**AdapterPatternExample**

**CODE :**

**PaymentProcessor.java**

public interface PaymentProcessor {

    void processPayment(double amount);

}

**PaymentTest.java**

public class PaymentTest {

    public static void main(String[] args) {

        PayPalAPI payPalAPI = new PayPalAPI();

        PaymentProcessor payPalProcessor = new PayPalAdapter(payPalAPI);

        payPalProcessor.processPayment(100.00);

StripeAPI stripeAPI = new StripeAPI();

        PaymentProcessor stripeProcessor = new StripeAdapter(stripeAPI);

        stripeProcessor.processPayment(200.00);

    }

}

**PayPalAPI.java**

public class PayPalAPI {

    public void sendPayment(double amountInDollars) {

        System.out.println("PayPal processed payment of $" + amountInDollars);

    }

}

**StripeAPI.java**

public class StripeAPI {

    public void makePayment(double amount) {

        System.out.println("Stripe processed payment of $" + amount);

    }

}

**PayPalAdapter.java**

public class PayPalAdapter implements PaymentProcessor {

    private PayPalAPI payPal;

public PayPalAdapter(PayPalAPI payPal) {

        this.payPal = payPal;

    }

@Override

    public void processPayment(double amount) {

        payPal.sendPayment(amount);

    }

}

**StripeAdapter.java**

public class StripeAdapter implements PaymentProcessor {

    private StripeAPI stripe;

public StripeAdapter(StripeAPI stripe) {

        this.stripe = stripe;

    }

@Override

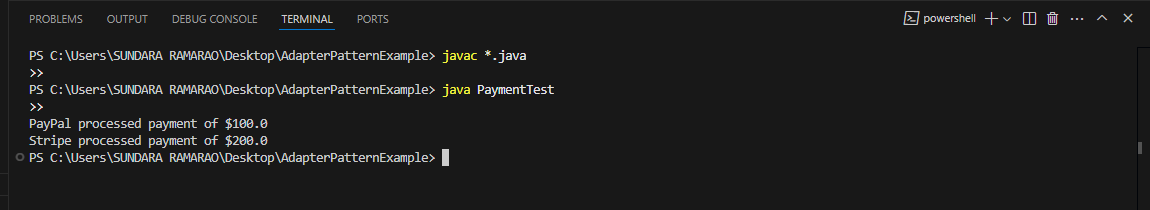
    public void processPayment(double amount) {

        stripe.makePayment(amount);

    }

}

**OUTPUT :**

****