

Java\E-Commerce\Product.java

```
1  public class Product {
2      protected int productId;
3      protected String name;
4      protected float price;
5
6      public int getProductId() {
7          return productId;
8      }
9
10     public void setProductId(int productId)
11     {
12         this.productId = Math.abs(productId)
13         ;
14     }
15
16     public String getName() {
17         return name;
18     }
19
20     public void setName(String name) {
21         this.name = name;
22     }
23
24     public float getPrice() {
25         return price;
26     }
27
28     public void setPrice(float price) {
29         this.price = Math.abs(price);
30     }
31 }
```

Java\E-Commerce\ElectronicProduct.java

```
1  public class ElectronicProduct extends
   Product{
2      private String brand;
3      private int warrantyPeriod;
4
5      public String getBrand() {
6          return brand;
7      }
8
9      public void setBrand(String brand) {
10         this.brand = brand;
11     }
12
13     public int getWarrantyPeriod() {
14         return warrantyPeriod;
15     }
16
17     public void setWarrantyPeriod(int
   warrantyPeriod) {
18         this.warrantyPeriod =
   Math.abs(warrantyPeriod);
19     }
20 }
```

Java\E-Commerce\ClothingProduct.java

```
1  public class ClothingProduct extends
   Product{
2      private String size, fabric;
3
4      public String getSize() {
5          return size;
6      }
7
8      public void setSize(String size) {
9          this.size = size;
10     }
11
12     public String getFabric() {
13         return fabric;
14     }
15
16     public void setFabric(String fabric) {
17         this.fabric = fabric;
18     }
19 }
```

Java\E-Commerce\BookProduct.java

```
1  public class BookProduct extends Product{
2      private String author, publisher;
3      public String getAuthor() {
4          return author;
5      }
6      public void setAuthor(String author) {
7          this.author = author;
8      }
9      public String getPublisher() {
10         return publisher;
11     }
12     public void setPublisher(String
publisher) {
13         this.publisher = publisher;
14     }
15 }
```

Java\E-Commerce\Customer.java

```
1  public class Customer {
2      private int customerId;
3      private String name, address;
4
5      public int getCustomerId() {
6          return customerId;
7      }
8
9      public void setCustomerId(int
customerId) {
10         this.customerId =
Math.abs(customerId);
11     }
12
13     public String getName() {
14         return name;
15     }
16
17     public void setName(String name) {
18         this.name = name;
19     }
20
21     public String getAddress() {
22         return address;
23     }
24
25     public void setAddress(String address) {
26         this.address = address;
27     }
28 }
```

Java\E-Commerce\Cart.java

```
1  import java.util.Scanner;
2
3  public class Cart {
4      static Scanner input = new Scanner(System.in);
5      private int customerId, nProducts;
6      Product[] products;
7
8      public int getCustomerId() {
9          return customerId;
10     }
11
12     public void setCustomerId(int customerId) {
13         this.customerId = Math.abs(customerId);
14     }
15
16     public int getnProducts() {
17         return nProducts;
18     }
19
20     public void setnProducts(int nProducts) {
21         this.nProducts = Math.abs(nProducts);
22     }
23
24     public Product[] getProducts() {
25         return products;
26     }
27
28     public void setProducts() {
29         this.products = new Product[nProducts];
30     }
31
32     public void addProduct(Product product) {
33         this.products[--nProducts] = product;
34     }
35
36     public void removeProduct(int index) {
37         this.products[index] = null;
38     }
39
40     public float calculatePrice() {
41         float totalPrice = 0;
42         for (int i = 0; i < products.length; i++) {
43             totalPrice += products[i].getPrice();
44         }
45         return totalPrice;
46     }
47
48     public boolean placeOrder() {
49         System.out.print("Your total price is: $" + calculatePrice() + "\t");
50         System.out.print("Would you like to place your order for the products in the cart?
(Y/N): ");
51         if (input.nextLine().toUpperCase().equals("Y")) {
52             return true;
53         } else {
54             return false;
55         }
56     }
57 }
```

Java\E-Commerce\Order.java

```
1  public class Order {
2      private int customerId, orderId = 1;
3      private Product[] products;
4      private float totalPrice;
5
6      public int getCustomerId() {
7          return customerId;
8      }
9
10     public void setCustomerId(int customerId) {
11         this.customerId = Math.abs(customerId);
12     }
13
14     public int getOrderId() {
15         return (++orderId);
16     }
17
18     public void setOrderId() {
19         this.orderId = Math.abs(orderId);
20     }
21
22     public Product[] getProducts() {
23         return products;
24     }
25
26     public void setProducts(Product[] products) {
27         this.products = products;
28     }
29
30     public float getTotalPrice() {
31         return totalPrice;
32     }
33
34     public void setTotalPrice(float totalPrice) {
35         this.totalPrice = Math.abs(totalPrice);
36     }
37
38     public void printOrderInfo() {
39         System.out.println("Here's your order's summary:");
40         System.out.println("Order id: " + orderId + "\tCustomer
id: " + customerId);
41         System.out.println("Products:");
42         for (int i = 0; i < products.length; i++) {
43             System.out.println(products[i].getName() + " - $" +
products[i].getPrice());
44             totalPrice += products[i].getPrice();
45         }
46         System.out.println("Total price: $" + totalPrice);
47     }
48 }
```

Java\E-Commerce\EcommerceSystemGUI.java

```

1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.*;
4
5 public class EcommerceSystemGUI extends JFrame {
6     private JTextField nameField, addressField, customerIdField, numProductsField;
7     private JButton submitButton, addProductButton;
8     private JLabel nameLabel, addressLabel, customerIdLabel, numProductsLabel;
9     private JList<String> productList;
10    private DefaultListModel<String> productListModel;
11    private int numSelectedProducts, numProducts;
12    private Cart cart;
13
14    public EcommerceSystemGUI() {
15        setTitle("E-commerce System");
16        setSize(500, 300);
17        setDefaultCloseOperation(EXIT_ON_CLOSE);
18        setLayout(new BorderLayout());
19        JPanel inputPanel = new JPanel(new GridLayout(5, 2));
20        nameLabel = new JLabel("Enter your name");
21        nameField = new JTextField();
22        addressLabel = new JLabel("Enter your address");
23        addressField = new JTextField();
24        customerIdLabel = new JLabel("Enter your id");
25        customerIdField = new JTextField();
26        numProductsLabel = new JLabel("Enter number of products");
27        numProductsField = new JTextField();
28        numProductsField.addFocusListener(new FocusAdapter() {
29            public void focusLost(FocusEvent e) {
30                if (!numProductsField.getText().isEmpty()) {
31                    numProducts = Integer.parseInt(numProductsField.getText());
32                    addProductButton.setEnabled(true);
33                }
34            }
35        });
36        inputPanel.add(nameLabel);
37        inputPanel.add(nameField);
38        inputPanel.add(addressLabel);
39        inputPanel.add(addressField);
40        inputPanel.add(customerIdLabel);
41        inputPanel.add(customerIdField);
42        inputPanel.add(numProductsLabel);
43        inputPanel.add(numProductsField);
44
45        JPanel productListPanel = new JPanel(new BorderLayout());
46        productListModel = new DefaultListModel<>();
47        productList = new JList<>(productListModel);
48        productListPanel.add(new JScrollPane(productList), BorderLayout.CENTER);
49
50        JPanel productButtonPanel = new JPanel(new FlowLayout());
51        addProductButton = new JButton("Add Product");
52        addProductButton.setEnabled(false);
53        addProductButton.addActionListener(e -> {
54            String selectedProduct = (String) JOptionPane.showInputDialog(null, "Select a product", "Product Selection", JOptionPane.QUESTION_MESSAGE, null, new String[]{"Smartphone", "T-shirt",
55"Book"}, "Smartphone");
56            if (selectedProduct != null && numSelectedProducts < numProducts) {
57                productListModel.addElement(selectedProduct);
58                numSelectedProducts++;
59                if (numSelectedProducts == numProducts) {
60                    addProductButton.setEnabled(false);
61                }
62            }
63        });
64        productButtonPanel.add(addProductButton);
65        productListPanel.add(productButtonPanel, BorderLayout.SOUTH);
66
67        JPanel mainPanel = new JPanel(new BorderLayout());
68        mainPanel.add(inputPanel, BorderLayout.NORTH);
69        mainPanel.add(productListPanel, BorderLayout.CENTER);
70
71        JPanel buttonPanel = new JPanel(new FlowLayout());
72        submitButton = new JButton("Submit");
73        submitButton.addActionListener(e -> {
74            createCart();
75            displayOrderSummary();
76        });
77        buttonPanel.add(submitButton);
78
79        add(mainPanel, BorderLayout.CENTER);
80        add(buttonPanel, BorderLayout.SOUTH);
81
82        numSelectedProducts = 0;
83        setVisible(true);
84    }
85
86    private void createCart() {
87        cart = new Cart();
88        cart.setCustomerId(Integer.parseInt(customerIdField.getText()));
89        cart.setnProducts(numProducts);
90        cart.setProducts();
91        for (int i = 0; i < productListModel.getSize(); i++) {
92            String product = productListModel.getElementAt(i);
93            switch (product) {
94                case "Smartphone":
95                    ElectronicProduct smartphone = new ElectronicProduct();
96                    smartphone.setName("Smartphone");
97                    smartphone.setPrice(599.9f);
98                    cart.addProduct(smartphone);
99                    break;
100                case "T-shirt":
101                    ClothingProduct tShirt = new ClothingProduct();
102                    tShirt.setName("T-shirt");
103                    tShirt.setPrice(19.99f);
104                    cart.addProduct(tShirt);
105                    break;
106                case "Book":
107                    BookProduct book = new BookProduct();
108                    book.setName("Book");
109                    book.setPrice(39.99f);
110                    cart.addProduct(book);
111                    break;
112            }
113        }
114    }
115
116    private void displayOrderSummary() {
117        StringBuilder message = new StringBuilder();
118        message.append("<html><body>");
119        message.append("<h2 style='text-align:center;'>Order Summary</h2>");
120        message.append("<p><b>Name:</b> " + nameField.getText() + "</p>");
121        message.append("<p><b>Address:</b> " + addressField.getText() + "</p>");
122        message.append("<p><b>Customer ID:</b> " + customerIdField.getText() + "</p>");
123        message.append("<p><b>Total Products:</b> " + numSelectedProducts + "</p>");
124        message.append("<p><b>Selected Products:</b></p>");
125        message.append("<ul>");
126        float totalPrice = 0.0f;
127        for (Product product : cart.getProducts()) {
128            message.append("<li>").append(product.getName()).append(" - $").append(product.getPrice()).append("</li>");
129            totalPrice += product.getPrice();
130        }
131        message.append("</ul>");
132        message.append("<p><b>Total Price:</b> $").append(totalPrice).append("</p>");
133        message.append("</body></html>");
134
135        JOptionPane.showMessageDialog(null, message.toString());
136    }
137
138    public static void main(String[] args) {
139        new EcommerceSystemGUI();
140    }
141 }

```