**Implementing the Strategy Pattern**

**StrategyPatternExample:**

interface PaymentStrategy {

    void pay(double amount);

}

class CreditCardPayment implements PaymentStrategy {

    public void pay(double amount) {

        System.out.println("₹" + amount + " has been paid using the Credit Card.");

    }

}

class PayPalPayment implements PaymentStrategy {

    public void pay(double amount) {

        System.out.println("₹" + amount + " has been paid using PayPal.");

    }

}

class PaymentContext {

    private PaymentStrategy strategy;

    public void setStrategy(PaymentStrategy strategy) {

        this.strategy = strategy;

    }

    public void makePayment(double amount) {

        strategy.pay(amount);

    }

}

public class StrategyPatternExample {

    public static void main(String[] args) {

        PaymentContext context = new PaymentContext();

        context.setStrategy(new CreditCardPayment());

        context.makePayment(5000);

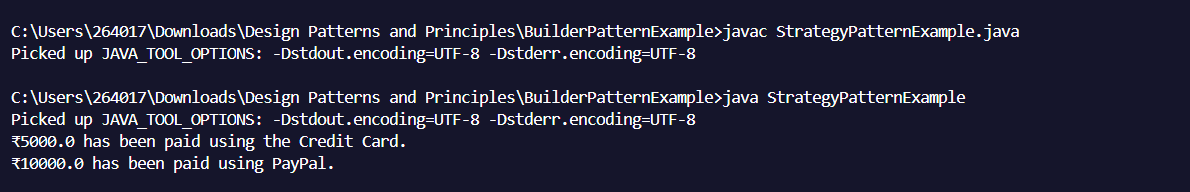
        context.setStrategy(new PayPalPayment());

        context.makePayment(10000);

    }

}

**Output:**

****