**NAME : RACHANA J N**

**Superset ID :   6416511**

**TASK 5 : Implementing the Decorator Pattern**

**Scenario:**

You are developing a notification system where notifications can be sent via multiple channels (e.g., Email, SMS). Use the Decorator Pattern to add functionalities dynamically.

**Steps:**

1. **Create a New Java Project:**
   * Create a new Java project named **DecoratorPatternExample**.
2. **Define Component Interface:**
   * Create an interface **Notifier** with a method **send()**.
3. **Implement Concrete Component:**
   * Create a class **EmailNotifier** that implements Notifier.
4. **Implement Decorator Classes:**
   * Create abstract decorator class **NotifierDecorator** that implements **Notifier** and holds a reference to a **Notifier** object.
   * Create concrete decorator classes like **SMSNotifierDecorator**, **SlackNotifierDecorator** that extend **NotifierDecorator**.
5. **Test the Decorator Implementation:**
   * Create a test class to demonstrate sending notifications via multiple channels using decorators.

**CODE :**

public class TestDecorator {

    public static void main(String[] args) {

        Notifier notifier = new EmailNotifier();

        Notifier smsNotifier = new SMSNotifier(notifier);

        Notifier allNotifier = new SlackNotifier(smsNotifier);

        allNotifier.send();

    }

}

interface Notifier {

    void send();

}

class EmailNotifier implements Notifier {

    public void send() {

        System.out.println("Sending Email Notification");

    }

}

abstract class NotifierDecorator implements Notifier {

    protected Notifier notifier;

    public NotifierDecorator(Notifier notifier) {

        this.notifier = notifier;

    }

    public void send() {

        notifier.send();

    }

}

class SMSNotifier extends NotifierDecorator {

    public SMSNotifier(Notifier notifier) {

        super(notifier);

    }

    public void send() {

        super.send();

        System.out.println("Sending SMS Notification");

    }

}

class SlackNotifier extends NotifierDecorator {

    public SlackNotifier(Notifier notifier) {

        super(notifier);

    }

    public void send() {

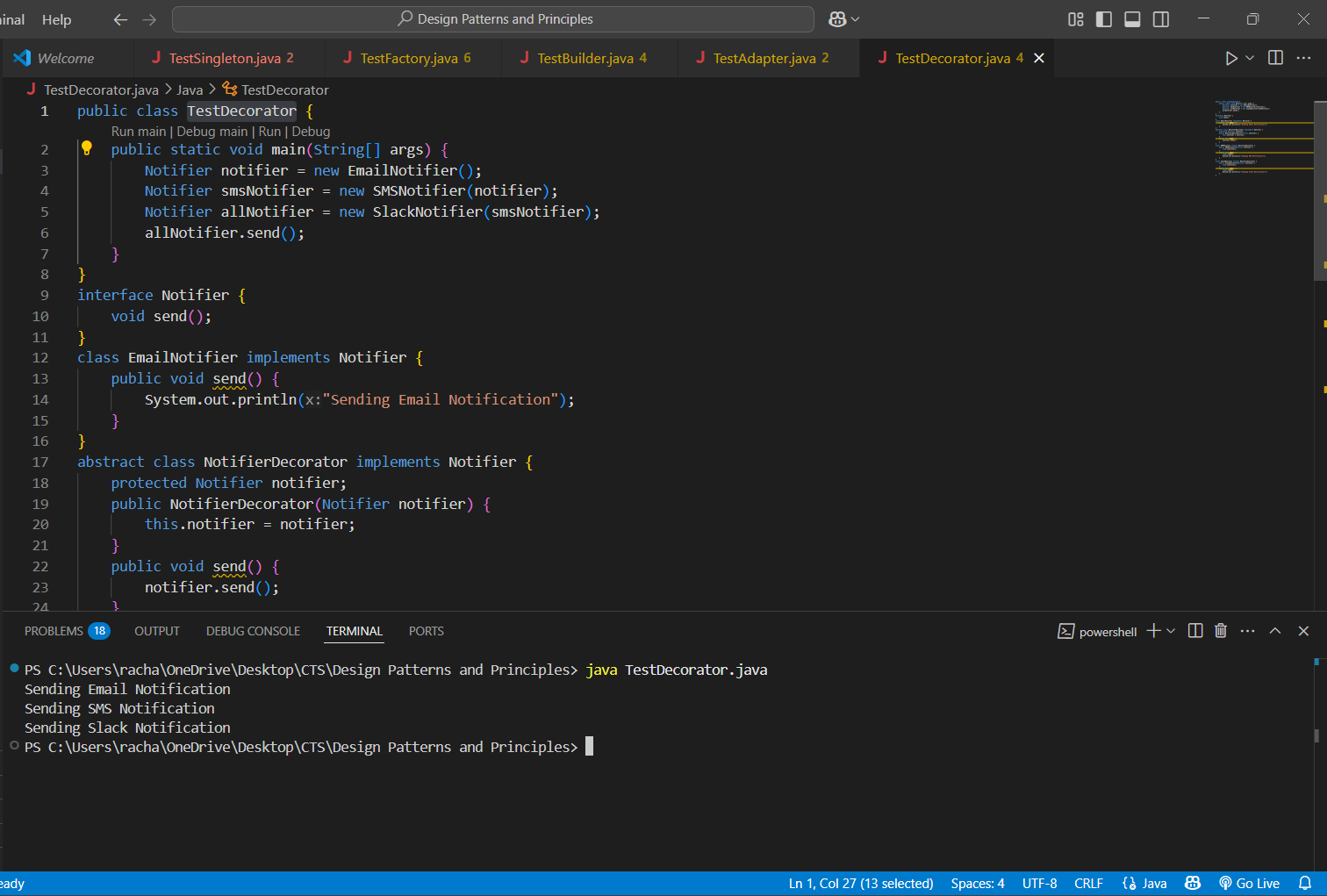
        super.send();

        System.out.println("Sending Slack Notification");

    }

}

**OUTPUT :**

****