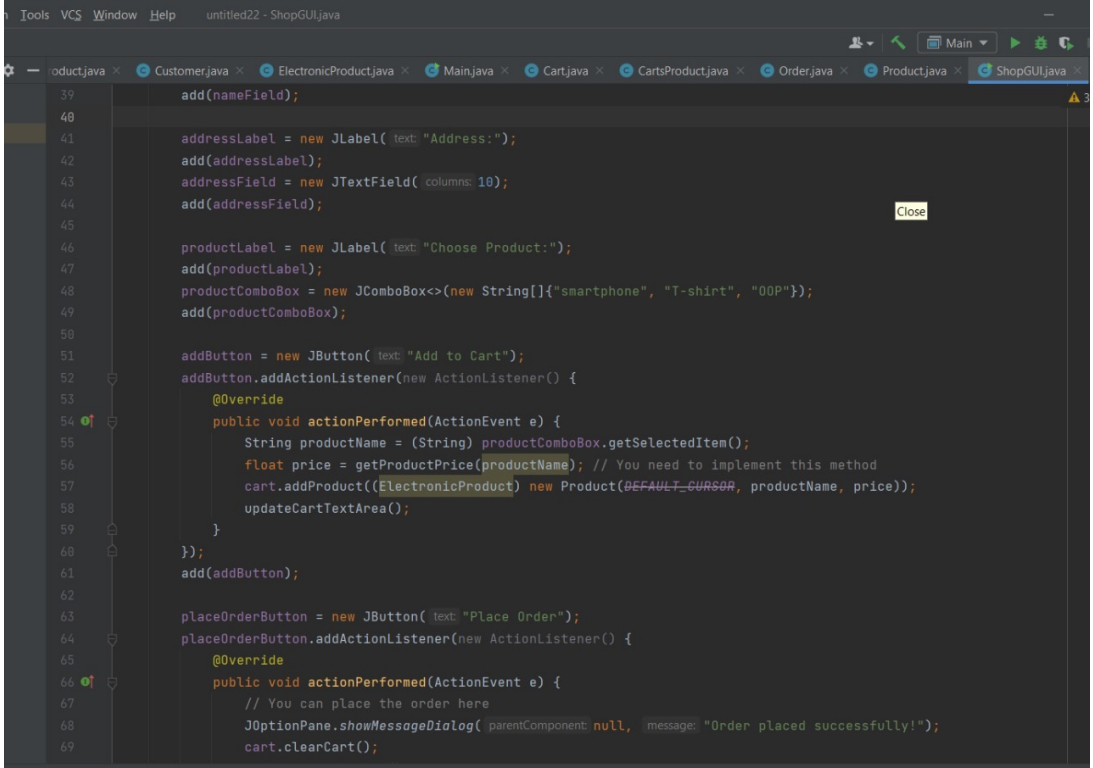
8 usages 2 inheritors

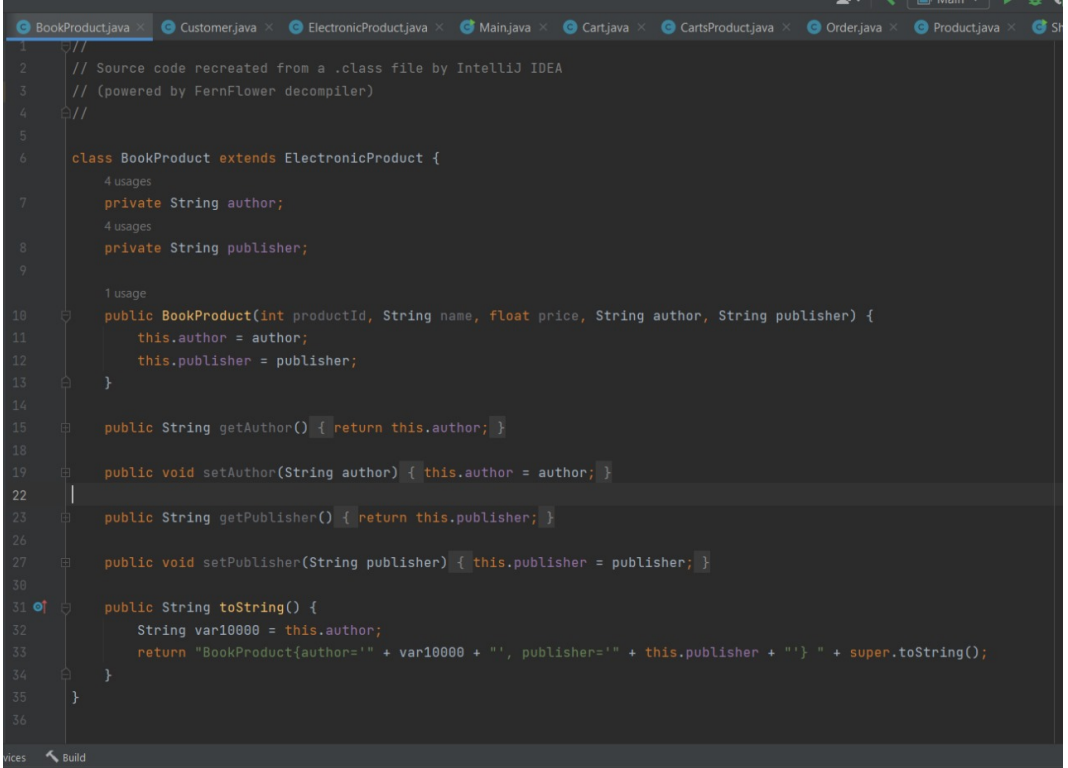
```
class Product {
    4 usages
    private int productId;
    3 usages
    private String name;
    3 usages
    private float price;

    4 usages
    public Product(int id, String name, float price) {
        this.productId = Math.abs(productId);
        this.name = name;
        this.price = Math.abs(price);
    }

    public int getProductId() { return this.productId; }

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

    1 usage
    public String getName() { return this.name; }

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

    3 usages
    public float getPrice() { return this.price; }

    public void setPrice(float price) { this.price = Math.abs(price); }
}
```

Close

```
13 usages   1 inheritor
public class ElectronicProduct extends Product {
    1 usage
    public ElectronicProduct(int id, String name, float price, String brand, int quantity) {
        super(id, name, price);
        // Initialize any additional fields specific to ElectronicProduct
    }

    1 usage
    public ElectronicProduct(int id, String name, float price, String size, String material) {
        super(id, name, price);
        // Initialize any additional fields specific to ElectronicProduct
    }

    1 usage
    public ElectronicProduct() {
        super( id: 0,  name: "",  price: 0.0f); // You may need to adjust the default values here
        // Initialize any additional fields specific to ElectronicProduct
    }
}
```


Tools VCS Window Help untitled22 - ShopGUI.java
Main
oduct.java Customer.java ElectronicProduct.java Main.java Cart.java CartsProduct.java Order.java Product.java ShopGUI.java
add(nameField);
addressLabel = new JLabel(text: "Address:");
add(addressLabel);
addressField = new JTextField(columns: 10);
add(addressField);
productLabel = new JLabel(text: "Choose Product:");
add(productLabel);
productComboBox = new JComboBox<>(new String[]{"smartphone", "T-shirt", "OOP"});
add(productComboBox);
addButton = new JButton(text: "Add to Cart");
addButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
String productName = (String) productComboBox.getSelectedItem();
float price = getProductPrice(productName); // You need to implement this method
cart.addProduct((ElectronicProduct) new Product(DEFAULT_CURSOR, productName, price));
updateCartTextArea();
}
});
add(addButton);
placeOrderButton = new JButton(text: "Place Order");
placeOrderButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
// You can place the order here
JOptionPane.showMessageDialog(parentComponent: null, message: "Order placed successfully!");
cart.clearCart();
Close

```
/.../

import java.util.Arrays;

4 usages
class Cart {
    3 usages
    private int customerId;
    9 usages
    private int nProducts;
    10 usages
    private ElectronicProduct[] products;
    2 usages
    int i = 0;

    2 usages
    public Cart(int customerId, int nProducts) {
        this.customerId = Math.abs(customerId);
        this.nProducts = Math.abs(nProducts);
        this.products = new ElectronicProduct[nProducts];
    }

    public int getCustomerId() { return this.customerId; }

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

    public int getnProducts() { return this.nProducts; }

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

    1 usage
```


BookProduct.java
Customer.java
ElectronicProduct.java
Main.java
Cart.java
CartsProduct.java
Order.java
Product.java
//
// Source code recreated from a .class file by IntelliJ IDEA
// (powered by FernFlower decompiler)
//
class BookProduct extends ElectronicProduct {
4 usages
private String author;
4 usages
private String publisher;
1 usage
public BookProduct(int productId, String name, float price, String author, String publisher) {
this.author = author;
this.publisher = publisher;
}
public String getAuthor() { return this.author; }
public void setAuthor(String author) { this.author = author; }
public String getPublisher() { return this.publisher; }
public void setPublisher(String publisher) { this.publisher = publisher; }
public String toString() {
String var10000 = this.author;
return "BookProduct{author='" + var10000 + "', publisher='" + this.publisher + "'} " + super.toString();
}
}
vices
Build

```
/.../

import java.util.Scanner;

public class Main {
    public Main() {
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        ElectronicProduct electronicProduct = new ElectronicProduct( id: 1,  name: "smartphone",  price: 599.9F,  brand: "Samsu
        ElectronicProduct clothingProduct = new ElectronicProduct( id: 2,  name: "T-shirt",  price: 19.99F,  size: "Medium",  ma
        BookProduct bookProduct = new BookProduct( productId: 3,  name: "OOP",  price: 39.99F,  author: "O'Reilly",  publisher: "X Pub
        System.out.println("Welcome to our shop!");
        System.out.println("Enter your ID:");
        int id = scanner.nextInt();
        System.out.println("Enter your name:");
        String name = scanner.next();
        System.out.println("Enter your address:");
        String address = scanner.next();
        new Customer(id, name, address);
        System.out.println("How many products do you want?");
        int n = scanner.nextInt();
        Cart cart = new Cart(id, n);

        int decision;
        for (int i = 0; i < n; ++i) {
            System.out.println("Which product would you like? (1 for smartphone, 2 for T-shirt, 3 for OOP)");
            decision = scanner.nextInt();
            switch (decision) {
                case 1:
```

Build

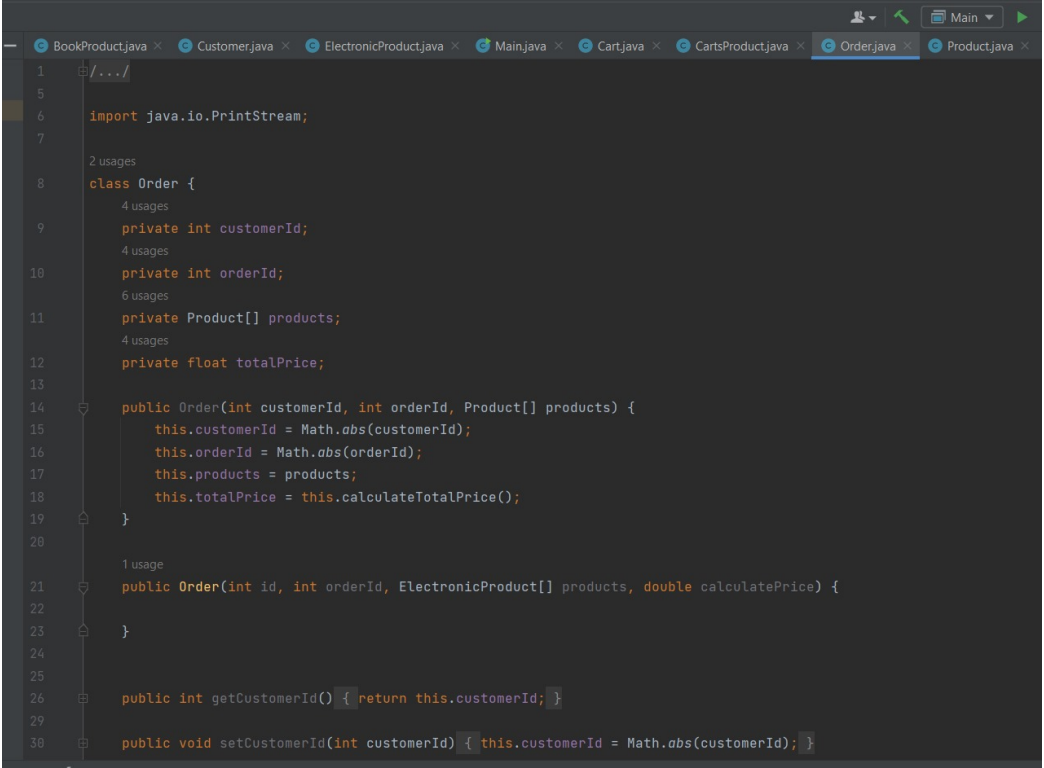
31:77 LF UT

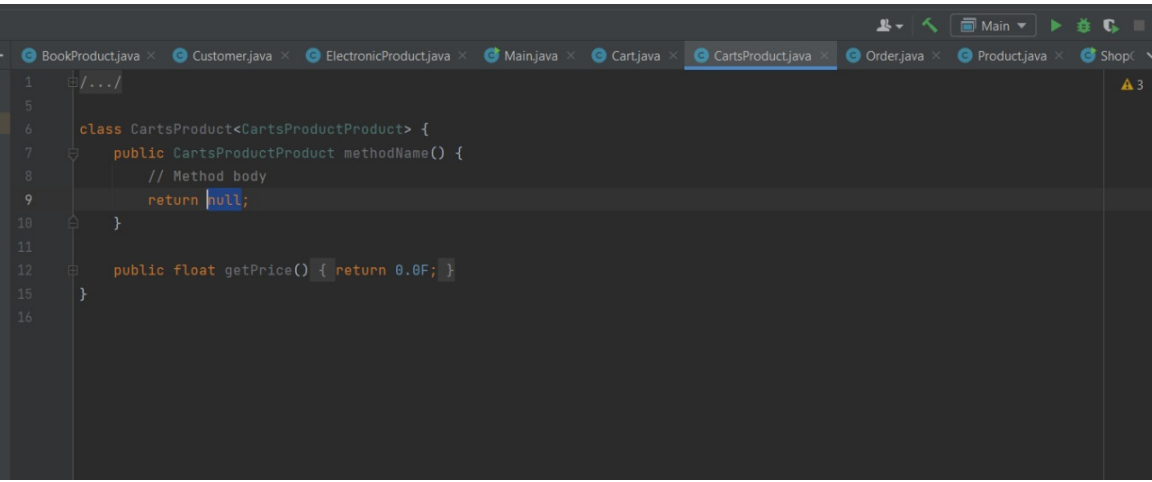
```
/.../

import java.util.Scanner;

public class Main {
    public Main() {
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        ElectronicProduct electronicProduct = new ElectronicProduct( id: 1,  name: "smartphone",  price: 599.9F,  brand: "Samsu
        ElectronicProduct clothingProduct = new ElectronicProduct( id: 2,  name: "T-shirt",  price: 19.99F,  size: "Medium",  ma
        BookProduct bookProduct = new BookProduct( productId: 3,  name: "OOP",  price: 39.99F,  author: "O'Reilly",  publisher: "X Pub
        System.out.println("Welcome to our shop!");
        System.out.println("Enter your ID:");
        int id = scanner.nextInt();
        System.out.println("Enter your name:");
        String name = scanner.next();
        System.out.println("Enter your address:");
        String address = scanner.next();
        new Customer(id, name, address);
        System.out.println("How many products do you want?");
        int n = scanner.nextInt();
        Cart cart = new Cart(id, n);

        int decision;
        for (int i = 0; i < n; ++i) {
            System.out.println("Which product would you like? (1 for smartphone, 2 for T-shirt, 3 for OOP)");
            decision = scanner.nextInt();
            switch (decision) {
                case 1:
```

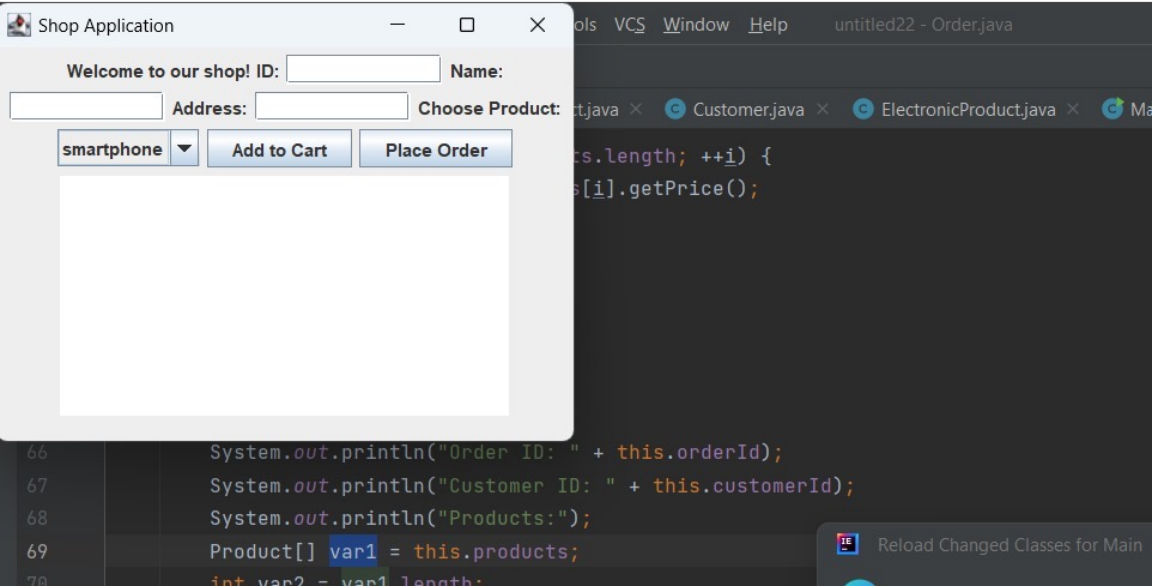
Build

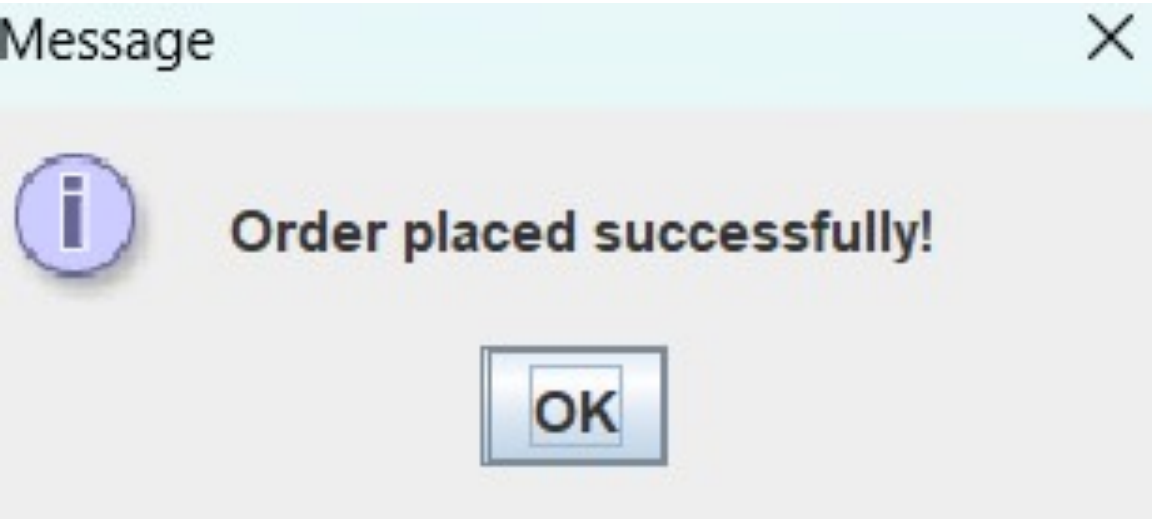
31:77 LF UT

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

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

    public int getCustomerId() { return this.customerId; }

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

    public String getName() { return this.name; }

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

    public String getAddress() { return this.address; }

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

    public String toString() {
        return "Customer{customerId=" + this.customerId + ", name='" + this.name + "', address='" + this.address + "'}";
    }
```


Main
BookProduct.java
Customer.java
ElectronicProduct.java
Main.java
Cart.java
CartsProduct.java
Order.java
Product.java
/.../
import java.io.PrintStream;
2 usages
class Order {
4 usages
private int customerId;
4 usages
private int orderId;
6 usages
private Product[] products;
4 usages
private float totalPrice;
public Order(int customerId, int orderId, Product[] products) {
this.customerId = Math.abs(customerId);
this.orderId = Math.abs(orderId);
this.products = products;
this.totalPrice = this.calculateTotalPrice();
}
1 usage
public Order(int id, int orderId, ElectronicProduct[] products, double calculatePrice) {
}
public int getCustomerId() { return this.customerId; }
public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }


Main
BookProduct.java
Customer.java
ElectronicProduct.java
Main.java
Cart.java
CartsProduct.java
Order.java
Product.java
Shop
/.../
class CartsProduct<CartsProductProduct> {
public CartsProductProduct methodName() {
// Method body
return null;
}
public float getPrice() { return 0.0F; }
}
3


Shop Application
Welcome to our shop! ID:
Name:
Address:
Choose Product:
smartphone
Add to Cart
Place Order
ols VCS Window Help
untitled22 - Order.java
t.java
Customer.java
ElectronicProduct.java
Ma
s.length; ++i) {
[i].getPrice();
System.out.println("Order ID: " + this.orderId);
System.out.println("Customer ID: " + this.customerId);
System.out.println("Products:");
Product[] var1 = this.products;
Reload Changed Classes for Main


Message
Order placed successfully!
OK