

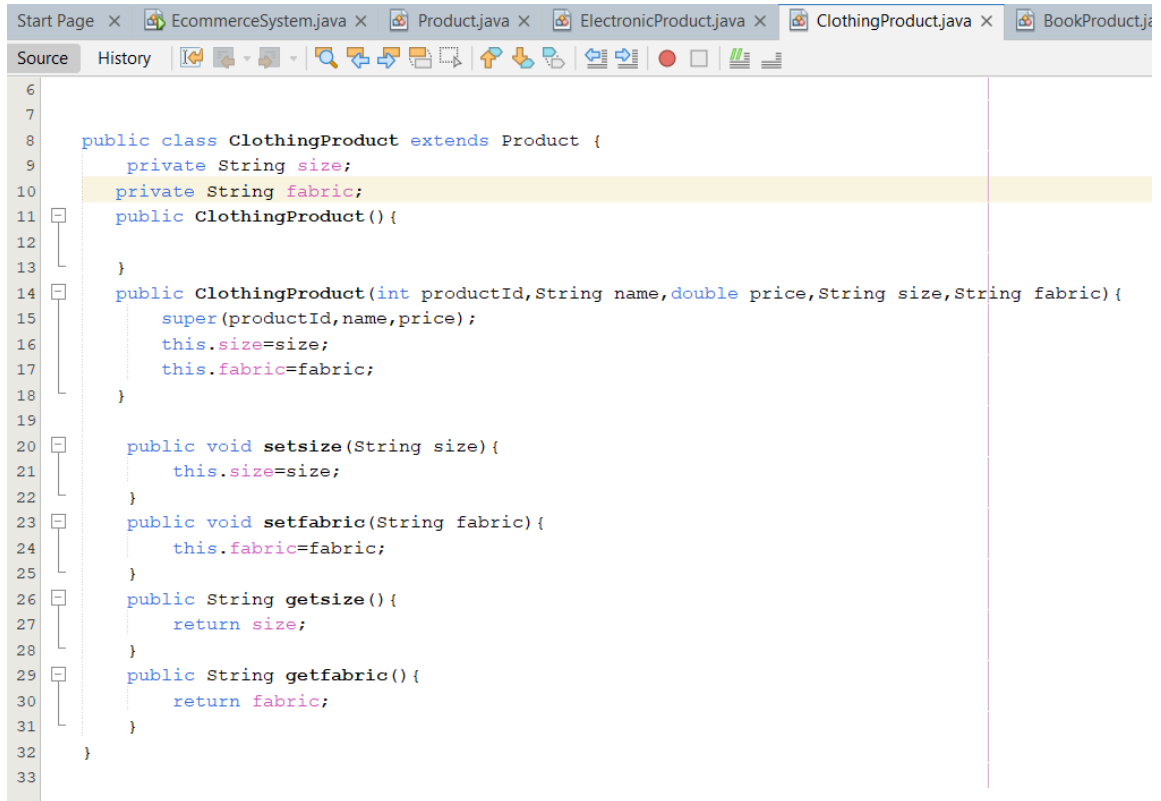
Product class:

```
Start Page x EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x
Source History
6 public class Product {
7     protected int productId;
8     protected String name;
9     protected double price;
10    public Product() {
11    }
12
13    public Product(int productId,String name,double price) {
14        this.productId=Math.abs(a: productId);
15        this.price=Math.abs(a: price);
16        this.name=name;
17    }
18
19    public void setproductId(int productId) {
20        this.productId=Math.abs(a: productId);
21    }
22
23    public void setname(String name){
24        this.name=name;
25    }
26
27    public void setprice(double price) {
28        if (price<0) {
29            this.price=Math.abs(a: price);
30        }
31        else{
32            this.price=price;
33        }
34    }
35
36    public String getname() {
37        return name;
38    }
39
40    public int getproductId() {
41        return productId;
42    }
43
44    public double getprice() {
45        return price;
46    }
47
48 }
```

Electronic Product class:

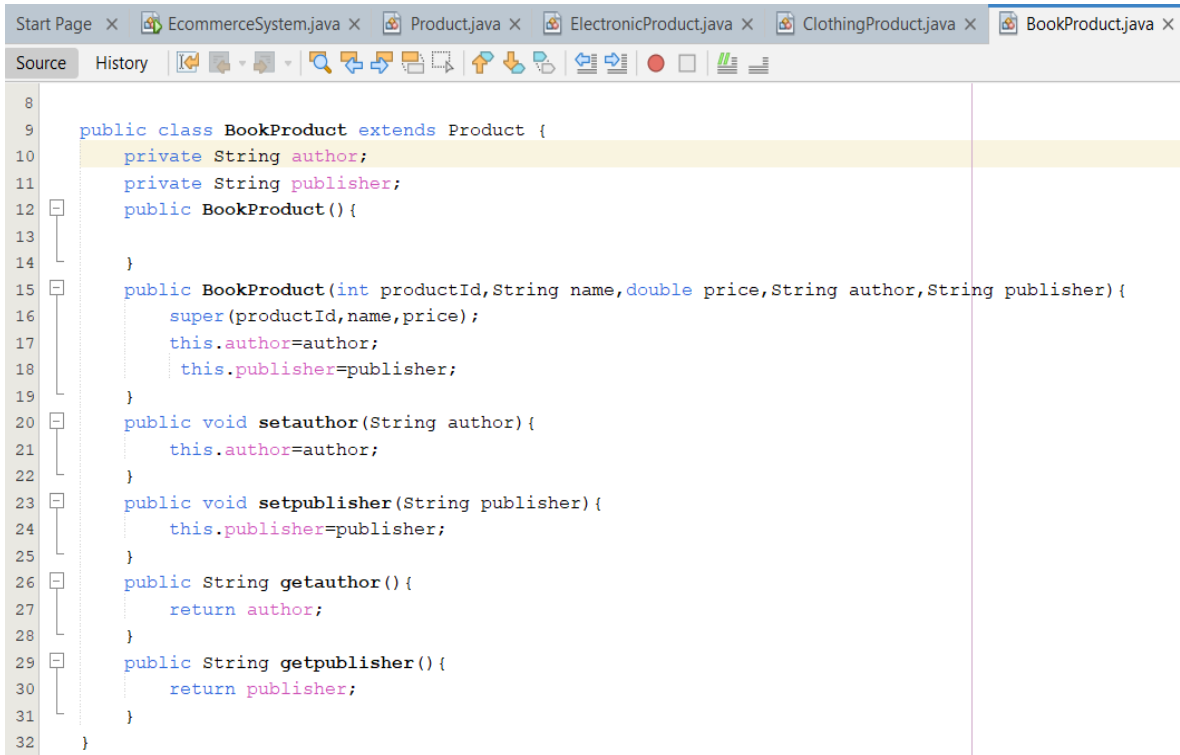
```
Start Page x EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x BookProduct.java x
Source History
6
7
8
9 public class ElectronicProduct extends Product {
10     private String brand;
11     private int warrantyPeriod;
12     public ElectronicProduct() {
13     }
14
15     public ElectronicProduct(int productId,String name,double price,String brand,int warrantyPeriod) {
16         super(productId,name,price);
17         this.warrantyPeriod=Math.abs(a: warrantyPeriod);
18         this.brand=brand;
19     }
20
21     public void setwarrantyPeriod(int warrantyPeriod) {
22         this.warrantyPeriod=Math.abs(a: warrantyPeriod);
23     }
24
25     public void setbrand(String brand) {
26         this.brand=brand;
27     }
28
29     public String getbrand() {
30         return brand;
31     }
32
33     public int getwarrantyPeriod() {
34         return warrantyPeriod;
35     }
36
37 }
```

clothing product class:



```
6
7
8 public class ClothingProduct extends Product {
9     private String size;
10    private String fabric;
11    public ClothingProduct() {
12    }
13
14    public ClothingProduct(int productId, String name, double price, String size, String fabric) {
15        super(productId, name, price);
16        this.size = size;
17        this.fabric = fabric;
18    }
19
20    public void setsize(String size) {
21        this.size = size;
22    }
23
24    public void setfabric(String fabric) {
25        this.fabric = fabric;
26    }
27
28    public String getsize() {
29        return size;
30    }
31
32    public String getfabric() {
33        return fabric;
34    }
35 }
```

book product class:



```
8
9 public class BookProduct extends Product {
10    private String author;
11    private String publisher;
12    public BookProduct() {
13    }
14
15    public BookProduct(int productId, String name, double price, String author, String publisher) {
16        super(productId, name, price);
17        this.author = author;
18        this.publisher = publisher;
19    }
20
21    public void setauthor(String author) {
22        this.author = author;
23    }
24
25    public void setpublisher(String publisher) {
26        this.publisher = publisher;
27    }
28
29    public String getauthor() {
30        return author;
31    }
32
33    public String getpublisher() {
34        return publisher;
35    }
36 }
```

customer class:

```
Start Page x EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x
Source History
6
7 public class Customer {
8     private int customerId;
9     private String address;
10    private String name;
11
12    public void setcustomerId(int customerId) {
13
14        this.customerId=Math.abs(a: customerId);
15    }
16    public void setaddress(String address){
17        this.address=address;
18    }
19    public void setname(String name){
20        this.name=name;
21    }
22
23    public String getname(){
24        return name;
25    }
26    public int getcustomerId(){
27        return customerId;
28    }
29    public String getaddress(){
30        return address;
31    }
32 }
33
```

Cart class:

```
Start Page x EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x
Source History
6 import java.util.Scanner;
7 public class Cart {
8     private int nProducts;
9     private int customerId;
10    private Product[] products=new Product [nProducts];
11    public Cart(){
12
13    }
14
15    public Cart(int customerId,int nProducts ){
16        this.nProducts=Math.abs(a: nProducts);
17        this.customerId=Math.abs(a: customerId);
18        this.products=new Product[nProducts];
19    }
20    public void setcustomerId(int customerId){
21
22        this.customerId=Math.abs(a: customerId);
23    }
24    public void setnProducts(int nProducts){
25        if (nProducts<=0){
26            this.nProducts=Math.abs(a: nProducts);
27        }
28        else{
29            this.nProducts=nProducts;
30        }
31    }
32
33    public void setproducts(Product[] products){
34        this.products =products;
35    }
36
37
38    public void addProduct(Product pr,int i) {
39        if (i < products.length) {
40            products[i] = pr;
41        }
42    }
43
44 }
```

```

Start Page x EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x BookProdi
Source History
42 }
43 public void removeProduct(Product pr) {
44     for (int i = 0; i < nProducts; i++) {
45         if (products[i] != null && products[i].getproductId() == pr.getproductId()) {
46             for (int j = i; j < nProducts - 1; j++) {
47                 products[j] = products[j + 1];
48             }
49             products[--nProducts] = null;
50             break;
51         }
52     }
53 }
54 public float calculatePrice() {
55     float totalPrice = 0;
56     for (int k=0;k<products.length;k++) {
57         totalPrice += products[k].getprice();
58     }
59     return totalPrice;
60 }
61
62 public void placeorder() {
63     System.out.println(x: "the order has been placed \nhere is your order summary:");
64 }
65 public Product[] getproducts(){
66     return products;
67 }
68 public int getcustomerId(){
69     return customerId;
70 }
71 public int getnProducts(){
72     return nProducts;
73 }
74 }

```

Order class:

```

public class Order {
    private int customerId;
    private int orderId;
    private Product[] products;
    private float totalPrice;
    private float calculateTotalPrice() {
        float totalPrice = 0;
        for (int k=0;k<products.length;k++) {
            totalPrice += products[k].getprice();
        }
        return totalPrice;
    }
    public Order(int customerId, int orderId, Product[] products) {
        this.customerId = Math.abs(a: customerId);
        this.orderId = Math.abs(a: orderId);
        this.products = products;
        this.totalPrice = calculateTotalPrice();
    }
    public void printOrderInfo() {
        System.out.println("Order ID: " + orderId);
        System.out.println("Customer ID: " + customerId);
        System.out.println(x: "Products:");
        for (int y=0;y<products.length;y++) {
            System.out.println("- " + products[y].getname() + ": " + products[y].getprice());
        }
        System.out.println("Total Price: " + totalPrice);
    }
}

```

## E commerce system:

```
...ge EcommerceSystem.java x Product.java x ElectronicProduct.java x ClothingProduct.java x BookProduct.java x Customer.java x Cart.java x Order.java x
Source History

7 import java.util.Scanner;
8 public class EcommerceSystem {
9
10     public static void main(String[] args){
11         Scanner input=new Scanner (source:System.in);
12         System.out.print(s: "enter your name:");
13         String n=input.next();
14         System.out.print(s: "enter your id:");
15         int id=input.nextInt();
16         System.out.print(s: "enter your address:");
17         String add=input.next();
18         Customer cul=new Customer();
19         cul.setname(name: n);
20         cul.setaddress(address: add);
21         cul.setcustomerId(customerId: id);
22         System.out.println(x: "How many products do you want to add to the cart?");
23         int nPr = input.nextInt();
24         int x=cul.getcustomerId();
25         Cart cart1=new Cart (customerId: x, nProducts: nPr);
26         for (int i = 0; i < nPr; i++) {
27             System.out.println(x: "which one would you like to add (1 for smartphone, 2 for T-shirt, 3 for OOP)?");
28             int productType = input.nextInt();
29             switch (productType) {
30                 case 1:
31                     ElectronicProduct e1 = new ElectronicProduct (productId:1,name: "smart phone",price: 599.9,brand: "samsung",warrantyPeriod: 1);
32                     cart1.addProduct(pr: e1,i);
33                     break;
34                 case 2:
35                     ClothingProduct c1 = new ClothingProduct (productId:2,name: "t-shirt",price: 19.99,size: "Medium",fabric: "cotton");
36                     cart1.addProduct(pr: c1,i);
37                     break;
38                 case 3:
39                     BookProduct b1 = new BookProduct (productId:3,name: "oop",price: 39.99,author: "o.Reiley",publisher: "xpublications");
40                     cart1.addProduct(pr: b1,i);
41                     break;
42                 default:
43                     System.out.println(x: "Invalid product type!");
44             }
45         }
46
47         System.out.println(x: "Do you want to place an order for the products in the cart? (1=yes 2=no)");
48         int choice = input.nextInt();
49         if (choice==1){
50             cart1.placeorder();
51
52             Order order=new Order (customerId: x,orderId: 1,products: cart1.getproducts());
53             order.printOrderInfo();
54         }
55         else if(choice==2){
56             System.out.println(x: "Order has been cancelled");
57         }
58         else{
59             System.out.println(x: "wrong input please choose 1 or 2");
60         }
61     }
62 }
63
64
65
66
```

The output:

```
run:
enter your name:hager
enter your id:23012070
enter your address:alex
How many products do you want to add to the cart?
3
which one would you like to add (1 for smartphone, 2 for T-shirt, 3 for OOP)?
1
which one would you like to add (1 for smartphone, 2 for T-shirt, 3 for OOP)?
2
which one would you like to add (1 for smartphone, 2 for T-shirt, 3 for OOP)?
3
Do you want to place an order for the products in the cart? (1=yes 2=no)
1
the order has been placed
here is your order summary:
Order ID: 1
Customer ID: 23012070
Products:
- smart phone: 599.9
- t-shirt: 19.99
- oop: 39.99
Total Price: 659.88
BUILD SUCCESSFUL (total time: 23 seconds)
|
```