```
//problem Statement
/*
  Implement and apply Strategy Design pattern for simple Shopping Cart where three payment
  strategies are used such as Credit Card, PayPal, Bit Coin. Create an interface for strategy pattern
  and give concrete implementation for payment.
*/

package assignment;
import java.util.Scanner;

//====================== INTERFACE PaymentProcessor =======================//
interface PaymentProcessor {

        void pay(int amount);//interface method pay

}

//====================== CLASS CreditCard =======================//
//implementing PaymentProcessor interface
class CreditCard implements PaymentProcessor {
        Scanner sc =new Scanner (System.in);//creating object of scanner class
        String name,ExpDate;//declaration of name,ExpDate
        double CardNo;//declaration of CardNo

        //Constructor of CreditCard class
```

```
    CreditCard(){
        super();//calling parent class constructor
        System.out.println("------------------------------------------------------");
        System.out.print("\tCard holder Name :: ");//printing on console
        this.name =sc.next();//taking Card holder Name as input from user
        System.out.print("\tCard Number :: ");//printing on console
        this.CardNo =sc.nextDouble();//taking Card Number as input from user
        System.out.print("\tCard Expire Date :: ");//printing on console
        this.ExpDate =sc.next();//taking Card Expire Date as input from user
        System.out.println("------------------------------------------------------");
    }

    @Override
    public void pay(int amount) {    //method for payment
        System.out.println("------------------------------------------------------");
       System.out.println("Paying through CreditCard payment: Charging $" + amount);
       System.out.println("------------------------------------------------------");
    }



}



//====================== CLASS PayPal ======================//
//implementing PaymentProcessor interface
```

```
class PayPal implements PaymentProcessor {


        //Constructor of PayPal class

        PayPal(){

                super();//calling parent class constructor

                System.out.println("\nChecking Internet Connection........");

          }


        @Override

         public void pay(int amount) {     //method for payment

                System.out.println("--------------------------------------------------------");

                System.out.println("Paying through PayPal payment: Charging $" + amount);

                System.out.println("--------------------------------------------------------");

         }


        }


//====================== CLASS BitCoin ======================//

//implementing PaymentProcessor interface

class BitCoin implements PaymentProcessor {

        Scanner sc =new Scanner (System.in);//creating object of scanner class

        String add;//declaration of add


        //Constructor of BitCoin class

        BitCoin(){
                super();//calling parent class constructor

                System.out.print("\nEnter Transaction 'Input Address' :: ");//asking user of address

                add= sc.next();//taking 'INPUT ADDRESS' as input from user


     }


    @Override

    public void pay(int amount) {      //method for payment

                System.out.println("--------------------------------------------------------");

                 System.out.println("Paying through BitCoin payment: Charging $" + amount);

                 System.out.println("--------------------------------------------------------");

     }


}




//====================== CLASS Order ======================//


class Order {


    private final PaymentProcessor paymentProcessor;//declaration of paymentProcessor object

    private final int amount;//declaration of amount


    //Order Method

    public Order(int amount, PaymentProcessor paymentProcessor) {
            this.amount = amount;//storing value

            this.paymentProcessor = paymentProcessor;//storing value

          }


        //process Method

        public void process() {

            paymentProcessor.pay(amount);//calling pay method

          }


}




//====================== CLASS Main ======================//

public class Main {

        //calling static void main method

        public static void main(String[] args) {

                int c,amt=0;//declaration of c, amt

                Order order;//reference of order assign to order obj

                Scanner sc = new Scanner(System.in);//creating object of scanner class

                while(true) {//while loop for menu driven

                        System.out.println();

                        //menu bar

                        System.out.println("**** SHOPING CART ****");

                        System.out.print("1.Credit Card \n2.PayPal \n3.BitCoin \n4.Exit");

                        System.out.print("\n\nEnter the Choice ::");
```

```
c=sc.nextInt();//taking input from user

System.out.println("---------------------------------------------------------");

if(c==1||c==2||c==3) {//check whether 0<c<4

            System.out.print("\nEnter amount tobe Tranfer :: ");

            amt = sc.nextInt();//taking amt as input from user

            System.out.println("---------------------------------------------------------");

}

//switch case

switch(c) {

 case 1://for input c ==1

             order = new Order(amt, new CreditCard());//creating obj of order class

             order.process();//calling process method of order class

            break;


 case 2://for input c == 2

             order = new Order(amt, new PayPal());//creating obj of order class

             order.process();//calling process method of order class

            break;


 case 3://for input c == 3

             order = new Order(amt, new BitCoin());//creating obj of order class

             order.process();//calling process method of order class

            break;


 case 4:
```

```
                                        System.out.println("\nThank you For Shopping !!!! ");//printing on
console
                                        System.out.println("------------------------------------------------------");
                                        return;//stop execution of program

                              default:
                                        System.out.println("Invalid Payment Mode !!!");// default
                                        System.out.println("------------------------------------------------------");
                              }
                    }
          }
}

/*

 ##OUTPUT##

**** SHOPING CART ****
1.Credit Card
2.PayPal
3.BitCoin
4.Exit
```

```
Enter the Choice ::1

---------------------------------------------------------


Enter amount tobe Tranfer :: 350

---------------------------------------------------------

---------------------------------------------------------

            Card holder Name :: Vaibhav

            Card Number :: 785423695628

            Card Expire Date :: 12/24

---------------------------------------------------------

---------------------------------------------------------

Paying through CreditCard payment: Charging $350

---------------------------------------------------------


**** SHOPING CART ****

1.Credit Card

2.PayPal

3.BitCoin

4.Exit


Enter the Choice ::2

---------------------------------------------------------


Enter amount tobe Tranfer :: 5000
```

```
---------------------------------------------------------


Checking Internet Connection........

---------------------------------------------------------

Paying through PayPal payment: Charging $5000

---------------------------------------------------------


**** SHOPING CART ****

1.Credit Card

2.PayPal

3.BitCoin

4.Exit


Enter the Choice ::3

---------------------------------------------------------


Enter amount tobe Tranfer :: 10000

---------------------------------------------------------


Enter Transaction 'Input Address' :: 5342.9324.2671.1354

---------------------------------------------------------

Paying through BitCoin payment: Charging $10000

---------------------------------------------------------


**** SHOPING CART ****
```

```
1.Credit Card
2.PayPal
3.BitCoin
4.Exit

Enter the Choice ::5
--------------------------------------------------------
Invalid Payment Mode !!!
--------------------------------------------------------

**** SHOPING CART ****
1.Credit Card
2.PayPal
3.BitCoin
4.Exit

Enter the Choice ::4
--------------------------------------------------------

Thank you For Shopping !!!!
--------------------------------------------------------
*/
```