**Exercise 8: Implementing the Strategy Pattern**

**Scenario:**

You are developing a payment system where different payment methods (e.g., Credit Card, PayPal) can be selected at runtime. Use the Strategy Pattern to achieve this.

**Steps:**

1. **Create a New Java Project:**
   * Create a new Java project named **StrategyPatternExample**.
2. **Define Strategy Interface:**
   * Create an interface PaymentStrategy with a method **pay()**.
3. **Implement Concrete Strategies:**
   * Create classes **CreditCardPayment**, **PayPalPayment** that implement **PaymentStrategy**.
4. **Implement Context Class:**
   * Create a class **PaymentContext** that holds a reference to **PaymentStrategy** and a method to execute the strategy.
5. **Test the Strategy Implementation:**
   * Create a test class to demonstrate selecting and using different payment strategies.

**CODE:-**

// Strategy Interface

interface PaymentStrategy {

    void pay(double amount);

}

// Concrete Strategy - Credit Card Payment

class CreditCardPayment implements PaymentStrategy {

    private String cardNumber;

    public CreditCardPayment(String cardNumber) {

        this.cardNumber = cardNumber;

    }

    @Override

    public void pay(double amount) {

        System.out.println("Paid $" + amount + " using Credit Card: " + cardNumber);

    }

}

// Concrete Strategy - PayPal Payment

class PayPalPayment implements PaymentStrategy {

    private String email;

    public PayPalPayment(String email) {

        this.email = email;

    }

    @Override

    public void pay(double amount) {

        System.out.println("Paid $" + amount + " using PayPal account: " + email);

    }

}

// Context Class

class PaymentContext {

    private PaymentStrategy paymentStrategy;

    public void setPaymentStrategy(PaymentStrategy paymentStrategy) {

        this.paymentStrategy = paymentStrategy;

    }

    public void pay(double amount) {

        if (paymentStrategy == null) {

            System.out.println("Payment method not set.");

        } else {

            paymentStrategy.pay(amount);

        }

    }

}

// Main Class to Test the Strategy Pattern

public class StrategyPatternExample {

    public static void main(String[] args) {

        PaymentContext context = new PaymentContext();

        // Pay using Credit Card

        context.setPaymentStrategy(new CreditCardPayment("1234-5678-9876-5432"));

        context.pay(250.00);

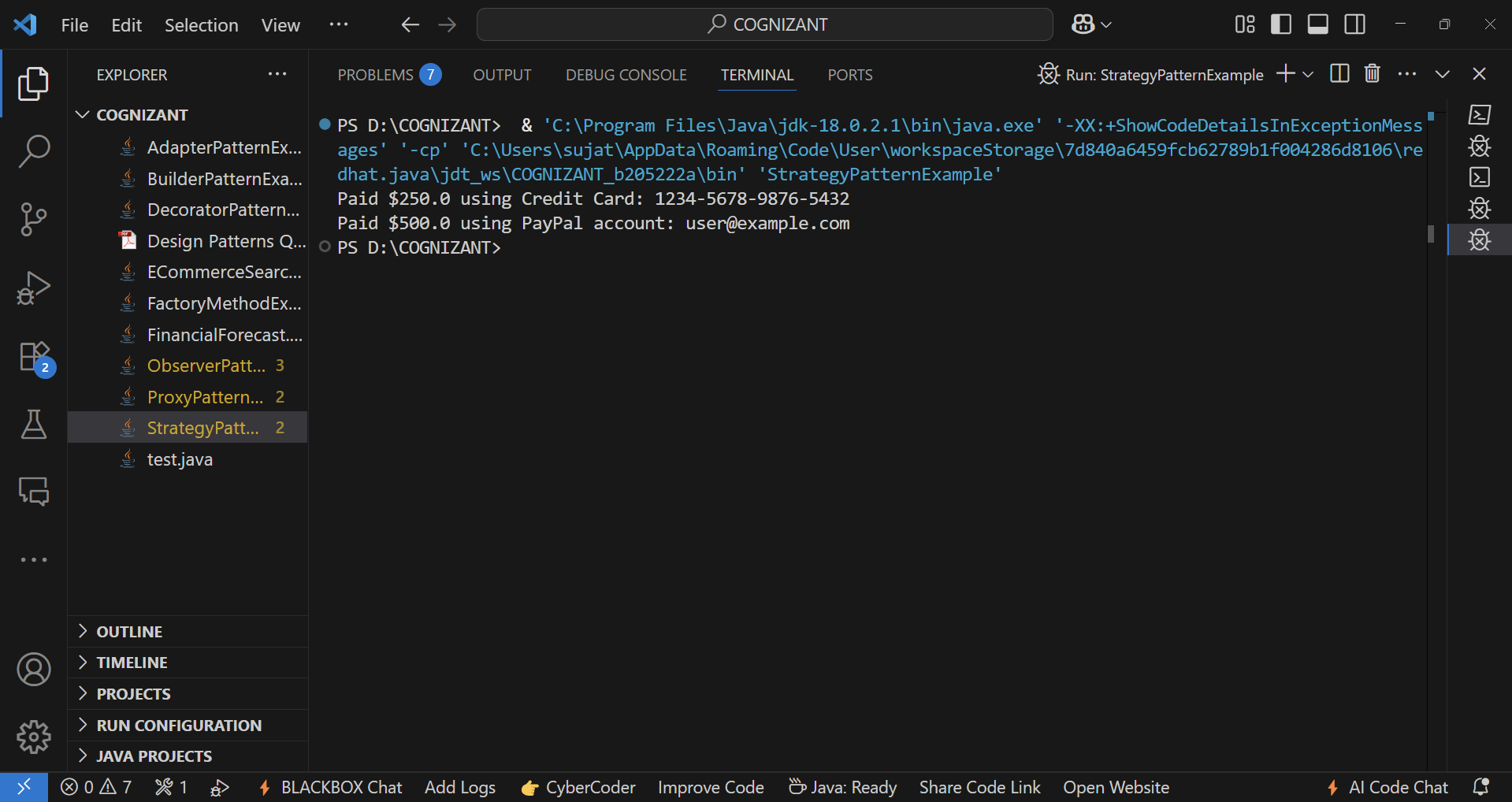
        // Pay using PayPal

        context.setPaymentStrategy(new PayPalPayment("user@example.com"));

        context.pay(500.00);

    }

}

**OUTPUT:-**