**Exercise 4: Implementing the Adapter Pattern**

**SCENARIO:**

You are developing a payment processing system that needs to integrate with multiple third-party payment gateways with different interfaces. Use the Adapter Pattern to achieve this.

**STEPS:**

**1. Create a New Java Project**:

* Create a new Java project named **AdapterPatternExample**.

**2. Define Target Interface**:

* Create an interface **PaymentProcessor** with methods like **processPayment().**

**3. Implement Adaptee Classes**:

* Create classes for different payment gateways with their own methods.

**4. Implement the Adapter Class**:

* Create an adapter class for each payment gateway that implements PaymentProcessor and translates the calls to the gateway-specific methods.

**5. Test the Adapter Implementation**:

* Create a test class to demonstrate the use of different payment gateways through the adapter.

**SOLUTION:**

**Step 1: Create a New Java Project**

Create a new Java project named AdapterPatternExample.

**Step 2: Define Target Interface**

// PaymentProcessor

interface PaymentProcessor {

void processPayment(int amount);

}

**Step 3: Implement Adaptee Classes**

**//**UPISystem

class UPISystem {

private String UPIid;

private String username;

private String phoneNumber;

public UPISystem(String UPIid, String username, String phoneNumber) {

this.UPIid = UPIid;

this.username = username;

this.phoneNumber = phoneNumber;

}

public void performUPIPayment(int amount) {

System.out.println("Processing UPI Payment");

System.out.println("Username: " + username);

System.out.println("UPI ID: " + UPIid);

System.out.println("Phone: " + phoneNumber);

System.out.println("Amount paid: " + amount);

}

}

//PayPalSystem

class PayPalSystem {

private String email;

private String username;

public PayPalSystem(String email, String username) {

this.email = email;

this.username = username;

}

public void performPayPalPayment(int amount) {

System.out.println("Processing PayPal Payment");

System.out.println("Username: " + username);

System.out.println("Email: " + email);

System.out.println("Amount paid: " + amount);

}

}

//CreditCardSystem

class CreditCardSystem {

private String cardNumber;

private String cardHolder;

public CreditCardSystem(String cardNumber, String cardHolder) {

this.cardNumber = cardNumber;

this.cardHolder = cardHolder;

}

public void performCreditCardPayment(int amount) {

System.out.println("Processing Credit Card Payment");

System.out.println("Card Holder: " + cardHolder);

System.out.println("Card Number: " + cardNumber);

System.out.println("Amount paid: " + amount);

}

}

**Step 4: Implement the Adapter Classes**

// UPIPaymentAdapter

class UPIPaymentAdapter implements PaymentProcessor {

private UPISystem upiSystem;

public UPIPaymentAdapter(UPISystem upiSystem) {

this.upiSystem = upiSystem;

}

@Override

public void processPayment(int amount) {

upiSystem.performUPIPayment(amount);

}

}

// PayPalPaymentAdapter

class PayPalPaymentAdapter implements PaymentProcessor {

private PayPalSystem payPalSystem;

public PayPalPaymentAdapter(PayPalSystem payPalSystem) {

this.payPalSystem = payPalSystem;

}

@Override

public void processPayment(int amount) {

payPalSystem.performPayPalPayment(amount);

}

}

// CreditCardPaymentAdapter

class CreditCardPaymentAdapter implements PaymentProcessor {

private CreditCardSystem creditCardSystem;

public CreditCardPaymentAdapter(CreditCardSystem creditCardSystem) {

this.creditCardSystem = creditCardSystem;

}

@Override

public void processPayment(int amount) {

creditCardSystem.performCreditCardPayment(amount);

}

}

**Step 5: Create AdapterPatternExample Class**

// UPI Payment

UPISystem upiSystem = new UPISystem("louis@examplebank", "Louis", "9876565544");

PaymentProcessor upiPayment = new UPIPaymentAdapter(upiSystem);

// PayPal Payment

PayPalSystem payPalSystem = new PayPalSystem("peter@example.com", "Peter");

PaymentProcessor payPalPayment = new PayPalPaymentAdapter(payPalSystem);

// Credit Card Payment

CreditCardSystem creditCardSystem = new CreditCardSystem("1234-5678-9876-5432", "Barney");

PaymentProcessor creditCardPayment = new CreditCardPaymentAdapter(creditCardSystem);

// Demonstrating different payment methods

System.out.println("Processing UPI Payment:");

upiPayment.processPayment(1500);

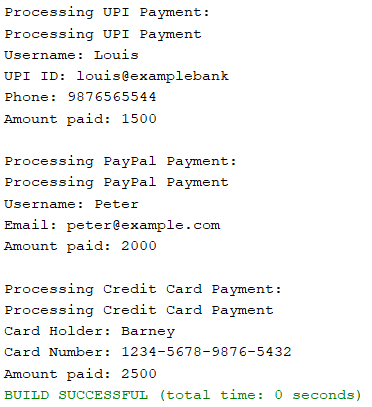
System.out.println("\nProcessing PayPal Payment:");

payPalPayment.processPayment(2000);

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

creditCardPayment.processPayment(2500);

**SAMPLE OUTPUT:**

****