```
import java.util.Scanner;


abstract class Car {
    public abstract void assemble();
    public abstract void addAccessories();
    public abstract void finalMakeup();
}


class Hatchback extends Car {
    @Override
    public void assemble() {
        System.out.println("Assembling Hatchback...");
    }

    @Override
    public void addAccessories() {
        System.out.println("Adding Hatchback accessories...");
    }

    @Override
    public void finalMakeup() {
        System.out.println("Applying final makeup for Hatchback...");
    }
}

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

    @Override
    public void addAccessories() {
        System.out.println("Adding Sedan accessories...");
    }

    @Override
    public void finalMakeup() {
        System.out.println("Applying final makeup for Sedan...");
    }
}

class SUV extends Car {
    @Override
    public void assemble() {
        System.out.println("Assembling SUV...");
    }

    @Override
    public void addAccessories() {
        System.out.println("Adding SUV accessories...");
    }
```

```
    @Override
    public void finalMakeup() {
        System.out.println("Applying final makeup for SUV...");
    }
}


class CarFactory {
    public static Car getCar(String carType) {
        if (carType == null) {
            return null;
        }
        switch (carType.toLowerCase()) {
            case "hatchback":
                return new Hatchback();
            case "sedan":
                return new Sedan();
            case "suv":
                return new SUV();
            default:
                System.out.println("Invalid car type. Please select
Hatchback, Sedan, or SUV.");
                return null;
        }
    }
}


public class TestFactoryPattern {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter the type of car to build (Hatchback,
Sedan, SUV): ");
        String carType = sc.nextLine();

        Car car = CarFactory.getCar(carType);
        if (car != null) {
            car.assemble();
            car.addAccessories();
            car.finalMakeup();
        }

        sc.close();
    }
}
```