String company\_name, String car\_name, String col) {

System.out.println("\n\*\*\*\*\*\*\*\*");

System.out.println("Company: "+company\_name);

System.out.println("Name of Car : "+car\_name);

System.out.println("Color : "+ col);

System.out.println("Fule Type : Diesel");

System.out.println("Gears : Automatic");

}

public void access(){

System.out.println("Type of wheels : Alloy Wheels");

System.out.println("Airbags : YES");

System.out.println("Back Wiper : YES");

System.out.println("Side Mirror : Two");

System.out.println("Touch Screen Music Player : YES");

System.out.println("Panoramic Sun-Roof : YES");

System.out.println("Automative Garbage Cans : "+rep);

System.out.println("Automotive Air Freshenar : "+rep);

System.out.println("Button Start : "+rep);

}

}

public class Main{

public static void main(String[] args){

Scanner sc1 = new Scanner(System.in);

int ch;

Car\_Factory c1;

do{

System.out.println("\n1. Hatchback \n2.Sedan Car \n3.Luxury Car \n4. EXIT");

System.out.println("Enter Your Choice : ");

ch = sc1.nextInt();

System.out.println();

switch(ch){

case 1:

c1 = new Hatchback();

c1.input();

c1.display(c1);

break;

case 2:

c1 = new Sedan();

c1.input();

c1.display(c1);

break;

case 3:

c1 = new Luxuray();

c1.input();

c1.display(c1);

break;

case 4:

System.out.println("\nThank You ");

break;

default:

System.out.println("INVALID CHOICE !!");

break;

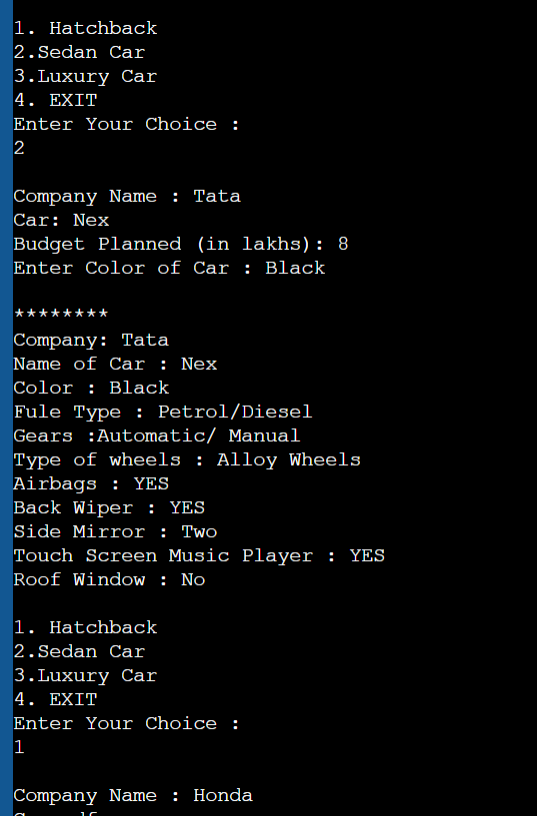
}

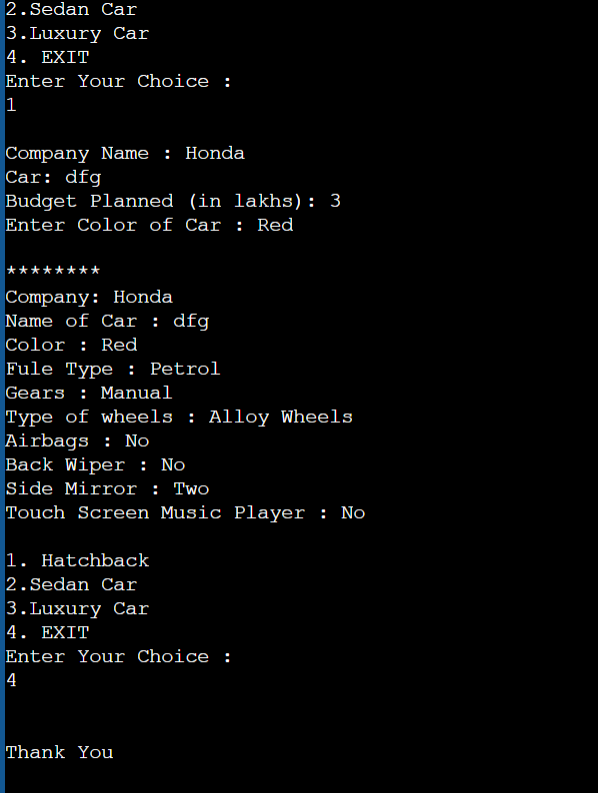
}while(ch!=4);

}

}

OUTPUT:





CONCLUSION

IN THIS ASSIGNMENT WE HAVE SUCCESSFULLY LEARNED THE CONCEPT OF FACTORY DESIGN IN JAVA.