**Exercise 8: Implementing the Strategy Pattern**

**PaymentStrategy.java**

package com.mycompany.strategypattern;

public interface PaymentStrategy {

    void pay(double amount);

}

**CreditCardPayment.java**

package com.mycompany.strategypattern;

public class CreditCardPayment implements PaymentStrategy {

    private String cardNumber;

    public CreditCardPayment(String cardNumber) {

        this.cardNumber = cardNumber;

    }

    @Override

    public void pay(double amount) {

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

    }

}

**PayPalPayment.java**

package com.mycompany.strategypattern;

public class PayPalPayment implements PaymentStrategy {

    private String email;

    public PayPalPayment(String email) {

        this.email = email;

    }

    @Override

    public void pay(double amount) {

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

    }

}

**UpiPayment.java**

package com.mycompany.strategypattern;

public class UpiPayment implements PaymentStrategy {

    private String upiId;

    public UpiPayment(String upiId) {

        this.upiId = upiId;

    }

    @Override

    public void pay(double amount) {

        System.out.println("Paid Rs. " + amount + " using UPI: " + upiId);

    }

}

**PaymentContext.java**

package com.mycompany.strategypattern;

public class PaymentContext {

    private PaymentStrategy strategy;

    public void setStrategy(PaymentStrategy strategy) {

        this.strategy = strategy;

    }

    public void executePayment(double amount) {

        if (strategy == null) {

            System.out.println("No payment method selected!");

        } else {

            strategy.pay(amount);

        }

    }

}

**StrategyPattern.java**

package com.mycompany.strategypattern;

import java.util.Scanner;

public class StrategyPattern {

 public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        PaymentContext context = new PaymentContext();

        while (true) {

            System.out.println("\n--- Select Payment Method ---");

            System.out.println("1. Credit Card");

            System.out.println("2. PayPal");

            System.out.println("3. UPI");

            System.out.println("4. Exit");

            System.out.print("Enter choice: ");

            int choice = scanner.nextInt();

            if (choice == 4) break;

            System.out.print("Enter amount: ₹");

            double amount = scanner.nextDouble();

            switch (choice) {

                case 1:

                    System.out.print("Enter card number: ");

                    String card = scanner.next();

                    context.setStrategy(new CreditCardPayment(card));

                    break;

                case 2:

                    System.out.print("Enter PayPal email: ");

                    String email = scanner.next();

                    context.setStrategy(new PayPalPayment(email));

                    break;

                case 3:

                    System.out.print("Enter UPI ID: ");

                    String upi = scanner.next();

                    context.setStrategy(new UpiPayment(upi));

                    break;

                default:

                    System.out.println("Invalid choice.");

                    continue;

            }

            context.executePayment(amount);

        }

        scanner.close();

        System.out.println("Payment system closed.");

    }

}

**Output**

