CreditCardStrategy .java

package strategyclass;

public class CreditCardStrategy implements PaymentStrategy {

private String name;

private String cardNumber;

private String cvv;

private String dateOfExpiry;

public CreditCardStrategy(String nm, String ccNum, String cvv, String expiryDate){

this.name=nm;

this.cardNumber=ccNum;

this.cvv=cvv;

this.dateOfExpiry=expiryDate;

}

@Override

public void pay(int amount) {

System.out.println(amount +" paid with credit/debit card");

}

}

Item.java

package strategyclass;

public class Item {

private String upcCode;

private int price;

public Item(String upc, int cost){

this.upcCode=upc;

this.price=cost;

}

public String getUpcCode() {

return upcCode;

}

public int getPrice() {

return price;

}

}

package strategyclass;

public interface PaymentStrategy {

public void pay(int amount);

}

package strategyclass;

public class PaypalStrategy implements PaymentStrategy {

private String emailId;

private String password;

public PaypalStrategy(String email, String pwd){

this.emailId=email;

this.password=pwd;

}

@Override

public void pay(int amount) {

System.out.println(amount + " paid using Paypal.");

}

}

package strategyclass;

import java.text.DecimalFormat;

import java.util.ArrayList;

import java.util.List;

public class ShoppingCart {

//List of items

List<Item> items;

public ShoppingCart(){

this.items=new ArrayList<Item>();

}

public void addItem(Item item){

this.items.add(item);

}

public void removeItem(Item item){

this.items.remove(item);

}

public int calculateTotal(){

int sum = 0;

for(Item item : items){

sum += item.getPrice();

}

return sum;

}

public void pay(PaymentStrategy paymentMethod){

int amount = calculateTotal();

paymentMethod.pay(amount);

}

}

package strategyclass;

public class ShoppingCartTest {

public static void main(String[] args) {

ShoppingCart cart = new ShoppingCart();

Item item1 = new Item("1234",60);

Item item2 = new Item("5678",40);

cart.addItem(item1);

cart.addItem(item2);

//pay by paypal

cart.pay(new PaypalStrategy("myemail@example.com", "pwd"));

//pay by credit card

cart.pay(new CreditCardStrategy("Aparna Bindage", "1234567890123456", "786", "12/15"));

}

}