

**Creative Assignment Of Design**  
**Pattern On**  
**“Decorator Design Pattern”**

**Submitted By**

Yash Mishra-CS20DA68

**Under the Guidance  
of**

Ms. Titiksha Bhagat



**Department of Computer Science &  
Engineering**

**S. B. Jain Institute of Technology  
Management  
And  
Research, Nagpur-44150**

**2020-2021**

## **Car.java**

```
public interface Car {  
    public void assemble();  
}
```

## **BasicCar.java**

```
public class BasicCar implements Car {  
  
    public void assemble()  
    { System.out.print("Basic  
        Car-");  
    }  
  
}
```

## **CarDecorator.java**

```
public class CarDecorator implements  
Car { protected Car car;  
  
    public CarDecorator(Car  
c){ this.car=c; }  
  
    public void assemble()  
    { this.car.assemble();  
    }
```

```
}
```

## **SportsCar.java**

```
public class SportsCar extends CarDecorator {  
  
    public SportsCar(Car c)  
    { super(c); }  
  
    public void assemble()  
    { super.assemble();  
        System.out.print(" Adding features of Sports Car.");  
    }  
}
```

## **LuxuryCar.java**

```
public class LuxuryCar extends CarDecorator {  
  
    public LuxuryCar(Car c)  
    { super(c); }  
  
    public void assemble()  
    { super.assemble();  
        System.out.print(" Adding features of Luxury Car.");  
    }  
}
```

## **DecoratorPatternTest.java**

```
public class DecoratorPatternTest { public

    static void main(String[] args) {

        System.out.println("This is Decorator Design Pattern :" +
            "\n"); Car sportsCar = new SportsCar(new
        BasicCar()); sportsCar.assemble();
        System.out.println("\n");

        Car sportsLuxuryCar = new SportsCar(new LuxuryCar(new
BasicCar()));
        sportsLuxuryCar.assemble();
    }

}
```

## Output:

```
This is Decorator Design Pattern :

Basic Car- Adding features of Sports Car.

Basic Car- Adding features of Luxury Car. Adding features of Sports Car.
PS C:\Users\Asus\Downloads\Decorator design pattern by chaitanya> █
```