//PROBLEM STATEMENT :

/* Implement a 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 specific car type.

* Hatchback, Sedan, SUV , could be the subclasses Car class. Car class and Car class its subclasses

* , CarFactory and Test Factory Pattern should be implemented */

```
package assignment;

import java.util.Scanner;

// ============================== ABSTRACT CLASS Car_Factory
==============================//

abstract class Car_Factory{

	//declaration of data member

	String compnay,car_name;

	double budget;


	//declaration of abstract methods

	abstract void getprice(double price);

	abstract void detail(String company_name,String car_name);

	abstract void accessories();


	//declaration and implentation of input method

	void input() {

		Scanner scan =new Scanner (System.in);//creating object of scanner class
```

```
                System.out.print("Company- ");
                 compnay=scan.next();//taking input from user
                System.out.print("Car- ");
                 car_name=scan.next();//taking input from user
                System.out.print("Rough Budget(in Lakhs)- ");
                 budget=scan.nextDouble();//taking input from user
        }
        void display(Car_Factory obj1) {
                //calling the methods//
                obj1.getprice(budget);//calling getprice method
                System.out.println("\n-----------------------------------");
                obj1.detail(compnay, car_name);//calling detail method
                System.out.println("\n-----------------------------------");
                obj1.accessories();//calling accessories method
                System.out.println("\n-----------------------------------");
        }

}

//============================  CLASS Small_car ============================//
class Small_car extends Car_Factory{
        String Ans;//declaration of data member

        //method for getprice
        public void getprice(double price) {
```

```
            if(price>2&&price<5)
                Ans="No";       //modify Ans
            else
                Ans="Yes";      //modify Ans

        }
        //method for displaying car detail//
        public void detail(String company_name,String car_name) {
            System.out.println("Company- "+company_name);
            System.out.println("Name of Car- "+car_name);
            System.out.println("Color- Black/White/Orange/Red");
            System.out.println("Fuel- Petrol");
            System.out.println("Gears- Manual");
        }

        //method to display accessories of car//
        public void accessories() {
            System.out.println("Types of Tyres- Alloy Wheels");
            System.out.println("Airbags- "+Ans);
            System.out.println("Back Wiper- "+Ans);
            System.out.println("Side Mirror- Two");
            System.out.println("Touch Screen Music Player- "+Ans);
        }
}
```

```
//============================  CLASS Sedan ============================//
class Sedan extends Car_Factory{
	String Ans;//declaration of data member

	//method for getprice
	public void getprice(double price) {
		if(price>6&&price<10)
			Ans="No";	//modify Ans
		else
			Ans="Yes";	//modify Ans
	}

	//method for displaying car detail//
	public void detail(String company_name,String car_name) {
		System.out.println("Company- "+company_name);
		System.out.println("Name of Car- "+car_name);
		System.out.println("Color- Black/White/Orange/Red");
		System.out.println("Fuel- Petrol/Diesel");
		System.out.println("Gears- Auto/Manual");
	}
	//method to display accessories of car//
	public void accessories() {
		System.out.println("Types of Tyres- 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- "+Ans);
	}
}

//============================  CLASS Small_car ============================//
class Luxary extends Car_Factory{
	String Ans;//declaration of data member

	//method for getprice
	public void getprice(double price) {
		if(price>10&&price<14)
			Ans="No";	//modify Ans
		else
			Ans="Yes";	//modify Ans
	}

	//method for displaying car detail//
	public void detail(String company_name,String car_name) {
		System.out.println("Company- "+company_name);
		System.out.println("Name of Car- "+car_name);
		System.out.println("Color- Black/White/Orange/Red");
		System.out.println("Fuel- Diesel");
		System.out.println("Gears- Auto");
```

```
	}

	//method to display accessories of car//
	public void accessories() {
		System.out.println("Types of Tyres- 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- YES");
		System.out.println("Automotive Garbage Cans- "+Ans);
		System.out.println("Automotice Air Freshner- "+Ans);
		System.out.println("Button Start- "+Ans);
	}
}

//============================== MAIN CLASS ==============================//
public class Main {

	//ststic main method
	public static void main(String[] args) {
		// TODO Auto-generated method stub
		Scanner scan = new Scanner(System.in);//creating object of scanner class
		int ch;
		//double price;
```

```
Car_Factory obj;// object of reference Car_Factory
while(true){
        //menu driven
        System.out.println("Which Car you want to See?- ");
        System.out.println("\n\t1.Small Car\n\t2.Sedan Car\n\t3.Luxary Car\n\t4.Exit");
        ch=scan.nextInt();//taking input from user
        System.out.println();
        //switch case
        switch(ch) {

                case 1:
                        obj= new Small_car(); //creating object of Small_car
                        obj.input();//calling input method
                        obj.display(obj);//calling display method
                        break;

                case 2:
                        obj= new Sedan();//creating object of Sedan
                        obj.input();//calling input method
                        obj.display(obj);//calling display method
                        break;

                case 3:
                        obj= new Luxary();//creating object of Luxary
                        obj.input();//calling input method
```

```
                    obj.display(obj);//calling display method
                    break;

                case 4:
                    System.out.println("\n-----------------------------------");
                    return;//stop execution of program

                default:
                    System.out.println("INVALID CHOICE !!");//default
                    System.out.println("\n-----------------------------------");
                    break;

                }
            }
        }
}

/*

 ##OUTPUT##



---------------------------------------
 Constructing Hatchback Car
---------------------------------------
```

Types of Tyres- Alloy Wheels

Airbags- YES

Back Wiper- YES

Side Mirror- one

Touch Screen Music Player- NO

Roof Window- YES

Automotive Garbage Cans- YES

Automotice Air Freshner- NO

Button Start- YES

---

assignment.hatchback1@17a7cec2

Constructing sedan car

Types of Tyres- Alloy Wheels

Airbags- YES

Back Wiper- NO

Side Mirror- ONE

Touch Screen Music Player- YES

Roof Window- YES

Automotive Garbage Cans- YES

Automotice Air Freshner- NO

Button Start- YES

---

assignment.sedan2@6f539caf

Constructing SUV Car

Types of Tyres- Alloy Wheels

Airbags- YES

Back Wiper- YES

Side Mirror- Two

Touch Screen Music Player- YES

Roof Window- YES

Automotive Garbage Cans- NO

Automotice Air Freshner- YES

Button Start- YES

---

assignment.suv@50040f0c

*/