E-commerce Class

EcommerceSystem.java

Source History

```
package project;
import java.util.Scanner;
public class EcommerceSystem {
    public static void main(String[] args) {
        EcommerceGui y = new EcommerceGui();
Scanner input = new Scanner(System.in);
System.out.println("Welcome to the E-Commerce System!");
System.out.println("Please enter your Id: ");
int customerId = input.nextInt();
System.out.println("Please enter your name: ");
String name = input.next();
System.out.println("Please enter your address: ");
String address = input.next();
 System.out.println("How many products you want to add to your cart?");
int ncart = input.nextInt();
 Customer myCustomer = new Customer(customerId, name, address);
 Cart myCart = new Cart(myCustomer.get_customerId(), ncart);
 for (int i = 0; i < ncart; i++) {
  System.out.println("Enter product type (1 for electronic, 2 for clothing, 3 for book):");
  int productType = input.nextInt();
  input.nextLine();
 switch (productType) {
  case 1 -> {
      ElectronicProduct newElectronicProduct = new ElectronicProduct(1,"smartphone", 599.99f,"samsung",1);
      myCart.addproduct(newElectronicProduct, i);
          }
  case 2 -> {
      ClothingProduct newClothingProduct = new ClothingProduct(2,"T-shirt", 19.99f,"Medium","cotton");
      myCart.addproduct(newClothingProduct, i);
```

EcommerceSystem.java

Source History

```
          }
  case 2 -> {
      ClothingProduct newClothingProduct = new ClothingProduct(2,"T-shirt", 19.99f,"Medium","cotton");
      myCart.addproduct(newClothingProduct, i);
          }
 case 3 -> {
    BookProduct newBookProduct = new BookProduct(3,"oop",39.99f,"O'Reilly","X Publications");
     myCart.addproduct(newBookProduct, i);
          }
 default -> {

          System.out.println("Invalid product type.");}
 }
 }
 System.out.println( "Your total is "+ myCart.calculatePrice()+ " $");
     System.out.println("\nDo you want to place an order for the products in your cart? (1-yes/2-no)");
     int response = input.nextInt();
     if (response==1) {
         System.out.println ("Here's your order's summary:");
order myOrder = new order(myCustomer.get_customerId(), 1, myCart.getProducts());
      myOrder.printOrderInfo();

     } else {
         System.out.println("Order not placed.");
     }
     System.exit(0);
      }
 }
```

Product Class

product.java

```java
package project;
public class product {
   int productId;
    String name  ;
  float  price ;
public product (int productId,String name , float price2){
    this.productId = productId;
    this.name = name;
    this.price = Math.abs(price2);
}
public int get_productId (){
      return Math.abs(productId) ;
}
public String get_name(){
      return name;
}
public float get_price (){
     return (float) Math.abs(price);
}
public void set_productId (int productId){
       this.productId=Math.abs(productId);
}
public void set_name ( String name ){
       this.name=name;
}
public void set_price ( float  price){
       this.price=price;
}
}
```

ElectronicProduct Class

ElectronicProduct.java

```java
package project;
public class ElectronicProduct extends product {
   String brand;
   int warrantyPeriod;
   public ElectronicProduct(int  productId, String name, float price, String brand, int warrantyPeriod) {
     super (productId , name,price);
     this.brand=brand;
     this.warrantyPeriod=warrantyPeriod;
   }
   public String get_brand(){
         return brand;
}
   public int get_warrantyPeriod (){
      return Math.abs(warrantyPeriod);
}
   public void set_brand (String brand){
       this.brand=brand;
}
public void set_warrantyPeriod ( int warrantyPeriod){
       this.warrantyPeriod=Math.abs(warrantyPeriod);
}
}
```

Clothingproduct Class

ClothingProduct.java

```java
package project;
public class ClothingProduct extends product {
String size;
String fabric;
public ClothingProduct(int  productId, String name, float price, String size, String fabric) {
                super (productId , name,price);
                this.size=size;
                this.fabric=fabric;
              }
  public String get_size(){
        return size;}
     public String get_fabric(){
         return fabric;}
        public void set_size(String size){
                this.size=size;
}
        public void set_fabric (String fabric){
            this. fabric = fabric;
}
}

```

Bookproduct Class

BookProduct.java

```java
package project;
public class BookProduct extends product {
    String author;
    String publisher;
    public BookProduct(int  productId, String name, float price, String author, String publisher) {
        super (productId , name  ,price);
        this.author=author;
        this.publisher=publisher;
     }
    public String get_author (){
    return author;
}
    public String get_publisher (){
    return publisher;
}
    public void set_author (String author ){
    this.author = author ;
}
    public void set_publisher (String publisher){
        this.publisher=publisher;
}
}


```

Customer Class

Customer.java

```java
package project;
public class Customer {
      int  customerId;
      String name;
      String  address;
public Customer(int customerId, String name, String address) {
      this.customerId=Math.abs(customerId);
      this.name=name;
    this.address=address;
}
public int get_customerId(){
    return Math.abs(customerId);
}
public String get_name(){
    return name;
}
public String get_address(){
    return address;
}
public void set_customerId(int  customerId){
    this.customerId=Math.abs(customerId);
}
public void set_name (String name){
    this.name=name;
}
public void set_address( String  address){
    this.address=address;
}
}
```

Order Class

order.java

```java
package project;
public class order {
    int customerId;
    int orderId;
    product[] products;
    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=calculateTotalPrice();
    }
    order() {
       }
    public float calculateTotalPrice() {
        float totalPrice = 0;
        for (product product : products) {
            if (product != null) {
                totalPrice += product.get_price();
            }
        }
        return totalPrice;
    }
    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) {
```

order.java

```
    }
    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.get_name() + ": $" + product.get_price());
            }
        }
        System.out.println("Total Price: $" + totalPrice);
    }
    public float getTotalPrice() {
        return totalPrice;
    }
}
```

Cart Class

Cart.java

```
package project;
public class Cart {
    int customerId;
    int   nProducts ;
    product[] products;
    public Cart (int customerId , int nproducts){
        this.customerId=Math.abs(customerId);
        this.nProducts= Math.abs(nproducts);
        this.products=new product[nproducts];
    }
   public int get_customerId(){
    return Math.abs(customerId);
}
   public int get_nproducts(){
       return Math.abs(nProducts);
   }
   public void set_customerId(int customerId){
    this.customerId = customerId;
   }
   public void set_nproducts(int nProducts){
    this.nProducts = nProducts;
   }
   public void addproduct(product product, int index){
       if (index >= 0 && index < nProducts) {
        products[index] = product;
        } else {
        System.out.println("Invalid index.");
        }
        }
```

Cart.java

Source History

```
            System.out.println("Invalid index.");
            }
            }
    public void removeproduct(int index){
     if (index >= 0 && index < nProducts) {
         products[index] = null;
         } else {
         System.out.println("Invalid index.");
         }
    }
    public float calculatePrice(){
     float totalPrice = 0;
for (product product : products) {
if (product != null) {
totalPrice += product.get_price();
}
}
return totalPrice;
}
 public product [] getProducts(){
   return products;
}
  public int getcustomerId(){
      return customerId;
  }
   public void placeOrder (){

   }
}
```

Output

BookProduct.java | Customer.java | ElectronicProduct.java | order.java | product.java | Cart.java | EcommerceSystem.java | Output - project (run)

```
run:
Welcome to the E-Commerce System!
Please enter your Id:
20231
Please enter your name:
Noureen
Please enter your address:
address
How many products you want to add to your cart?
4
Enter product type (1 for electronic, 2 for clothing, 3 for book):
2
Enter product type (1 for electronic, 2 for clothing, 3 for book):
3
Enter product type (1 for electronic, 2 for clothing, 3 for book):
2
Enter product type (1 for electronic, 2 for clothing, 3 for book):
1
Your total is 679.95996 $

Do you want to place an order for the products in your cart? (1-yes/2-no)
1
Here's your order's summary:
Order ID: 1
Customer ID: 20231
Products:
- T-shirt: $19.99
- oop: $39.99
- T-shirt: $19.99
- smartphone: $599.99
Total Price: $679.95996
BUILD SUCCESSFUL (total time: 49 seconds)
```

1:1 INS

Gui Classes

1-E-commerceGui class

EcommerceGui.java

Source History

```
1  package project;
2  import javax.swing.*;
3  import java.awt.event.ActionEvent;
4  import java.awt.event.ActionListener;
5  public class EcommerceGui extends JFrame implements ActionListener {
6      JPanel p1;
7      JLabel l1, l2, l3, l4, l5;
8      JTextField t1, t2, t3, t4;
9      JButton b1, b2;
10     private cartGui cartGui;
       private String customerId;private String orderid;
12     EcommerceGui() {
13         this.setTitle("E-commerce");
14         this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
15         this.setResizable(false);
16         this.setSize(500, 400);
17         this.setLocation(200, 50);
18         l1 = new JLabel("Please enter your Id: ");
19         l2 = new JLabel("Please enter your name: ");
20         l3 = new JLabel("Please enter your address: ");
21         l4 = new JLabel("How many products you want to add to cart?");
22         l5 = new JLabel("Welcome to the E-commerce system: ");
23         p1 = new JPanel();
24         p1.setLayout(null);
25         t1 = new JTextField();
26         t2 = new JTextField();
27         t3 = new JTextField();
28         t4 = new JTextField();
29         l1.setBounds(10, 50, 300, 25);
```

EcommerceGui.java

Source History

```
27         t3 = new JTextField();
28         t4 = new JTextField();
29         l1.setBounds(10, 50, 300, 25);
30         t1.setBounds(280, 50, 200, 25);
31         l2.setBounds(10, 90, 300, 25);
32         t2.setBounds(280, 90, 200, 25);
33         t3.setBounds(280, 130, 200, 25);
34         l3.setBounds(10, 130, 300, 25);
35         l4.setBounds(10, 170, 300, 25);
36         t4.setBounds(280, 170, 200, 25);
37         l5.setBounds(10, 10, 300, 25);
38         this.add(p1);
39         p1.add(l1);p1.add(l2);p1.add(l3);p1.add(t4);p1.add(l4);
40         p1.add(t1);p1.add(t2);p1.add(t3);p1.add(l5);
41         this.setVisible(true);
           t1.addActionListener(this);
           t4.addActionListener(this);
44         b1 = new JButton("create");
45         b1.setBounds(50, 230, 150, 25);
46         p1.add(b1);
           b1.addActionListener(this);
48         b2 = new JButton("Exist");
49         b2.setBounds(250, 230, 150, 25);
50         p1.add(b2);
           b2.addActionListener(this);
52     }
       EcommerceGui(String customerId, String orderid){
54         this.customerId=customerId;
55         this.orderid=orderid;
```

EcommerceGui.java

```
            p1.add(b2);
            b2.addActionListener(this);
        }
        EcommerceGui(String customerId, String orderid){
            this.customerId=customerId;
            this.orderid=orderid;
        }
        @Override
        public void actionPerformed(ActionEvent e) {
            if (e.getSource() == b1) {
                String nproduct = t4.getText().toString();
                int x = Integer.parseInt(nproduct);
                String orderId = "1";
                String customerID = t1.getText().toString();
                cartGui a = new cartGui(customerID, orderId, x);
                dispose();
            }
            if (e.getSource()== b2){

                this.dispose();
            }
        }
        public cartGui getMyFrame() {
            return cartGui;
        }
    }
```

2-OrderGui class

orderGui.java

```
package project;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class orderGui extends JFrame implements ActionListener {
    private JLabel label1,label2,label3,label4,label5;
    private JButton buttton;
    orderGui(String customerid, String orderid, String selectedItemText, float totalPrice) {
        setTitle("Order place : ");
        setSize(700,400);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBackground(Color.gray);
        this.setResizable(false);
        setLayout(new BoxLayout(getContentPane(),BoxLayout.Y_AXIS));
        label1 = new JLabel("Here's your summary : ");
        label2 = new JLabel("CustomerID : " + customerid);
        label3 = new JLabel("OrderID : "+ orderid);
        label4 = new JLabel("<html>" + "Selected items : "+selectedItemText+"</html>");
        label5 = new JLabel("Total price : " + totalPrice + " $");
        buttton = new JButton("Ok");
        label1.setBounds(5,50,300,25);
        label2.setBounds(5,100,300,25);
        label3.setBounds(5,150,300,25);
        label4.setPreferredSize(new Dimension(label4.getPreferredSize().width,label4.getPreferredSize().height));
        label5.setBounds(5,250,400,10);
        buttton.setBounds(50,350,10,25);
        add(label1);
        add(Box.createVerticalStrut(10));
```

orderGui.java

```
            label1.setBounds(5,50,300,25);
            label2.setBounds(5,100,300,25);
            label3.setBounds(5,150,300,25);
            label4.setPreferredSize(new Dimension(label4.getPreferredSize().width,label4.getPreferredSize().height));
            label5.setBounds(5,250,400,10);
            buttton.setBounds(50,350,10,25);
            add(label1);
            add(Box.createVerticalStrut(10));
            add(label2);
            add(Box.createVerticalStrut(10));
            add(label3);
            add(Box.createVerticalStrut(10));
            add(label4);
            add(Box.createVerticalStrut(10));
            add(label5);
            add(Box.createVerticalStrut(30));
            add(buttton);
            buttton.addActionListener(this);
            setVisible(true);
        }
        @Override
        public void actionPerformed(ActionEvent e) {
            if (e.getSource() == buttton) {
               this.dispose();
               System.exit(0);
            }
        }
    }

```

3-cartGui class

cartGui.java | EcommerceGui.java | orderGui.java | EcommerceSystem.java

```
package project;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class cartGui extends JFrame implements ActionListener {
    ButtonGroup g1 = new ButtonGroup();
    private JRadioButton radioButton1, radioButton2;
    private JLabel l11;
    private int count = 0;
    private int maxClicks;
    private float totalprice = 0;
    private JLabel totalPriceLabel;
    private String selectedItemsText = "";
    private String customerid;
    private String orderid;
    cartGui(String x, String y, int maxClicks) {
        customerid = x;
        orderid = y;
        this.maxClicks = maxClicks;
        initializeFrame();
    }
    private void initializeFrame() {
        this.setTitle("E-commerce products ");
        this.setSize(500, 600);
        this.setLocationRelativeTo(null);
        this.setResizable(false);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setLayout(null);
```

cartGui.java | EcommerceGui.java | orderGui.java | EcommerceSystem.java

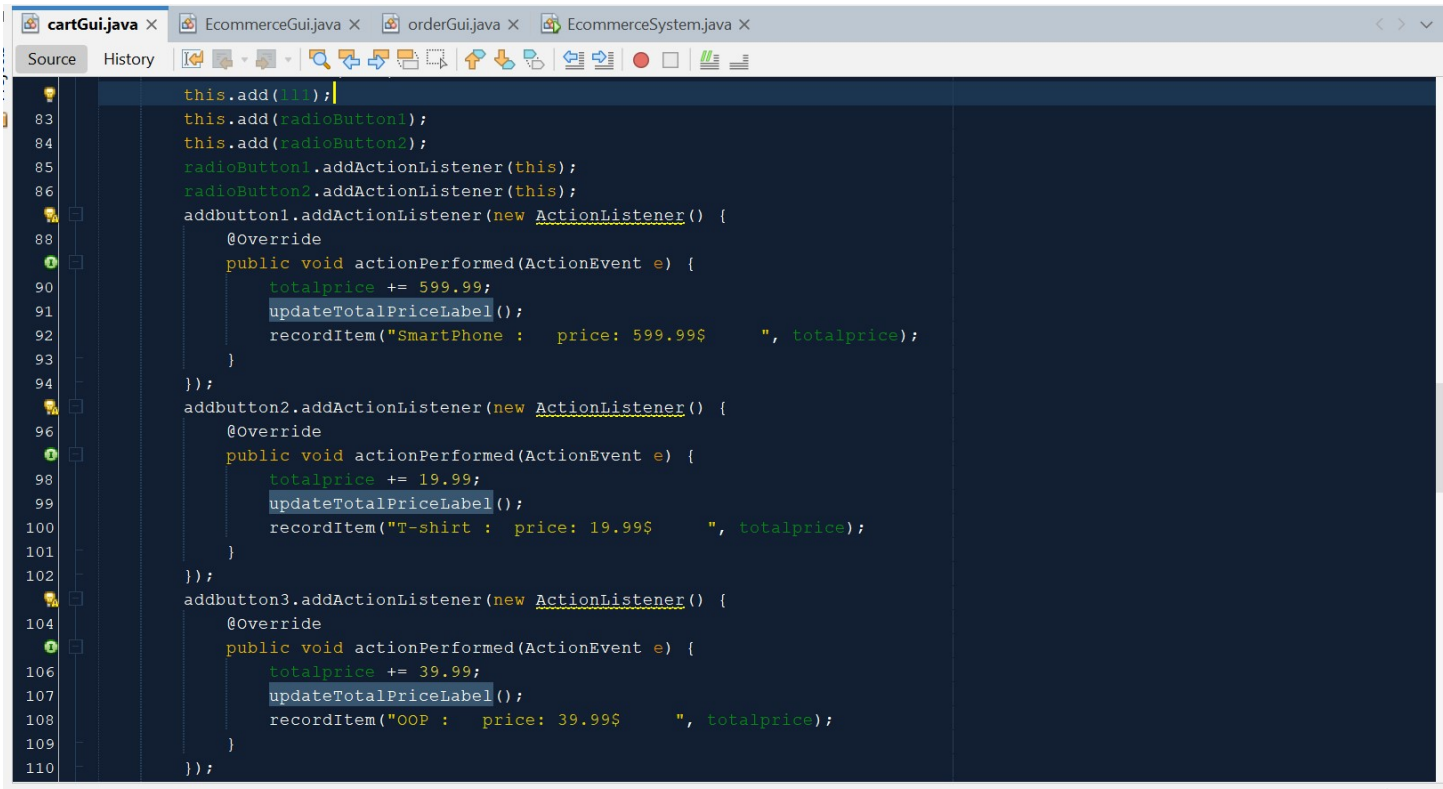
Source | History

```
26          this.setLocationRelativeTo(null);
27          this.setResizable(false);
28          this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
29          this.setLayout(null);
30          JLabel l1 = new JLabel("Enter product type ");
31          JLabel l2 = new JLabel("electronic");
32          JLabel l3 = new JLabel("clothing");
33          JLabel l4 = new JLabel("book");
34          JButton addbutton1 = new JButton("Add");
35          JButton addbutton2 = new JButton("Add");
36          JButton addbutton3 = new JButton("Add");
37          JButton removebutton1 = new JButton("Remove");
38          JButton removebutton2 = new JButton("Remove");
39          JButton removebutton3 = new JButton("Remove");
            JButton orderplace = new JButton("Place order");
41          l1.setBounds(10, 20, 160, 40);
42          l2.setBounds(10, 60, 160, 40);
43          l3.setBounds(10, 100, 160, 40);
44          l4.setBounds(10, 140, 160, 40);
45          addbutton1.setBounds(210, 60, 60, 30);
46          addbutton2.setBounds(210, 100, 60, 30);
47          addbutton3.setBounds(210, 140, 60, 30);
48          removebutton1.setBounds(310, 60, 160, 30);
49          removebutton2.setBounds(310, 100, 160, 30);
50          removebutton3.setBounds(310, 140, 160, 30);
51          addbutton1.addActionListener(this);
52          addbutton2.addActionListener(this);
53          addbutton3.addActionListener(this);
54          removebutton1.addActionListener(this);
55          removebutton2.addActionListener(this);
```

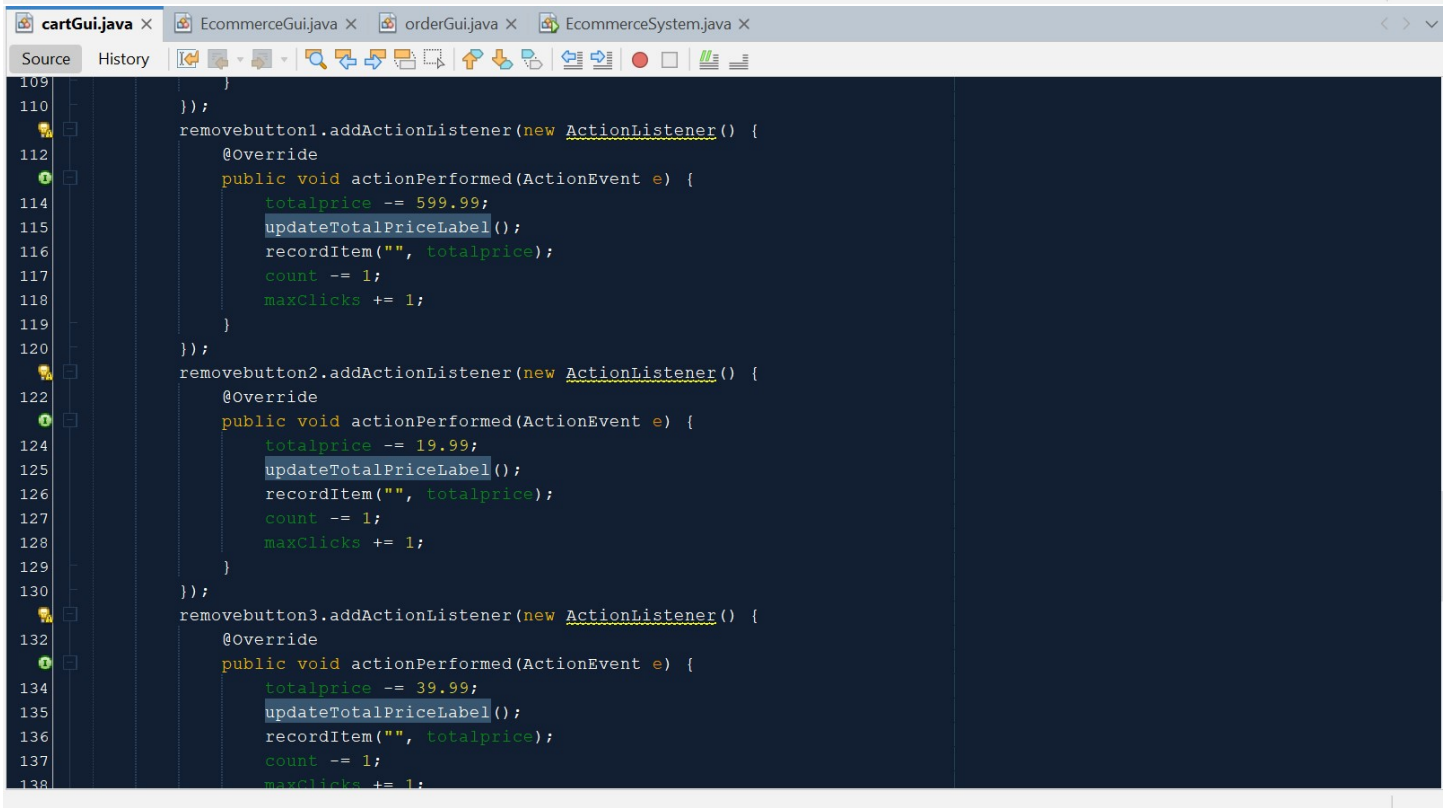
cartGui.java | EcommerceGui.java | orderGui.java | EcommerceSystem.java

Source | History

```
54          removebutton1.addActionListener(this);
55          removebutton2.addActionListener(this);
56          removebutton3.addActionListener(this);
57          this.add(addbutton1);
58          this.add(addbutton2);
59          this.add(addbutton3);
60          this.add(removebutton1);
61          this.add(removebutton2);
62          this.add(removebutton3);
63          this.add(l1);
64          this.add(l2);
65          this.add(l3);
66          this.add(l4);
67          this.setVisible(true);
68          totalPriceLabel = new JLabel("Total Price: $0.00");
69          totalPriceLabel.setBounds(70, 190, 200, 30);
70          this.add(totalPriceLabel);
71          l11 = new JLabel("Do you want to place an order for the products in your cart?");
72          radioButton1 = new JRadioButton("yes");
73          radioButton2 = new JRadioButton("No");
74          l11.setBounds(10, 170, 350, 150);
75          radioButton1.setBounds(100, 300, 50, 30);
76          radioButton2.setBounds(250, 300, 50, 30);
77          g1.add(radioButton1);
78          g1.add(radioButton2);
79          radioButton1.setVisible(true);
80          radioButton2.setVisible(true);
81          l11.setVisible(true);
            this.add(l11);
```



```
cartGui.java × | EcommerceGui.java × | orderGui.java × | EcommerceSystem.java ×
Source | History
        this.add(111);
83      this.add(radioButton1);
84      this.add(radioButton2);
85      radioButton1.addActionListener(this);
86      radioButton2.addActionListener(this);
        addbutton1.addActionListener(new ActionListener() {
88          @Override
            public void actionPerformed(ActionEvent e) {
90              totalprice += 599.99;
91              updateTotalPriceLabel();
92              recordItem("SmartPhone :   price: 599.99$     ", totalprice);
93          }
94      });
        addbutton2.addActionListener(new ActionListener() {
96          @Override
            public void actionPerformed(ActionEvent e) {
98              totalprice += 19.99;
99              updateTotalPriceLabel();
100             recordItem("T-shirt :  price: 19.99$     ", totalprice);
101         }
102     });
        addbutton3.addActionListener(new ActionListener() {
104         @Override
            public void actionPerformed(ActionEvent e) {
106             totalprice += 39.99;
107             updateTotalPriceLabel();
108             recordItem("OOP :   price: 39.99$     ", totalprice);
109         }
110     });
```



```
cartGui.java × | EcommerceGui.java × | orderGui.java × | EcommerceSystem.java ×
Source | History
109         }
110     });
        removebutton1.addActionListener(new ActionListener() {
112         @Override
            public void actionPerformed(ActionEvent e) {
114             totalprice -= 599.99;
115             updateTotalPriceLabel();
116             recordItem("", totalprice);
117             count -= 1;
118             maxClicks += 1;
119         }
120     });
        removebutton2.addActionListener(new ActionListener() {
122         @Override
            public void actionPerformed(ActionEvent e) {
124             totalprice -= 19.99;
125             updateTotalPriceLabel();
126             recordItem("", totalprice);
127             count -= 1;
128             maxClicks += 1;
129         }
130     });
        removebutton3.addActionListener(new ActionListener() {
132         @Override
            public void actionPerformed(ActionEvent e) {
134             totalprice -= 39.99;
135             updateTotalPriceLabel();
136             recordItem("", totalprice);
137             count -= 1;
138             maxClicks += 1;
```

cartGui.java | EcommerceGui.java | orderGui.java | EcommerceSystem.java

Source | History

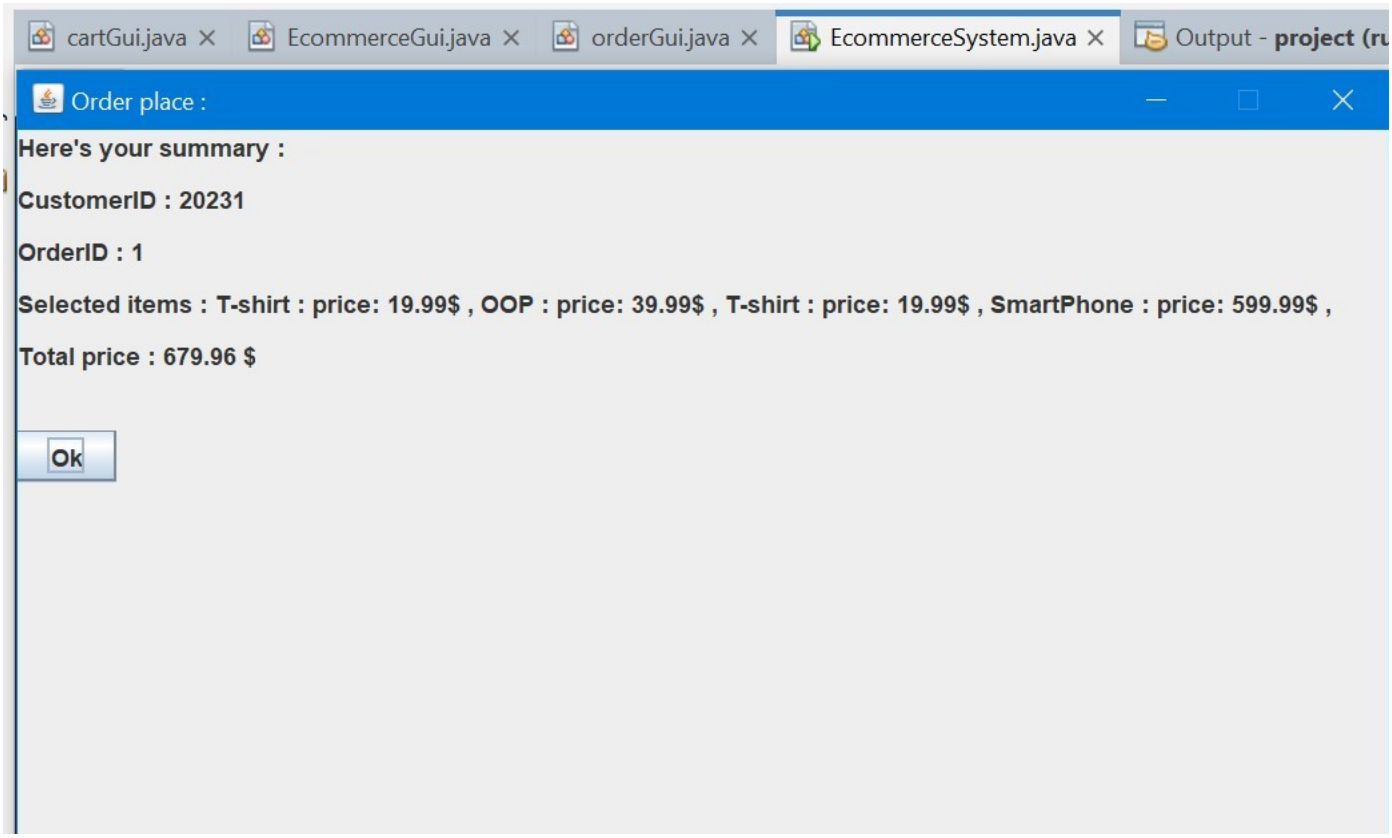
```
        });
        removebutton3.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                totalprice -= 39.99;
                updateTotalPriceLabel();
                recordItem("", totalprice);
                count -= 1;
                maxClicks += 1;
            }
        });
    }
    private void updateTotalPriceLabel() {
        totalPriceLabel.setText("Total Price: $" + String.format("%.2f", totalprice));
    }
    private void recordItem(String itemName, float totalprice) {
        selectedItemsText += itemName + " , ";
    }
    private void disableButtons() {
        Component[] components = this.getContentPane().getComponents();
        for (Component component : components) {
            if (component instanceof JButton) {
                JButton button = (JButton) component;
                button.setEnabled(false);
            }
        }
    }
    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() instanceof JButton) {
```

cartGui.java | EcommerceGui.java | orderGui.java | EcommerceSystem.java

Source | History

```
        Component[] components = this.getContentPane().getComponents();
        for (Component component : components) {
            if (component instanceof JButton) {
                JButton button = (JButton) component;
                button.setEnabled(false);
            }
        }
    }
    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() instanceof JButton) {
            JButton button = (JButton) e.getSource();
            count++;
            if (count >= maxClicks) {
                disableButtons();
            }
        }
        if (e.getSource() == radioButton1) {
            String customerId = customerid;
            String orderedItem = orderid;
            orderGui Guiorder = new orderGui(customerId, orderedItem, selectedItemsText, totalprice);
            dispose();
        } else if (e.getSource() == radioButton2) {
            JOptionPane.showMessageDialog(null, "order not placed");
            this.dispose();
        }
    }
}
```

GUI output


cartGui.java
EcommerceGui.java
orderGui.java
EcommerceSystem.java
Output - project (ru
Order place :
Here's your summary :
CustomerID : 20231
OrderID : 1
Selected items : T-shirt : price: 19.99$, OOP : price: 39.99$, T-shirt : price: 19.99$, SmartPhone : price: 599.99$,
Total price : 679.96 $
Ok