

```
public class EcommerceSystem {  
  
    public static void main(String[] args) {  
        Scanner sc = new Scanner(System.in) ;  
  
        JOptionPane.showMessageDialog(null,"Welcome to E-commerce System");  
  
        int id = Integer.parseInt(JOptionPane.showInputDialog("Please, Enter your id")) ;  
        String name = JOptionPane.showInputDialog("Enter your name") ;  
        String address = JOptionPane.showInputDialog("Enter your address") ;  
        int numOfItems = Integer.parseInt(JOptionPane.showInputDialog("How many products you want to add to your cart"));  
  
        Customer c = new Customer(id , name , address) ;  
  
        if(numOfItems<0){  
            numOfItems = Math.abs(numOfItems) ;  
        }  
        else if(numOfItems>0){  
            numOfItems = numOfItems ;  
        }  
    }  
}
```

```
Cart cartt = new Cart(id , numOfItems) ;
```

```
ElectronicProduct ep = new ElectronicProduct(1 , "smart phone" , (float) -599.9, "Samsung" , 1);
```

```
ClothingProduct cp = new ClothingProduct(2 , "T-shirt" , (float)19.99 , "Medium" , "Cotton");
```

```
BookProduct bp = new BookProduct(3 , "OOP" , (float)-39.99 , "O'Reilly" , "X Publications");
```

```
for(int i = 0 ; i<numOfItems ; i++){
```

```
    int Itemnum = Math.abs(Integer.parseInt(JOptionPane.showInputDialog("Which product you would like to add? 1-Smart phone 2-T-shirt 3-Book")));
```

```
    switch(Itemnum){
```

```
        case 1 :
```

```
            cartt.addProduct(ep);
```

```
            break ;
```

```
        case 2 :
```

```
            cartt.addProduct(cp);
```

```
public class Product {  
    private int productId ;  
    private String name ;  
    private float price ;  
  
    Product() {  
  
    }  
    Product( int productId , String name , float price ){  
        this.productId = Math.abs(productId) ;  
        this.name = name ;  
        this.price = Math.abs(price) ;  
    }  
  
    public int getProductId() {  
        return productId;  
    }  
  
    public void setProductId(int productId) {  
        this.productId = Math.abs(productId);  
    }  
  
    public String getName() {  
        return name;  
    }  
}
```

```
        break ;
    case 3 :

        cartt.addProduct(bp);

        break ;
    default :
        JOptionPane.showMessageDialog(null, "the option you selected is not found!");
    }
}

cartt.placeOrder();
```

```
public int getProductId() {  
    return productId;  
}  
  
public void setProductId(int productId) {  
    this.productId = Math.abs(productId);  
}  
  
public String getName() {  
    return name;  
}  
  
public void setName(String name) {  
    this.name = name;  
}  
  
public float getPrice() {  
    return price;  
}  
  
public void setPrice(float price) {  
    this.price = Math.abs(price);  
}
```

```
public class ElectronicProduct extends Product {  
    private String brand ;  
    private int warrantyPeriod ;  
  
    ElectronicProduct(){  
  
    }  
  
    ElectronicProduct(int productId , String name , float price , String brand , int warrantyPeriod ){  
        super(productId , name , price) ;  
        this.brand = brand ;  
        this.warrantyPeriod = Math.abs(warrantyPeriod) ;  
  
    }  
  
    public String getBrand() {  
        return brand;  
    }  
  
    public void setBrand(String brand) {  
        this.brand = brand;  
    }  
  
    public int getWarrantyPeriod() {  
        return warrantyPeriod;  
    }  
}
```

```
public String getBrand() {  
    return brand;  
}
```

```
public void setBrand(String brand) {  
    this.brand = brand;  
}
```

```
public int getWarrantyPeriod() {  
    return warrantyPeriod;  
}
```

```
public void setWarrantyPeriod(int warrantyPeriod) {  
    this.warrantyPeriod = Math.abs(warrantyPeriod) ;  
}
```

```
public class ClothingProduct extends Product {

    private String size ;
    private String fabric ;

    ClothingProduct(){

    }

    ClothingProduct( int productId ,String name , float price , String size , String fabric ){
        super(productId , name , price ) ;
        this.size = size ;
        this.fabric = fabric ;
    }

    public String getSize() {
        return size;
    }

    public void setSize(String size) {
        this.size = size;
    }

    public String getFabric() {
```



```
public String getSize() {  
    return size;  
}
```

```
public void setSize(String size) {  
    this.size = size;  
}
```

```
public String getFabric() {  
    return fabric;  
}
```

```
public void setFabric(String fabric) {  
    this.fabric = fabric;  
}
```

```
public class BookProduct extends Product {

    private String author ;
    private String publisher ;

    BookProduct(){

    }

    BookProduct( int productId , String name , float price , String author , String publisher ) {
        super(productId , name , price ) ;
        this.author = author ;
        this.publisher = publisher ;
    }

    public String getAuthor() {
        return author;
    }

    public void setAuthor(String author) {
        this.author = author;
    }

    public String getPublisher() {
        return publisher;
    }
}
```

```
public class Customer {  
    private int customerId ;  
    private String name ;  
    private String address ;  
  
    Customer( ) {  
  
    }  
  
    Customer( int customerId ,String name ,String address) {  
  
        this.customerId = Math.abs(customerId) ;  
        this.name = name ;  
        this.address = address ;  
  
    }  
  
    public int getCustomerId() {  
        return customerId;  
    }  
  
    public void setCustomerId(int customerId) {  
        this.customerId = Math.abs(customerId);  
    }  
}
```

```
        this.publisher = publisher ;  
    }  
  
}
```

```
public String getAuthor() {  
    return author;  
}
```

```
public void setAuthor(String author) {  
    this.author = author;  
}
```

```
public String getPublisher() {  
    return publisher;  
}
```

```
public void setPublisher(String publisher) {  
    this.publisher = publisher;  
}
```

```
}
```

```
        return customerId;
    }

    public void setCustomerId(int customerId) {
        this.customerId = Math.abs(customerId);
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }
}
```

```
public void setCustomerId(int customerId) {  
    this.customerId = Math.abs(customerId);  
}
```

```
public int getnProducts() {  
    return nProducts;  
}
```

```
public void setnProducts(int nProducts) {  
    this.nProducts = Math.abs(nProducts);  
}
```

```
public Product[] getProducts() {  
    return products;  
}
```

```
public void setProducts(Product[] products) {  
    this.products = products;  
}
```

```
public Product[] addProduct(Product pro){  
  
    products[this.element]=pro;  
    this.element++;  
}
```

```

] import javax.swing.JOptionPane;
public class Cart {
    private int customerId ;
    private int nProducts ;
    private int element=0;
    private Product [] products ;    // size nProducts

]    Cart(){
-
    }

]    Cart( int customerId ,int nProducts ){
    this.customerId = Math.abs(customerId);
    this.nProducts = Math.abs(nProducts) ;
    this.products= new Product[this.nProducts] ;    // size nProducts
-
    }

]    public int getCustomerId() {
    return customerId;
-
    }

```




```
public Product[] addProduct(Product pro) {

    products[this.element]=pro;
    this.element++;
    return products ;
}

public void removeProduct(int productIndex) {
    if (productIndex < 0 || productIndex >= nProducts) {
        System.out.println("Invalid product index. Cannot remove product.");
        return;
    }
    for (int i = productIndex; i < nProducts - 1; i++) {
        products[i] = products[i + 1];
    }
    nProducts -- ;
}

public float calculatePrice(){
    float sum = 0 ;

    for ( Product r: products )
    {
        sum+=r.getPrice();
    }
}
```



```

float sum = 0 ;

for ( Product r: products )
{
    sum+=r.getPrice();
}

return sum ;
}

public void placeOrder(){

    int choice = Integer.parseInt(JOptionPane.showInputDialog("your total is $" + calculatePrice() + " .Would you like
    switch(choice){
        case 1 :
            Order o = new Order(this.customerId ,-1, products ,calculatePrice());
            o.printOrderInfo();
            break ;
        default:
            break ;
    }
}
}

```

```
import javax.swing.JOptionPane;
```

```
/**
```

```
 *
```

```
 * @author Malak Elsayed
```

```
 */
```

```
public class Order {
```

```
    private int customerId ;
```

```
    private int orderId ;
```

```
    private Product[] products ;
```

```
    private float totalPrice ;
```

```
    Order() {
```

```
    }
```

```
    Order(int customerId , int orderId ,Product []products , float totalPrice){
```

```
        this.customerId = Math.abs(customerId) ;
```

```
        this.orderId = Math.abs(orderId) ;
```

```
        this.totalPrice = Math.abs(totalPrice) ;
```

```
        this.products= products ;
```

```
    }
```

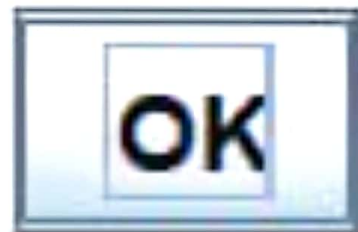
```
    public void printOrderInfo(){
```

```
        this.orderId = Math.abs(orderId) ;  
        this.totalPrice = Math.abs(totalPrice) ;  
        this.products= products ;  
    }  
  
    public void printOrderInfo(){  
        String messege = "" ;  
        for( Product p : products){  
            messege += (p.getName() + " - " +p.getPrice() + "$" + "\n") ;  
        }  
    }  
  
    JOptionPane.showMessageDialog(null, "Here is your order's summary\n" + "Order ID : " + this.orderId + "\n Customer ID : " + th  
        JOptionPane.showMessageDialog(null, "Thank you for ordering from  E-commerce System!");  
    }  
}
```

Message



Welcome to E-commerce System



Input



Please, Enter your id

23011157

OK

Cancel

Input



Enter your name

malak elsayed

OK

Cancel

Input



Enter your address

address

OK

Cancel

Input



How many products you want to add to your cart

OK

Cancel

Input



Which product you would like to add? 1-Smart phone 2-T-shirt 3 -OOP

OK

Cancel

Input



your total is \$679.87 .Would you like to place order? 1-yes 2-no

1

OK

Cancel

Message



Here is your order's summary

Order ID : 1

Customer ID :23011157

Products :

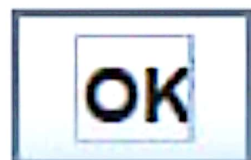
T-shirt - 19.99\$

OOP - 39.99\$

T-shirt - 19.99\$

smart phone - 599.9\$

Total price : 679.87



Message



Thank you for ordering from E-commerce System!

