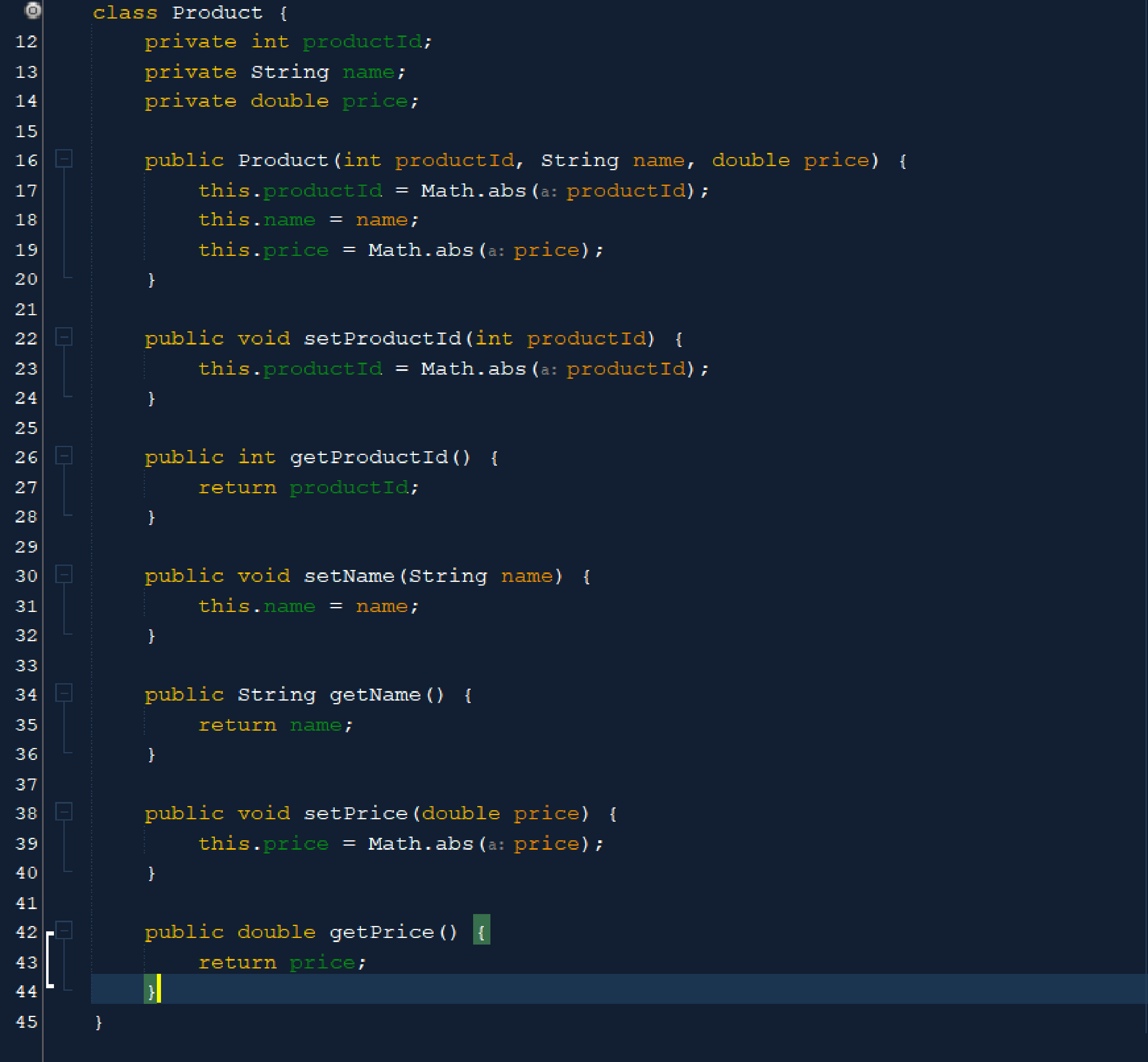
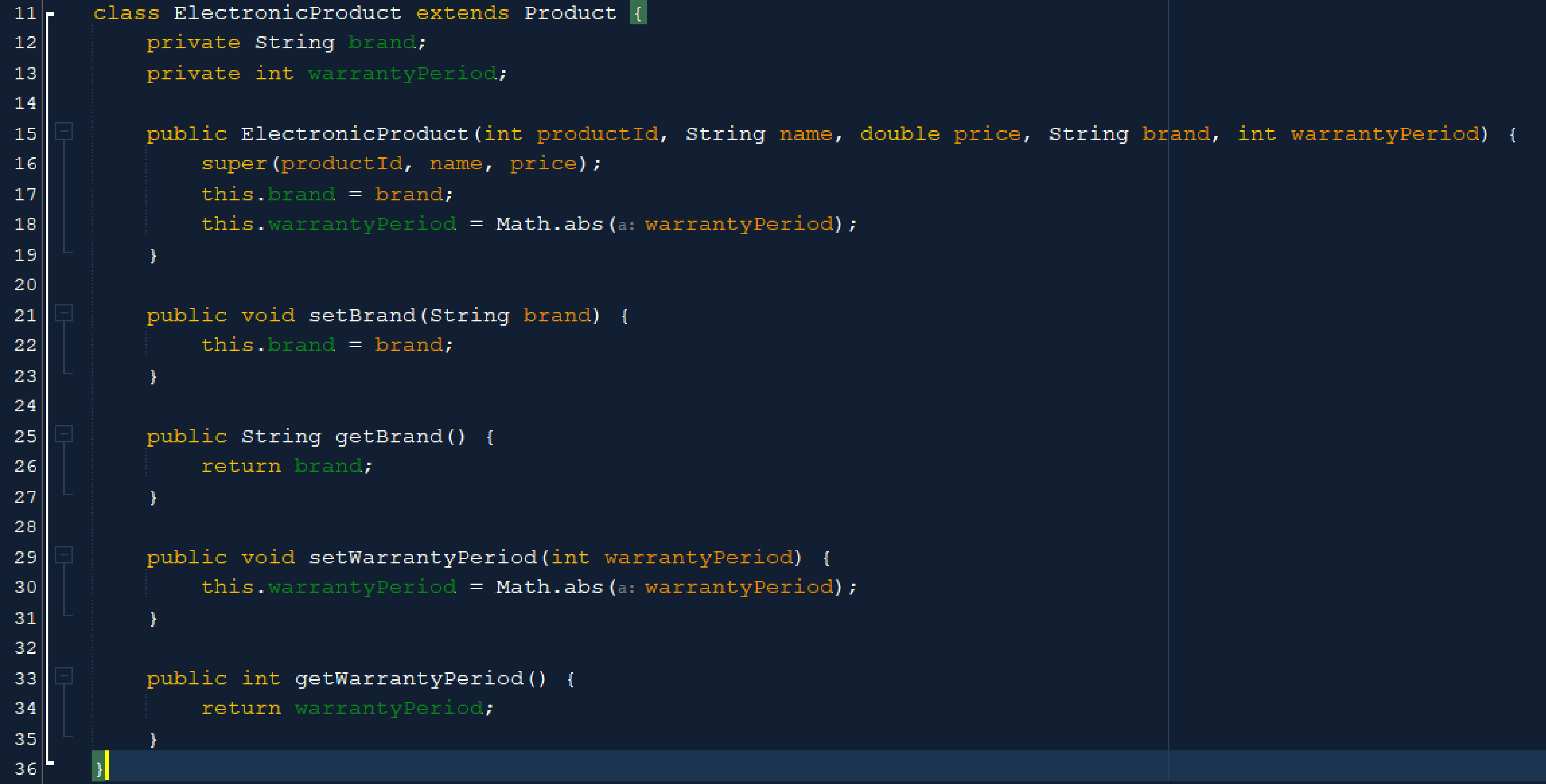
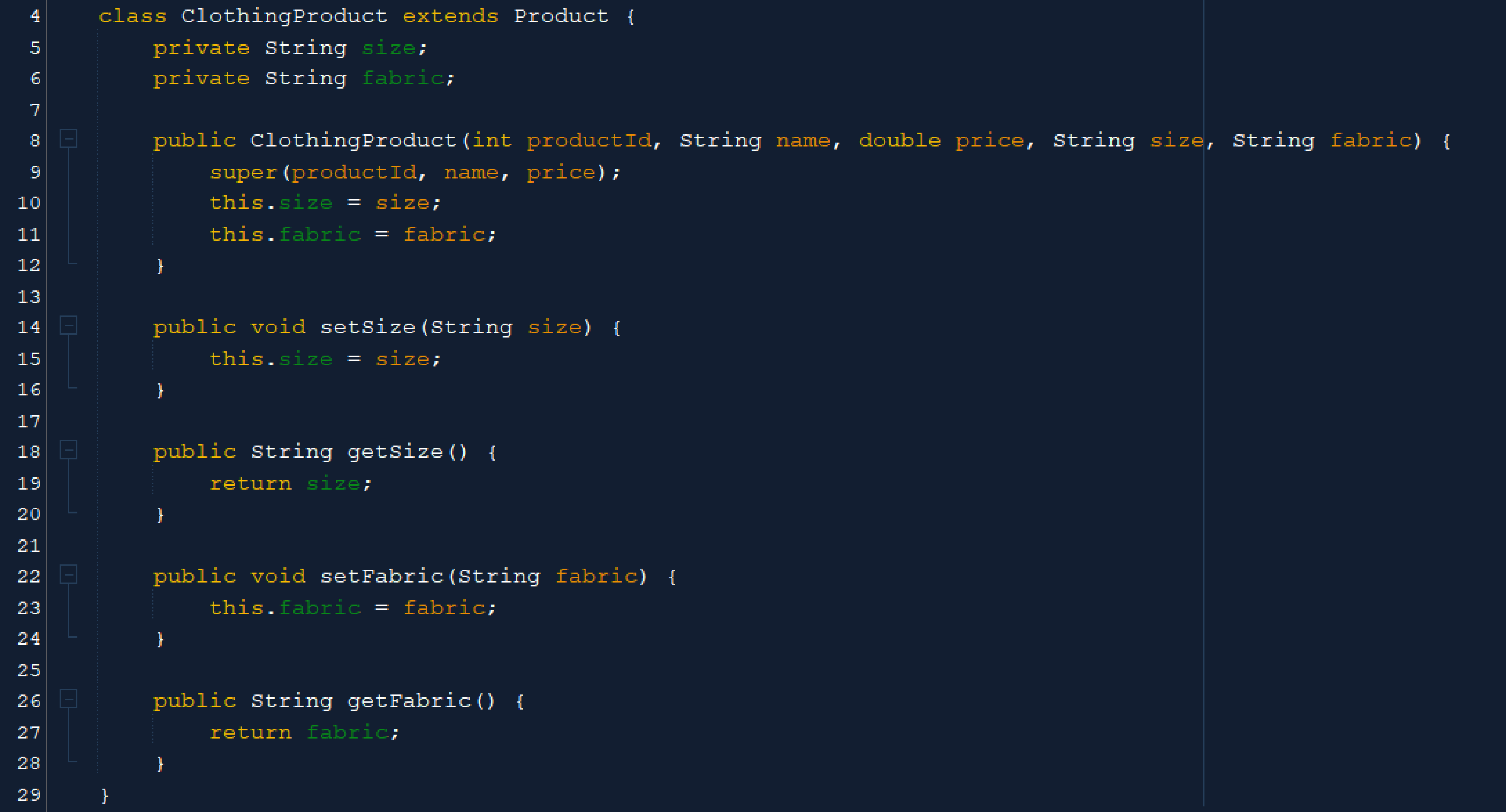


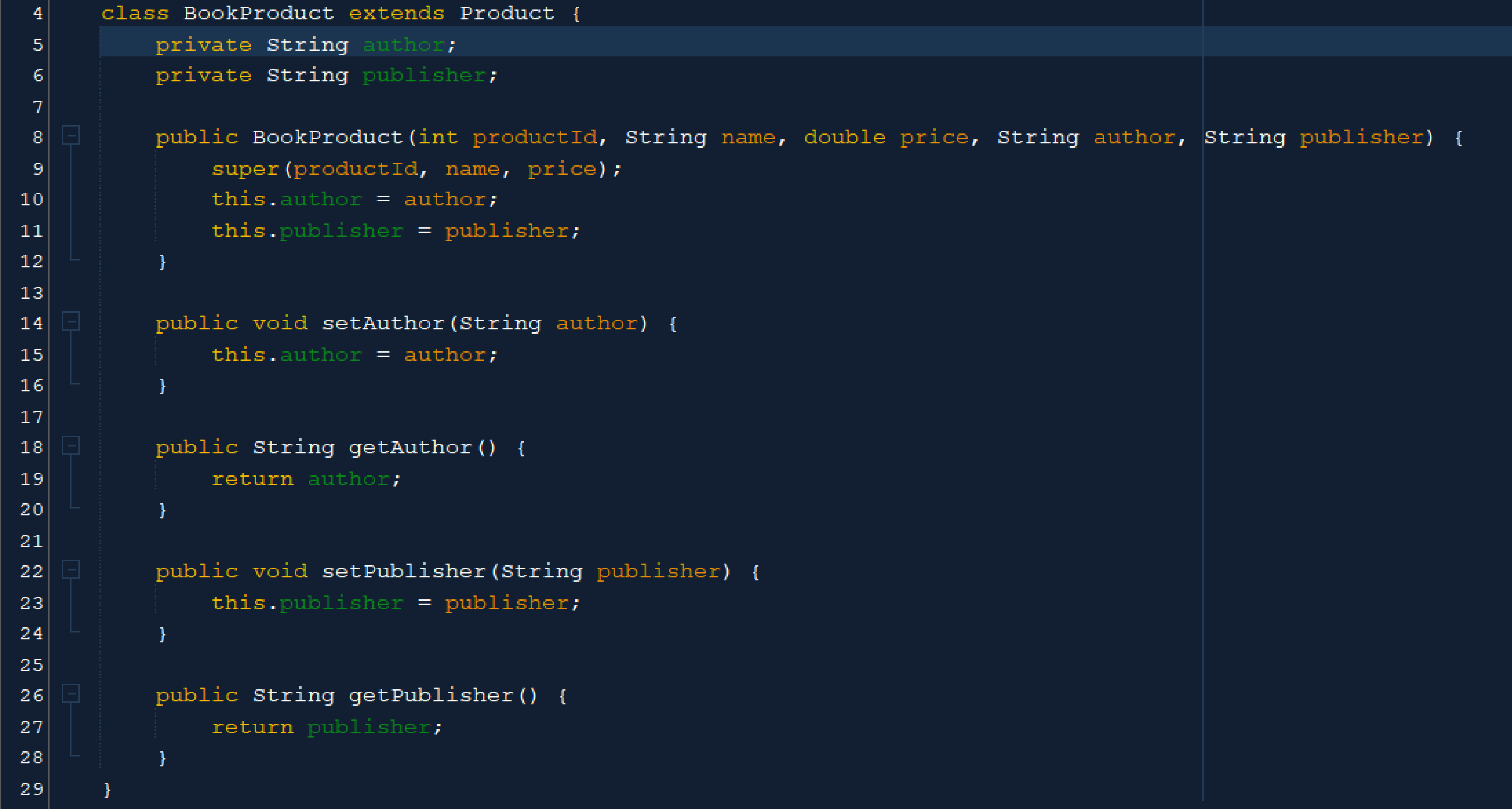
```java
class Product {
    private int productId;
    private String name;
    private double price;

    public Product(int productId, String name, double price) {
        this.productId = Math.abs(a: productId);
        this.name = name;
        this.price = Math.abs(a: price);
    }

    public void setProductId(int productId) {
        this.productId = Math.abs(a: productId);
    }

    public int getProductId() {
        return productId;
    }

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

    public String getName() {
        return name;
    }

    public void setPrice(double price) {
        this.price = Math.abs(a: price);
    }

    public double getPrice() {
        return price;
    }
}
```



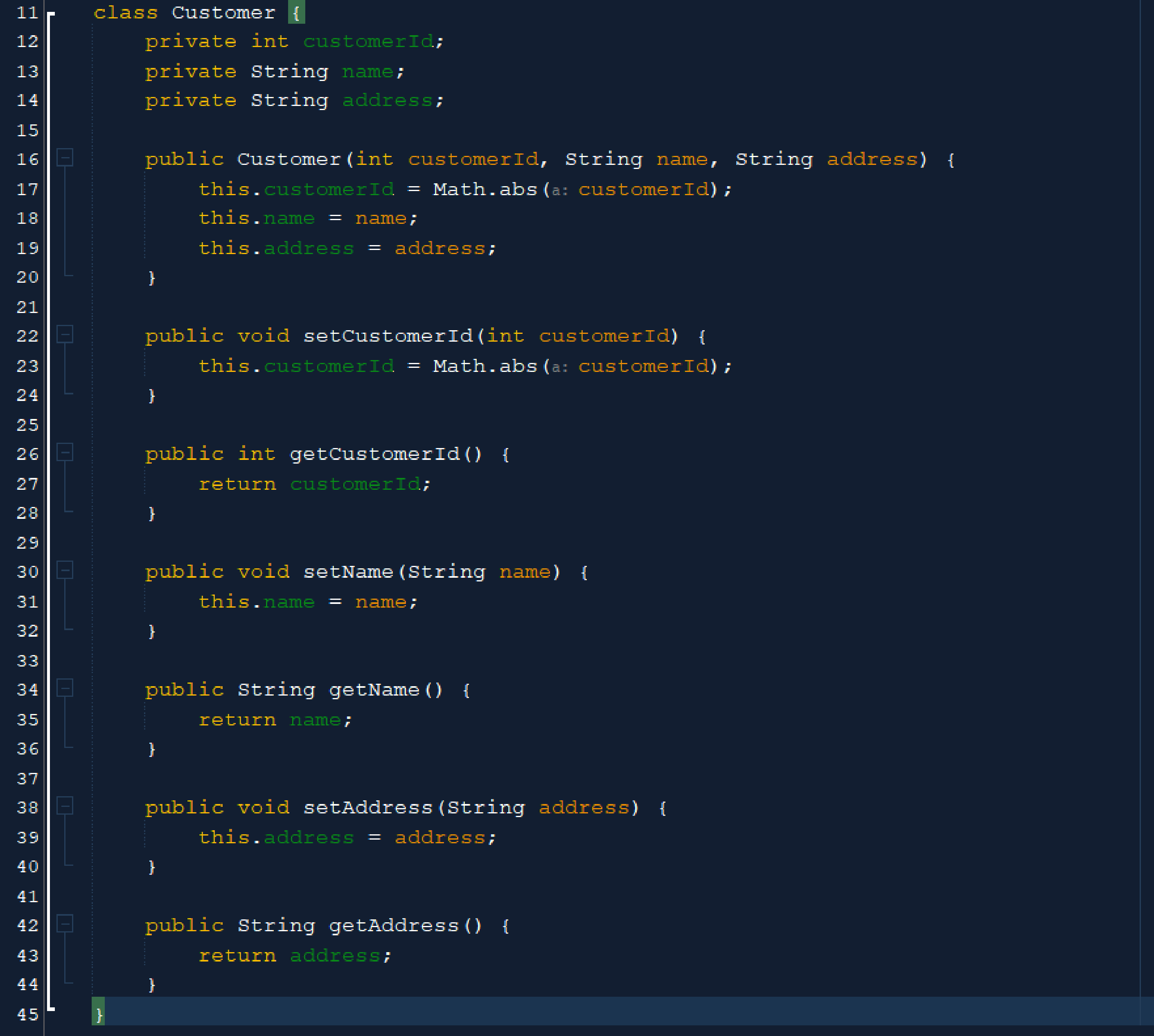
```java
class ElectronicProduct extends Product {
    private String brand;
    private int warrantyPeriod;

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

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

    public String getBrand() {
        return brand;
    }

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

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



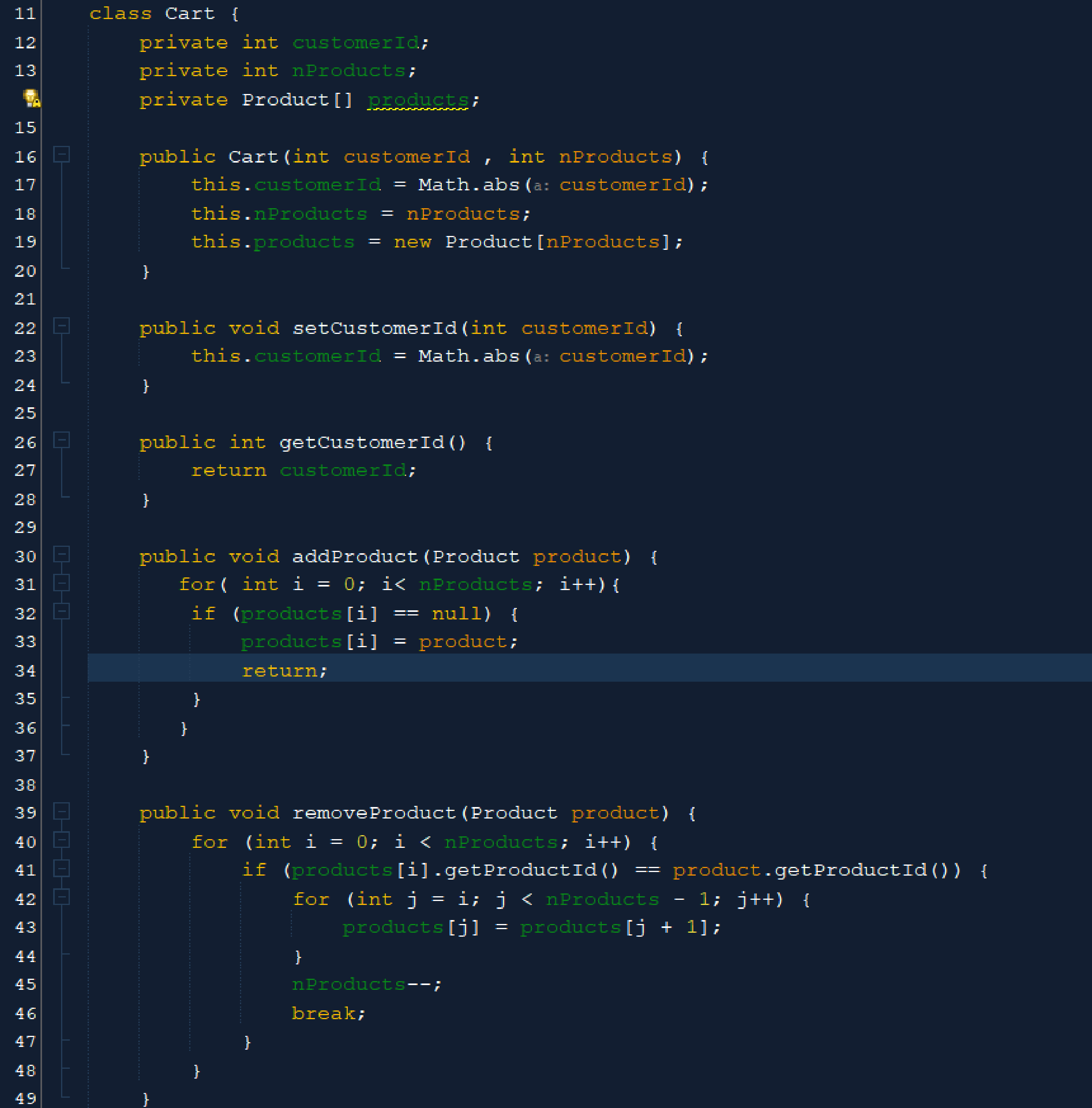
```java
class ClothingProduct extends Product {
    private String size;
    private String fabric;

    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 String getSize() {
        return size;
    }

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

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



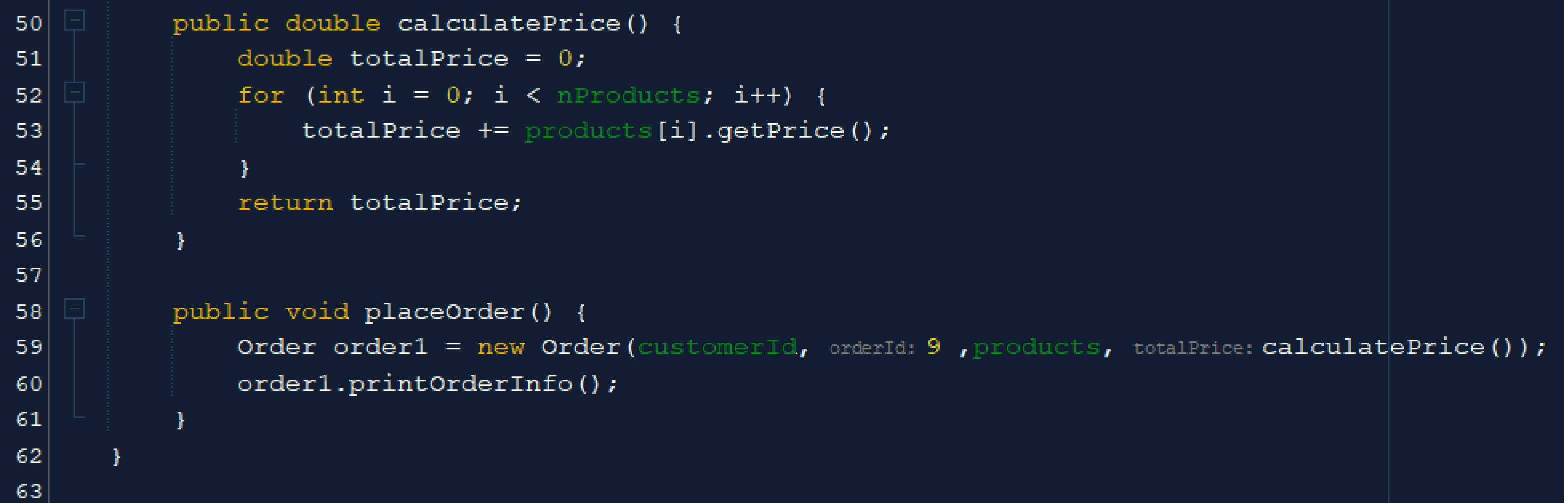
```java
class BookProduct extends Product {
    private String author;
    private String publisher;

    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 String getAuthor() {
        return author;
    }

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

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



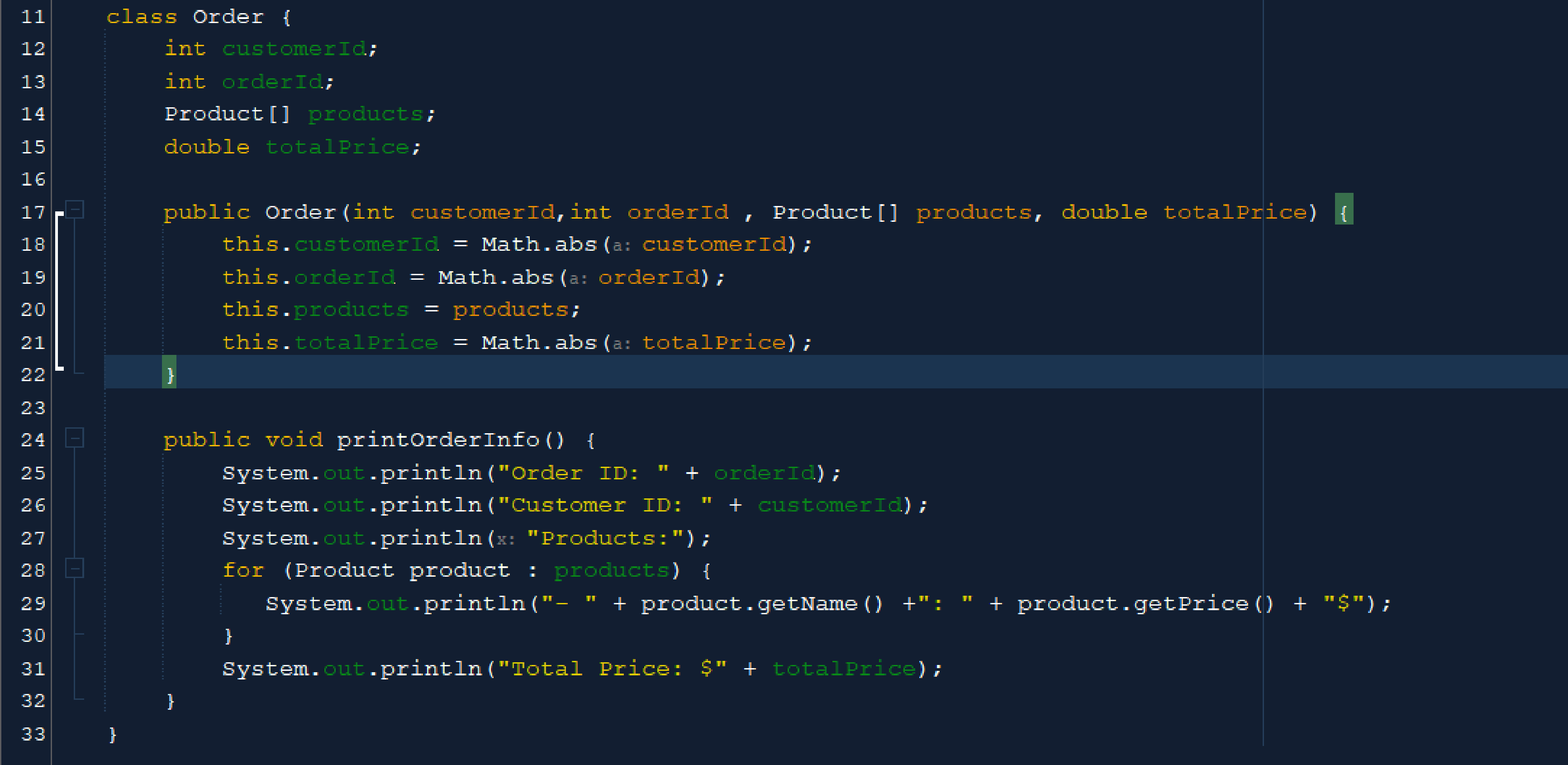
```java
class Customer {
    private int customerId;
    private String name;
    private String address;

    public Customer(int customerId, String name, String address) {
        this.customerId = Math.abs(a: customerId);
        this.name = name;
        this.address = address;
    }

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

    public int getCustomerId() {
        return customerId;
    }

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

    public String getName() {
        return name;
    }

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

    public String getAddress() {
        return address;
    }
}
```



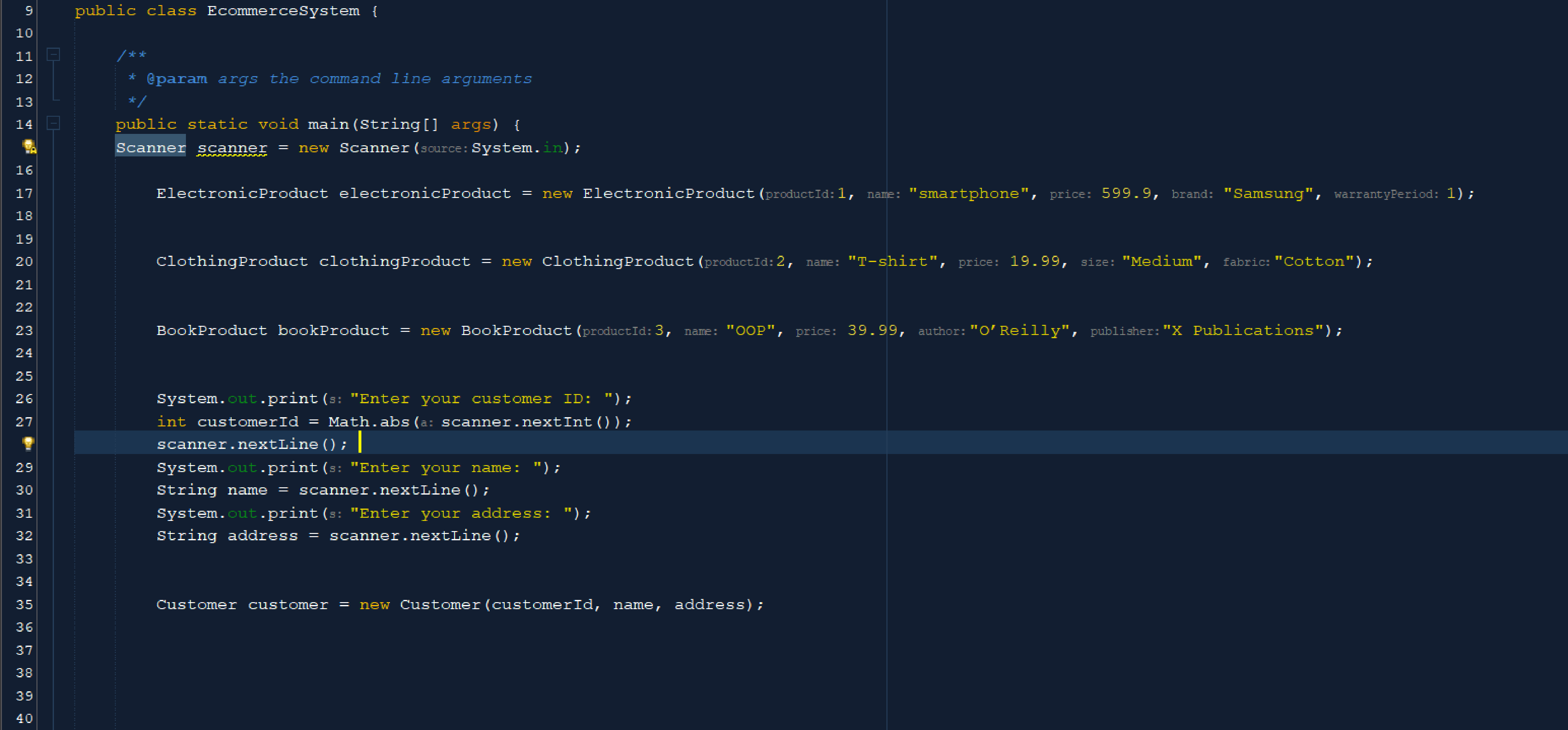
```java
class Cart {
    private int customerId;
    private int nProducts;
    private Product[] products;

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

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

    public int getCustomerId() {
        return customerId;
    }

    public void addProduct(Product product) {
       for( int i = 0; i< nProducts; i++){
        if (products[i] == null) {
            products[i] = product;
            return;
        }
       }
    }

    public void removeProduct(Product product) {
        for (int i = 0; i < nProducts; i++) {
            if (products[i].getProductId() == product.getProductId()) {
                for (int j = i; j < nProducts - 1; j++) {
                    products[j] = products[j + 1];
                }
                nProducts--;
                break;
            }
        }
    }
```



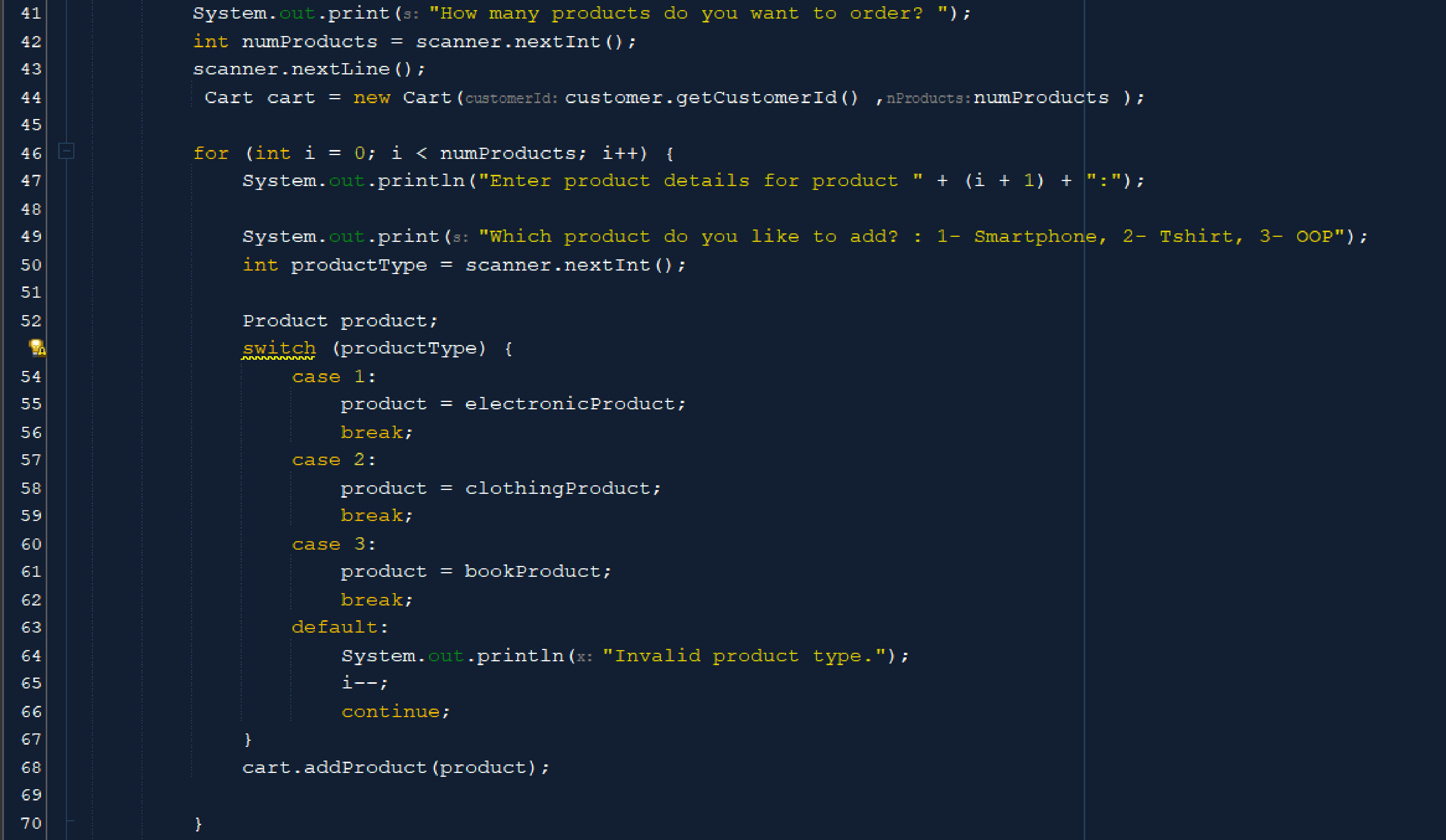
```java
    public double calculatePrice() {
        double totalPrice = 0;
        for (int i = 0; i < nProducts; i++) {
            totalPrice += products[i].getPrice();
        }
        return totalPrice;
    }

    public void placeOrder() {
        Order order1 = new Order(customerId, orderId: 9 ,products, totalPrice: calculatePrice());
        order1.printOrderInfo();
    }
}

```



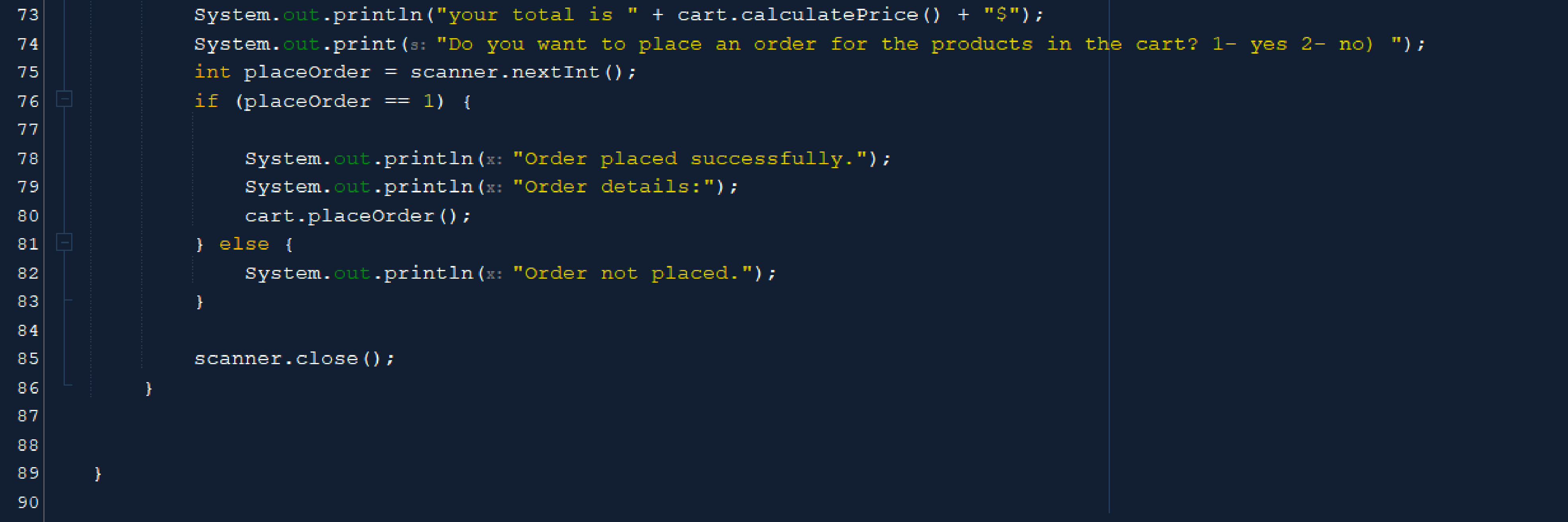
```java
class Order {
    int customerId;
    int orderId;
    Product[] products;
    double totalPrice;

    public Order(int customerId,int orderId , Product[] products, double totalPrice) {
        this.customerId = Math.abs(a: customerId);
        this.orderId = Math.abs(a: orderId);
        this.products = products;
        this.totalPrice = Math.abs(a: totalPrice);
    }

    public void printOrderInfo() {
        System.out.println("Order ID: " + orderId);
        System.out.println("Customer ID: " + customerId);
        System.out.println(x: "Products:");
        for (Product product : products) {
            System.out.println("- " + product.getName() +": " + product.getPrice() + "$");
        }
        System.out.println("Total Price: $" + totalPrice);
    }
}
```



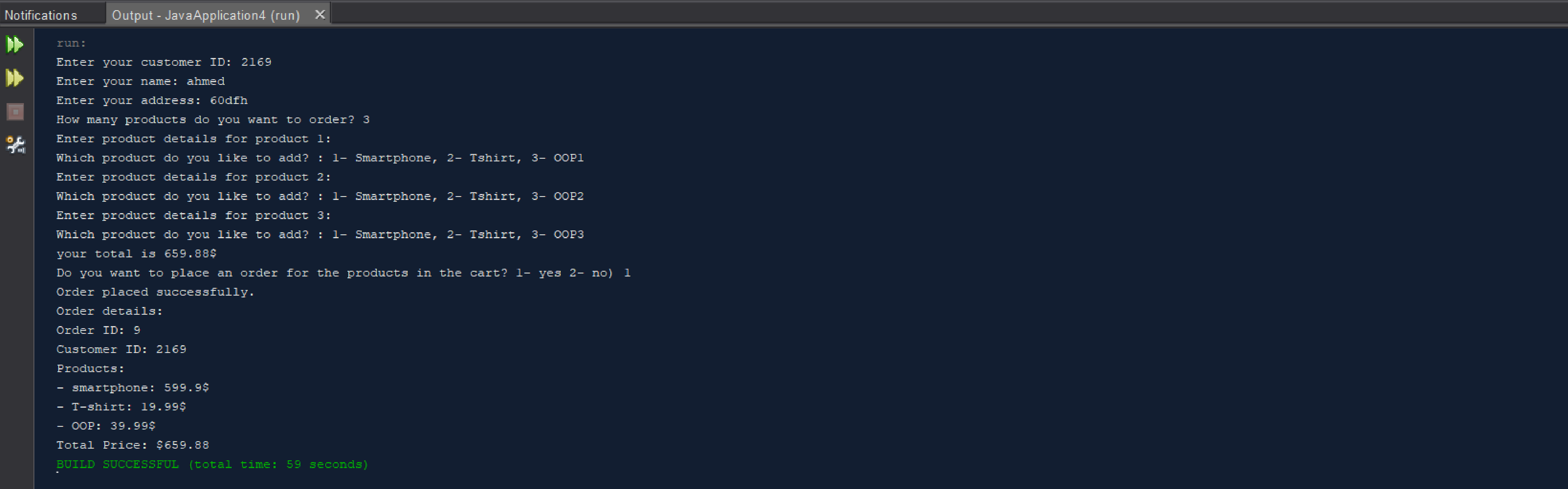
```
public class EcommerceSystem {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
    Scanner scanner = new Scanner(source: System.in);

        ElectronicProduct electronicProduct = new ElectronicProduct(productId: 1, name: "smartphone", price: 599.9, brand: "Samsung", warrantyPeriod: 1);


        ClothingProduct clothingProduct = new ClothingProduct(productId: 2, name: "T-shirt", price: 19.99, size: "Medium", fabric: "Cotton");


        BookProduct bookProduct = new BookProduct(productId: 3, name: "OOP", price: 39.99, author: "O’Reilly", publisher: "X Publications");


        System.out.print(s: "Enter your customer ID: ");
        int customerId = Math.abs(a: scanner.nextInt());
        scanner.nextLine();
        System.out.print(s: "Enter your name: ");
        String name = scanner.nextLine();
        System.out.print(s: "Enter your address: ");
        String address = scanner.nextLine();


        Customer customer = new Customer(customerId, name, address);





```



```
        System.out.print(s: "How many products do you want to order? ");
        int numProducts = scanner.nextInt();
        scanner.nextLine();
         Cart cart = new Cart(customerId: customer.getCustomerId() ,nProducts:numProducts );

        for (int i = 0; i < numProducts; i++) {
            System.out.println("Enter product details for product " + (i + 1) + ":");

            System.out.print(s: "Which product do you like to add? : 1- Smartphone, 2- Tshirt, 3- OOP");
            int productType = scanner.nextInt();

            Product product;
            switch (productType) {
                case 1:
                    product = electronicProduct;
                    break;
                case 2:
                    product = clothingProduct;
                    break;
                case 3:
                    product = bookProduct;
                    break;
                default:
                    System.out.println(x: "Invalid product type.");
                    i--;
                    continue;
            }
            cart.addProduct(product);

        }
```



```
            System.out.println("your total is " + cart.calculatePrice() + "$");
            System.out.print(s: "Do you want to place an order for the products in the cart? 1- yes 2- no) ");
            int placeOrder = scanner.nextInt();
            if (placeOrder == 1) {

                System.out.println(x: "Order placed successfully.");
                System.out.println(x: "Order details:");
                cart.placeOrder();
            } else {
                System.out.println(x: "Order not placed.");
            }

            scanner.close();
        }


    }

```



```
run:
Enter your customer ID: 2169
Enter your name: ahmed
Enter your address: 60dfh
How many products do you want to order? 3
Enter product details for product 1:
Which product do you like to add? : 1- Smartphone, 2- Tshirt, 3- OOP1
Enter product details for product 2:
Which product do you like to add? : 1- Smartphone, 2- Tshirt, 3- OOP2
Enter product details for product 3:
Which product do you like to add? : 1- Smartphone, 2- Tshirt, 3- OOP3
your total is 659.88$
Do you want to place an order for the products in the cart? 1- yes 2- no) 1
Order placed successfully.
Order details:
Order ID: 9
Customer ID: 2169
Products:
- smartphone: 599.9$
- T-shirt: 19.99$
- OOP: 39.99$
Total Price: $659.88
BUILD SUCCESSFUL (total time: 59 seconds)
```