**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 –

package Design\_Patterns\_And\_Principles.StrategyPatternExample;

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.");

    }

}

 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.");

    }

}

 class PaymentContext {

    private PaymentStrategy paymentStrategy;

    public PaymentContext(PaymentStrategy paymentStrategy) {

        this.paymentStrategy = paymentStrategy;

    }

    public void executePayment(double amount) {

        paymentStrategy.pay(amount);

    }

}

public class StrategyPatternDemo {

    public static void main(String[] args) {

        // Create different payment strategies

        PaymentStrategy creditCard = new CreditCardPayment("John Doe", "1234-5678-9012-3456");

        PaymentStrategy paypal = new PayPalPayment("john.doe@example.com");

        // Create PaymentContext and set strategy

        PaymentContext paymentContext = new PaymentContext(creditCard);

        System.out.println("Using Credit Card Payment:");

        paymentContext.executePayment(100.00);

        System.out.println();

        // Change strategy to PayPal

        paymentContext = new PaymentContext(paypal);

        System.out.println("Using PayPal Payment:");

        paymentContext.executePayment(200.00);

    }

}