```java
public class Product {
    private int productId;

    private String name;

    private float price;

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

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

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

    public int getProductId() { return productId; }

    public String getName() { return name; }

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

```java
public class ElectronicProduct extends Product{
    private String brand;

    private int warrantyPeriod;

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

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

    public String getBrand() { return brand; }

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

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

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

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

    no usages
    public String getSize() {
        return size;
    }

    no usages
    public String getFabric() {
        return fabric;
    }
}
```

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

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

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

    no usages
    public String getAuthor() { return author; }

    no usages
    public String getPublisher() { return publisher; }
}
```

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


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

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

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

    no usages
    public int getCustomerId() { return customerId; }

    no usages
    public String getName() { return name; }

    no usages
    public String getAddress() { return address; }
}
```

```java
public class Cart {
    2 usages
    private int customerId, nProducts;
    5 usages
    private Product[] products;
    1 usage
    public void setCustomerId(int customerId) { this.customerId = Math.abs(customerId); }
    1 usage
    public void setNProducts(int nProducts) {
        this.nProducts = Math.abs(nProducts);
        this.products = new Product[nProducts];
    }
    no usages
    public int getCustomerId() {
        return customerId;
    }
    no usages
    public int getNProducts() { return nProducts; }
    1 usage
    public Product[] getProducts() { return products; }
    1 usage
    public void addProduct(int index, Product product) {
        products[index] = product;
    }
    no usages
    public void removeProduct(int index) { products[index] = null; }
    1 usage
    public float calculatePrice() {
        float totalPrice = 0;
        for (Product product : products) {
            if (product != null) {
                totalPrice += product.getPrice();
            }
        }
        return totalPrice;
    }
    no usages
    public void placeOrder() {
        System.out.println("do you want to place order for the product in this cart? (yes or no)");
    }
}
```

Screenshot 1 — Order.java

```java
public class Order {
    private int customerId;
    private int orderId;
    private Product[] products;
    private float totalPrice;

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

    public void setOrderId(int orderId) { this.orderId = Math.abs(orderId); }

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

    // Getters
    public int getCustomerId() { return customerId; }

    public int getOrderId() { return orderId; }

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

    public void printOrderInfo() {
        System.out.println("Order id: " + orderId);
        System.out.println("Customer id: " + customerId);
        System.out.println("Products: ");
        for (Product product : products) {
            if (product != null) {
                System.out.println(product.getName() + " - " + product.getPrice());
                totalPrice += product.getPrice();
            }
        }
        System.out.println("Total price: " + totalPrice);
    }
}
```

Screenshot 2 — EcommerceSystem.java

```java
import java.util.Scanner;
public class EcommerceSystem {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Welcome to the e-commerce system!");
        System.out.println("Please enter your id:");
        int customerId = Math.abs(scanner.nextInt());
        scanner.nextLine();

        System.out.println("Please enter your name:");
        String name = scanner.nextLine();

        System.out.println("Please enter your address:");
        String address = scanner.nextLine();

        Customer customer = new Customer();
        customer.setCustomerId(customerId);
        customer.setName(name);
        customer.setAddress(address);

        System.out.println("How many products would you like to add to your cart?");
        int nProducts = Math.abs(scanner.nextInt());
        scanner.nextLine();

        Cart cart = new Cart();
        cart.setCustomerId(customerId);
        cart.setNProducts(nProducts);

        for (int i = 0; i < nProducts; i++) {
            System.out.println("Which product would you like to add? ");
            System.out.println("1 - smartphone");
            System.out.println("2 - T-shirt");
            System.out.println("3 - OOP");
            int choice = scanner.nextInt();
            scanner.nextLine();

            Product product;
            switch (choice) {
                case 1:
                    product = new ElectronicProduct();
                    product.setProductId(1);
```

EcommerceSystem.java Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java

```java
                product.setProductId(1);
                product.setName("smartphone");
                product.setPrice(599.9f);
                ((ElectronicProduct) product).setBrand("Samsung");
                ((ElectronicProduct) product).setWarrantyPeriod(1);
                break;
            case 2:
                product = new ClothingProduct();
                product.setProductId(2);
                product.setName("T-shirt");
                product.setPrice(19.99f);
                ((ClothingProduct) product).setSize("Medium");
                ((ClothingProduct) product).setFabric("Cotton");
                break;
            case 3:
                product = new BookProduct();
                product.setProductId(3);
                product.setName("OOP");
                product.setPrice(39.99f);
                ((BookProduct) product).setAuthor("O'Reilly");
                ((BookProduct) product).setPublisher("X Publications");
                break;
            default:
                System.out.println("Invalid choice. Please try again.");
                i--;
                continue;
        }
        cart.addProduct(i, product);
    }
    System.out.println("Your total is "+cart.calculatePrice()+ " would you like to place the order? yes or no");
    String confirmation=scanner.next();

    if(confirmation.equals("yes")){
        System.out.println("Here's Your order's summary:");
        Order order = new Order();
        order.setCustomerId(customerId);
        order.setOrderId(1);
        order.setProducts(cart.getProducts());
        order.printOrderInfo();}
}
}
```

untitled > src > EcommerceSystem   1:26 LF UTF-8 4 spaces

System.java Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java ECommerceGUI.java

```java
package ecommercesystem;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class ECommerceGUI extends JFrame {
    private JLabel customerIdLabel, nameLabel, addressLabel, productsLabel, totalPriceLabel;
    private JTextField customerIdField, nameField, addressField;
    private JComboBox<String> productsComboBox;
    private JButton addProductButton, calculatePriceButton, printOrderButton;
    private String[] selectedProducts = new String[3];
    private double totalPrice = 0;

    public ECommerceGUI() {
        setTitle("E-Commerce System");
        setSize( width: 800, height: 800);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setLayout(new GridLayout( rows: 7, cols: 2));

        customerIdLabel = new JLabel( text: "Customer ID:");
        add(customerIdLabel);
        customerIdField = new JTextField();
        add(customerIdField);
```

untitled > src > ecommercesystem > ECommerceGUI > selectedProducts   12:55 CRLF UTF-8 4 spaces

```java
            nameLabel = new JLabel( text: "Name:");
            add(nameLabel);
            nameField = new JTextField();
            add(nameField);

            addressLabel = new JLabel( text: "Address:");
            add(addressLabel);
            addressField = new JTextField();
            add(addressField);

            productsLabel = new JLabel( text: "Select Product:");
            add(productsLabel);
            String[] products = {"T-shirt - $19.99", "Smartphone - $599.9", "OOP Book - $39.99"};
            productsComboBox = new JComboBox<>(products);
            add(productsComboBox);

            addProductButton = new JButton( text: "Add Product");
            add(addProductButton);

            totalPriceLabel = new JLabel( text: "Total Price: $0.0");
            add(totalPriceLabel);

            calculatePriceButton = new JButton( text: "Calculate Total Price");
            add(calculatePriceButton);

            printOrderButton = new JButton( text: "Print Order Info");
            add(printOrderButton);

            addProductButton.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent e) {
                    for (int i = 0; i < selectedProducts.length; i++) {
```

```java
                @Override
                public void actionPerformed(ActionEvent e) {
                    for (int i = 0; i < selectedProducts.length; i++) {
                        if (selectedProducts[i] == null) {
                            selectedProducts[i] = (String) productsComboBox.getSelectedItem();
                            JOptionPane.showMessageDialog( parentComponent: null,  message: "Product added successfully!");
                            break;
                        }
                    }
                }
            });

            calculatePriceButton.addActionListener(new ActionListener() {
                @Override
                public void actionPerformed(ActionEvent e) {
                    totalPrice = 0;
                    for (String product : selectedProducts) {
                        if (product != null) {
                            if (product.contains("T-shirt")) {
                                totalPrice += 19.99;
                            } else if (product.contains("Smartphone")) {
                                totalPrice += 599.9;
                            } else if (product.contains("OOP Book")) {
                                totalPrice += 39.99;
                            }
                        }
                    }
                    totalPriceLabel.setText("Total Price: $" + totalPrice);
                }
            });

            printOrderButton.addActionListener(new ActionListener() {
```

```java
        printOrderButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                StringBuilder orderInfo = new StringBuilder();
                orderInfo.append("Customer ID: ").append(customerIdField.getText()).append("\n");
                orderInfo.append("Name: ").append(nameField.getText()).append("\n");
                orderInfo.append("Address: ").append(addressField.getText()).append("\n");
                orderInfo.append("Selected Products:\n");
                for (String product : selectedProducts) {
                    if (product != null) {
                        orderInfo.append("- ").append(product).append("\n");
                    }
                }
                orderInfo.append("Total Price: $").append(totalPrice);
                JOptionPane.showMessageDialog( parentComponent: null, orderInfo.toString(), title: "Order Information", JOptionPane.INFORMATION_MESSAGE);
            }
        });

        setVisible(true);
    }

    public static void main(String[] args) { SwingUtilities.invokeLater(() -> new ECommerceGUI()); }
}
```

titled > src > ecommercesystem > © ECommerceGUI > ⓕ selectedProducts   12:55   CRLF   UTF-8   4 spaces

---

Run - EcommerceSystem.java

Run   EcommerceSystem ×

```
D:\Java\bin\java.exe "-javaagent:D:\IntelliJ IDEA Community Edition 2023.3.6\lib\idea_rt.jar=52223:D:\IntelliJ IDEA Community Edition 2023.3.6\bin" -Dfile.encoding=UTF-8 -classpath C:\Us
Welcome to the e-commerce system!
Please enter your id:
23012068
Please enter your name:
mina wahid
Please enter your address:
address
How many products would you like to add to your cart?
4
Which product would you like to add?
1 - smartphone
2 - T-shirt
3 - OOP
2
Which product would you like to add?
1 - smartphone
2 - T-shirt
3 - OOP
3
Which product would you like to add?
1 - smartphone
2 - T-shirt
3 - OOP
2
Which product would you like to add?
1 - smartphone
2 - T-shirt
3 - OOP
1
Your total is 679.87 would you like to place the order? yes or no
yes
```

```
Your total is 679.87 would you like to place the order? yes or no
yes
Here's Your order's summary:
Order id: 1
Customer id: 23012068
Products:
T-shirt - 19.99
OOP - 39.99
T-shirt - 19.99
smartphone - 599.9
Total price: 679.87


Process finished with exit code 0
```

E-Commerce System

Customer ID:

23012068

Name:

mina wahid

Address:

address

Order Information

Customer ID: 23012068
Name: mina wahid
Address: address
Selected Products:
- Smartphone - $599.9
- OOP Book - $39.99
- T-shirt - $19.99
Total Price: $659.88

OK

Select Product:

T-shirt - $19.99

Add Product

Total Price: $659.88

Calculate Total Price

Print Order Info

Type here to search

19°C عائم غائم

ENG
US 5:19 PM
4/22/2024