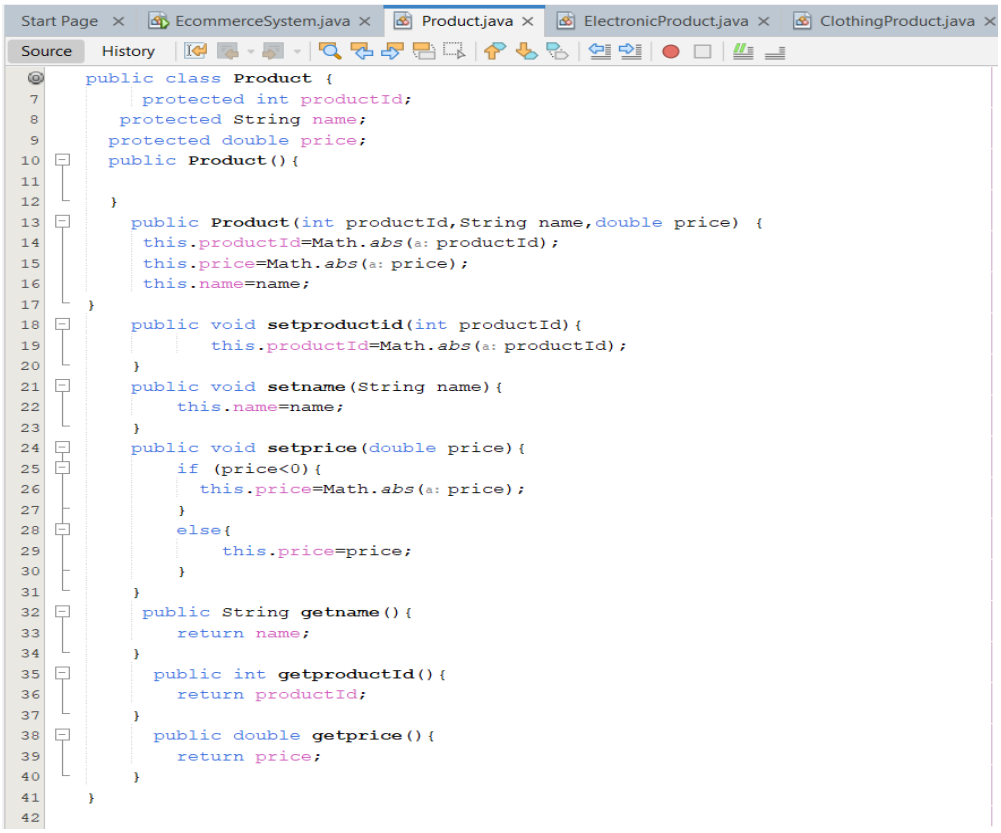
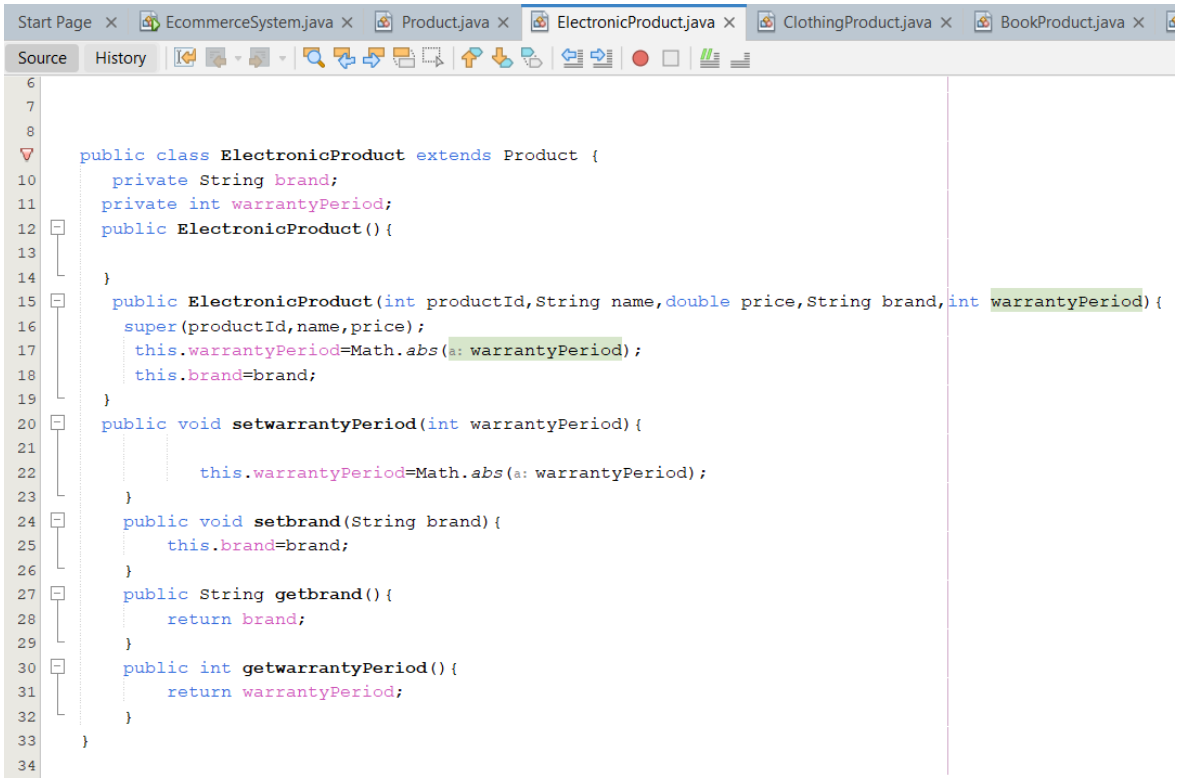
Product class:



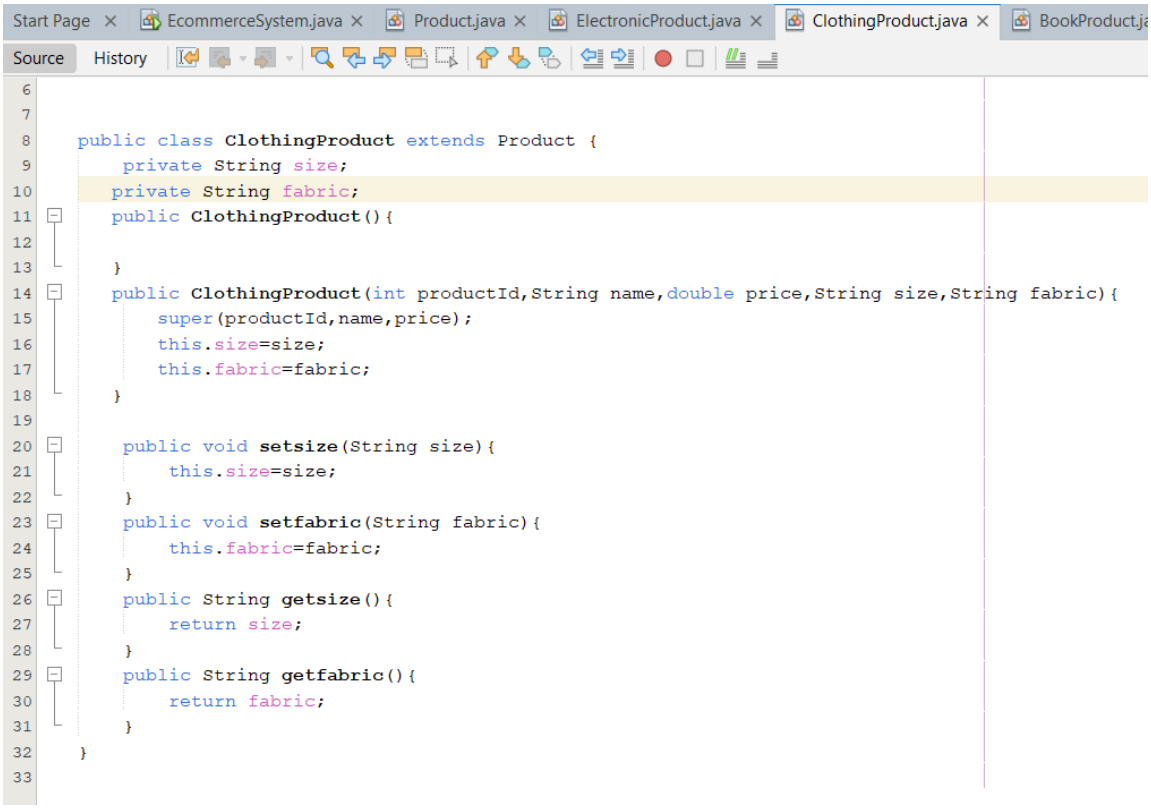
```java
public class Product {
    protected int productId;
   protected String name;
  protected double price;
  public Product(){

  }
    public Product(int productId,String name,double price) {
     this.productId=Math.abs(a: productId);
     this.price=Math.abs(a: price);
     this.name=name;
}
    public void setproductid(int productId){
          this.productId=Math.abs(a: productId);
    }
    public void setname(String name){
        this.name=name;
    }
    public void setprice(double price){
        if (price<0){
          this.price=Math.abs(a: price);
        }
        else{
            this.price=price;
        }
    }
     public String getname(){
        return name;
    }
      public int getproductId(){
        return productId;
    }
      public double getprice(){
        return price;
    }
}
```

Electronic Product class:



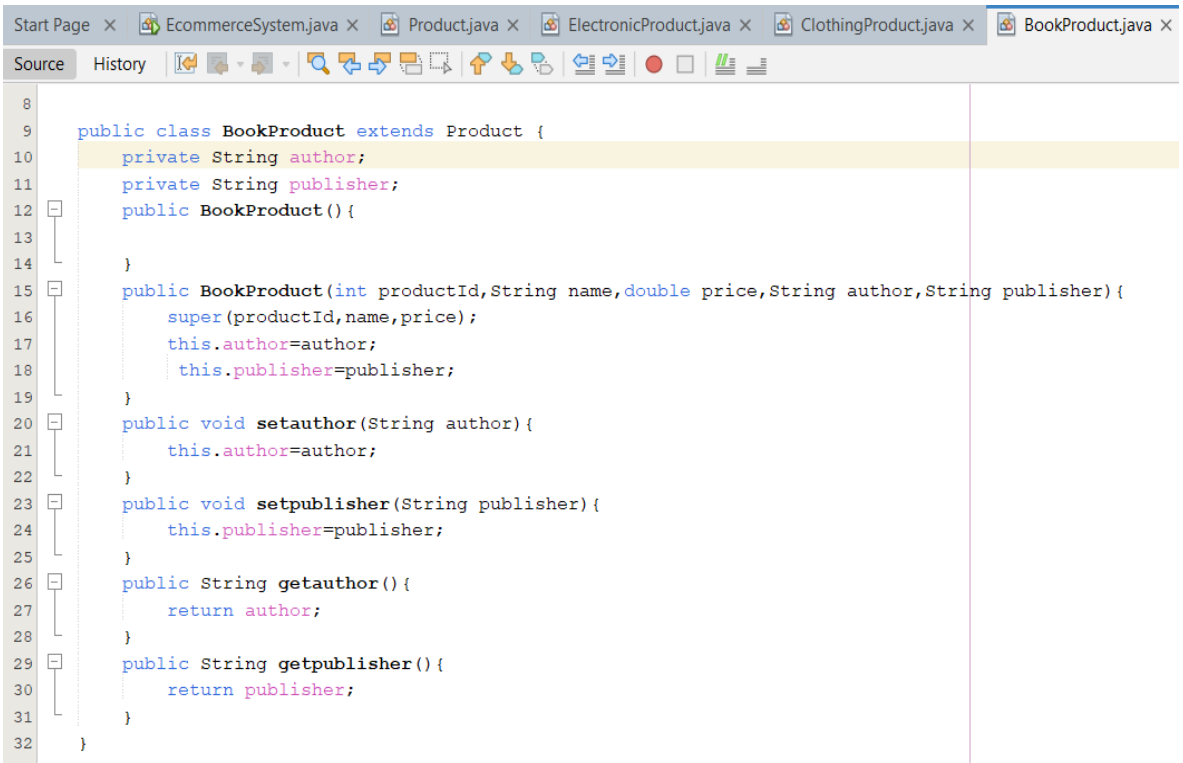
```java
public class ElectronicProduct extends Product {
   private String brand;
  private int warrantyPeriod;
  public ElectronicProduct(){

  }
   public ElectronicProduct(int productId,String name,double price,String brand,int warrantyPeriod){
    super(productId,name,price);
     this.warrantyPeriod=Math.abs(a: warrantyPeriod);
     this.brand=brand;
  }
  public void setwarrantyPeriod(int warrantyPeriod){

        this.warrantyPeriod=Math.abs(a: warrantyPeriod);
    }
    public void setbrand(String brand){
        this.brand=brand;
    }
    public String getbrand(){
        return brand;
    }
    public int getwarrantyPeriod(){
        return warrantyPeriod;
    }
}
```

clothing product class:



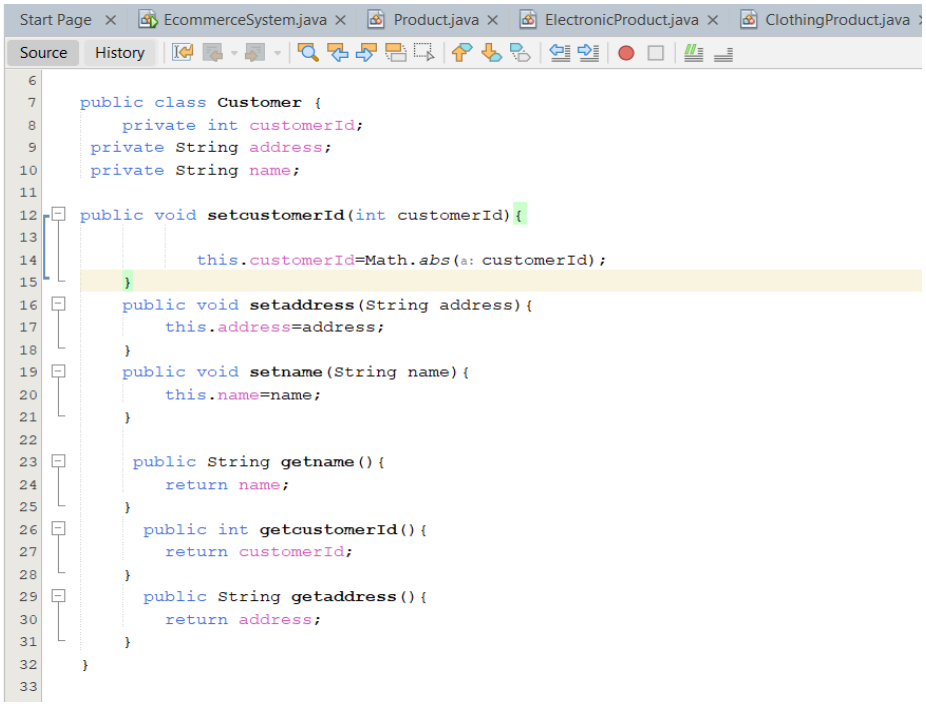
```java


public class ClothingProduct extends Product {
    private String size;
   private String fabric;
   public ClothingProduct(){

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

    public void setsize(String size){
        this.size=size;
    }
    public void setfabric(String fabric){
        this.fabric=fabric;
    }
    public String getsize(){
        return size;
    }
    public String getfabric(){
        return fabric;
    }
}

```

book product class:



```java

public class BookProduct extends Product {
    private String author;
    private String publisher;
    public BookProduct(){

    }
    public BookProduct(int productId,String name,double price,String author,String publisher){
        super(productId,name,price);
        this.author=author;
         this.publisher=publisher;
    }
    public void setauthor(String author){
        this.author=author;
    }
    public void setpublisher(String publisher){
        this.publisher=publisher;
    }
    public String getauthor(){
        return author;
    }
    public String getpublisher(){
        return publisher;
    }
}
```

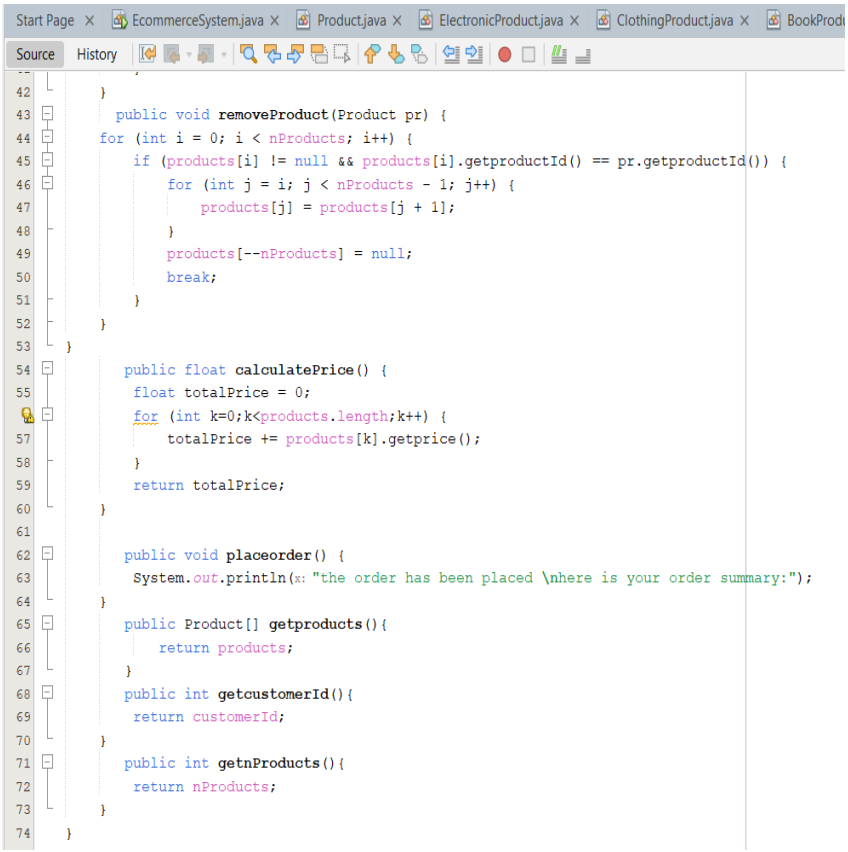
customer class:



```java

public class Customer {
    private int customerId;
 private String address;
 private String name;

public void setcustomerId(int customerId){

            this.customerId=Math.abs(a: customerId);
    }
    public void setaddress(String address){
        this.address=address;
    }
    public void setname(String name){
        this.name=name;
    }

     public String getname(){
        return name;
    }
      public int getcustomerId(){
        return customerId;
    }
      public String getaddress(){
        return address;
    }
}

```

Cart class:

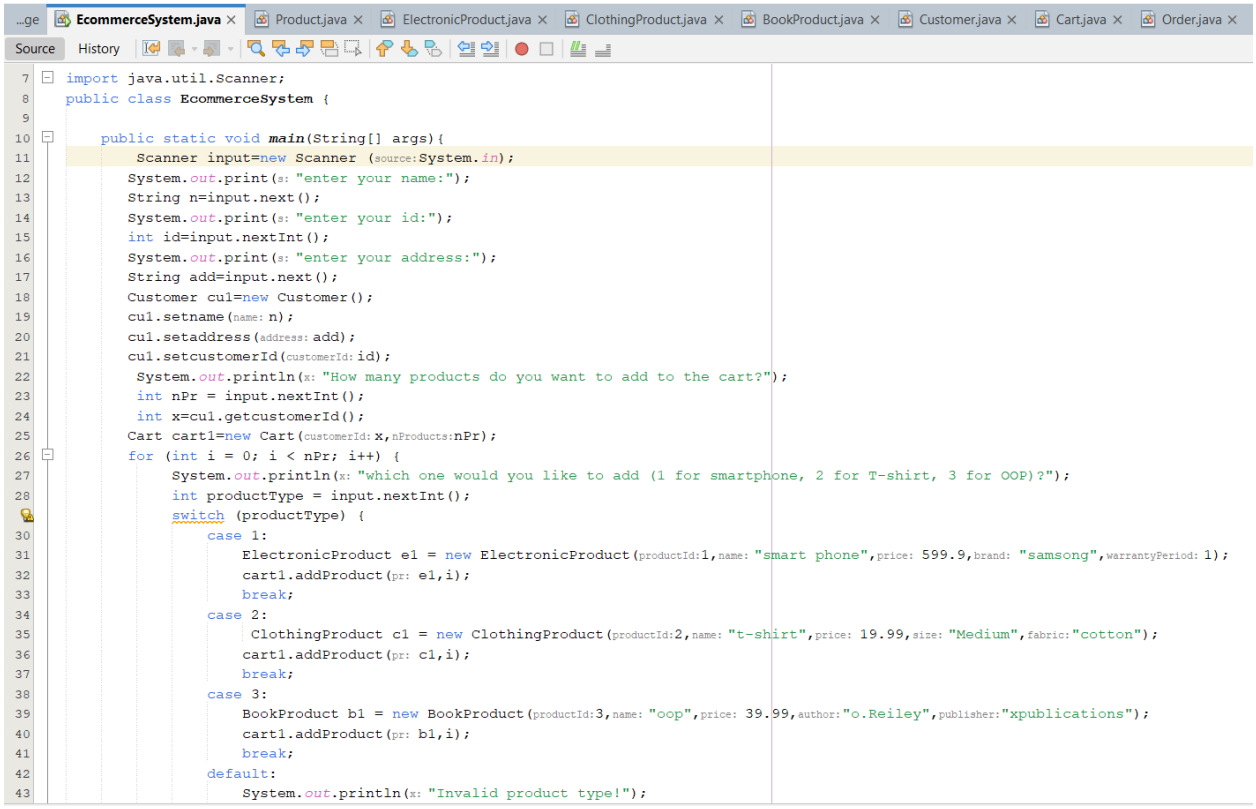
```java
import java.util.Scanner;
public class Cart {
   private int nProducts;
    private int customerId;
    private Product[] products=new Product [nProducts];
    public Cart(){

    }

    public Cart(int customerId,int nProducts ){
        this.nProducts=Math.abs(a: nProducts);
        this.customerId=Math.abs(a: customerId);
        this.products=new Product[nProducts];
    }
     public void setcustomerId(int customerId){

            this.customerId=Math.abs(a: customerId);
    }
     public void setnProducts(int nProducts){
        if (nProducts<=0){
            this.nProducts=Math.abs(a: nProducts);
        }
        else{
            this.nProducts=nProducts;
        }

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


      public void addProduct(Product pr,int i) {
          if (i < products.length) {
            products[i] = pr;
        }
    }
```



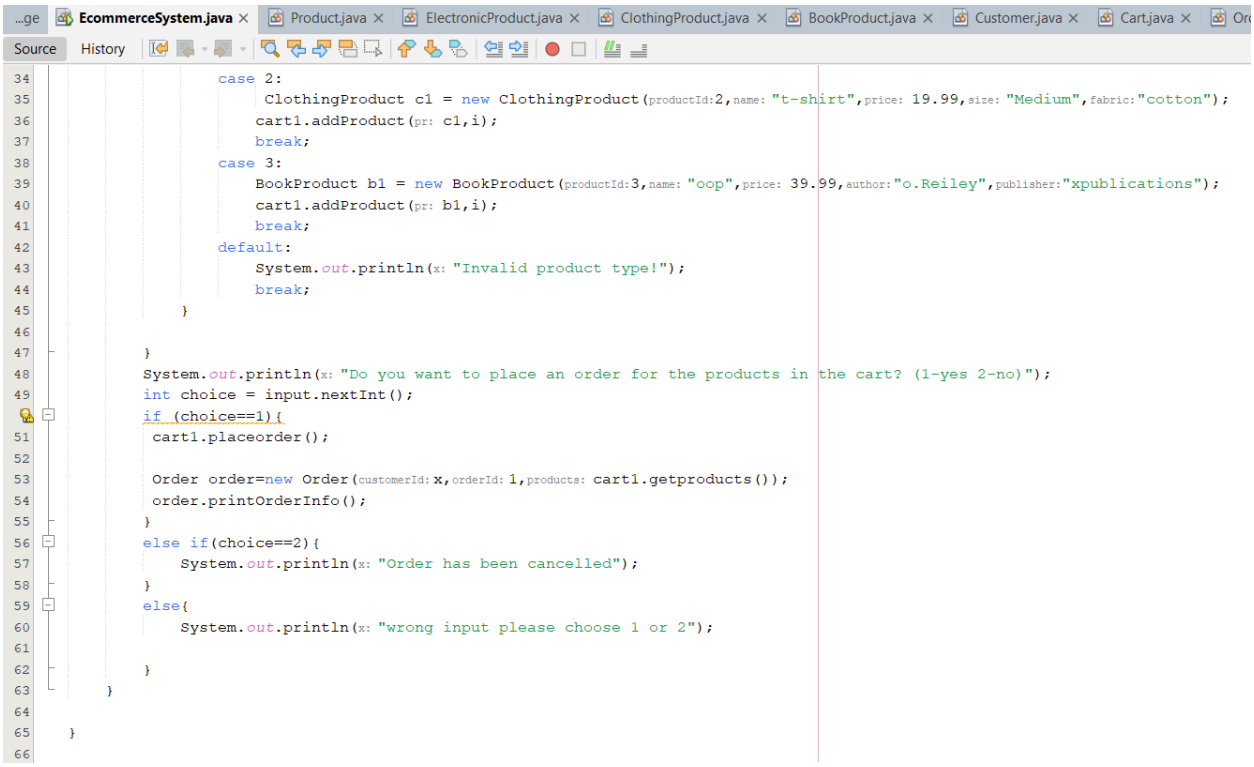
Order class:

```
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:



```
import java.util.Scanner;
public class EcommerceSystem {

    public static void main(String[] args){
        Scanner input=new Scanner (source:System.in);
       System.out.print(s: "enter your name:");
       String n=input.next();
       System.out.print(s: "enter your id:");
       int id=input.nextInt();
       System.out.print(s: "enter your address:");
       String add=input.next();
       Customer cu1=new Customer();
       cu1.setname(name: n);
       cu1.setaddress(address: add);
       cu1.setcustomerId(customerId: id);
        System.out.println(x: "How many products do you want to add to the cart?");
        int nPr = input.nextInt();
        int x=cu1.getcustomerId();
       Cart cart1=new Cart(customerId: x,nProducts:nPr);
       for (int i = 0; i < nPr; i++) {
            System.out.println(x: "which one would you like to add (1 for smartphone, 2 for T-shirt, 3 for OOP)?");
            int productType = input.nextInt();
            switch (productType) {
                case 1:
                    ElectronicProduct e1 = new ElectronicProduct(productId:1,name: "smart phone",price: 599.9,brand: "samsong",warrantyPeriod: 1);
                    cart1.addProduct(pr: e1,i);
                    break;
                case 2:
                     ClothingProduct c1 = new ClothingProduct(productId:2,name: "t-shirt",price: 19.99,size: "Medium",fabric:"cotton");
                    cart1.addProduct(pr: c1,i);
                    break;
                case 3:
                    BookProduct b1 = new BookProduct(productId:3,name: "oop",price: 39.99,author:"o.Reiley",publisher:"xpublications");
                    cart1.addProduct(pr: b1,i);
                    break;
                default:
                    System.out.println(x: "Invalid product type!");
```



```
                case 2:
                     ClothingProduct c1 = new ClothingProduct(productId:2,name: "t-shirt",price: 19.99,size: "Medium",fabric:"cotton");
                    cart1.addProduct(pr: c1,i);
                    break;
                case 3:
                    BookProduct b1 = new BookProduct(productId:3,name: "oop",price: 39.99,author:"o.Reiley",publisher:"xpublications");
                    cart1.addProduct(pr: b1,i);
                    break;
                default:
                    System.out.println(x: "Invalid product type!");
                    break;
            }

        }
        System.out.println(x: "Do you want to place an order for the products in the cart? (1-yes 2-no)");
        int choice = input.nextInt();
        if (choice==1){
         cart1.placeorder();

         Order order=new Order(customerId: x,orderId: 1,products: cart1.getproducts());
         order.printOrderInfo();
        }
        else if(choice==2){
            System.out.println(x: "Order has been cancelled");
        }
        else{
            System.out.println(x: "wrong input please choose 1 or 2");

        }
    }

}
```

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)
```