**Exercise 4: Implementing the Adapter Pattern**

CODE:

public class AdapterPatternExample {

interface PaymentProcessor {

void processPayment(double amount);

}

static class PayPal {

public void sendPayment(double amount) {

System.*out*.println("Processing PayPal payment of $" + amount);

}

}

static class Stripe {

public void makePayment(double amountInDollars) {

System.*out*.println("Processing Stripe payment of $" + amountInDollars);

}

}

static class PayPalAdapter implements PaymentProcessor {

private PayPal paypal;

public PayPalAdapter(PayPal paypal) {

this.paypal = paypal;

}

@Override

public void processPayment(double amount) {

paypal.sendPayment(amount);

}

}

static class StripeAdapter implements PaymentProcessor {

private Stripe stripe;

public StripeAdapter(Stripe stripe) {

this.stripe = stripe;

}

@Override

public void processPayment(double amount) {

stripe.makePayment(amount);

}

}

public static void main(String[] args) {

PaymentProcessor paypalProcessor = new PayPalAdapter(new PayPal());

paypalProcessor.processPayment(150.00);

PaymentProcessor stripeProcessor = new StripeAdapter(new Stripe());

stripeProcessor.processPayment(275.50);

}

}

OUTPUT:

