**Exercise 8: Implementing the Strategy Pattern**

public class Main {

    public static void main(String[] args) {

        PaymentContext context;

        context = new PaymentContext(new CreditCardPayment("John Doe", "1234-5678-9012-3456"));

        context.pay(250.00);

        context = new PaymentContext(new PayPalPayment("john@example.com"));

        context.pay(180.50);

    }

}

interface PaymentStrategy {

    void pay(double amount);

}

class CreditCardPayment implements PaymentStrategy {

    private String name;

    private String cardNumber;

    public CreditCardPayment(String name, String cardNumber) {

        this.name = name;

        this.cardNumber = cardNumber;

    }

    @Override

    public void pay(double amount) {

        System.out.println("Paid ₹" + amount + " using Credit Card [" + cardNumber + "] by " + name);

    }

}

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);

    }

}

class PaymentContext {

    private PaymentStrategy strategy;

    public PaymentContext(PaymentStrategy strategy) {

        this.strategy = strategy;

    }

    public void pay(double amount) {

        strategy.pay(amount);

    }

}

Output:

C:\Users\VANITHA\AppData\Local\Packages\Microsoft.Windows.Photos_8wekyb3d8bbwe\TempState\ShareServiceTempFolder\Screenshot 2025-06-20 191126.jpeg