

EcommerceSystem (with JavaFX) by

يوسف السيد ابراهيم حسن الفقي

23011629

مجموعة الاثنين 10:30

Ecommerce System

```
1 package application;
2
3 import javafx.application.Application;
4
5 public class EcommerceSystem extends Application {
6
7     @Override
8     public void start(Stage primaryStage) {
9         try {
10             Parent root = FXMLLoader.load(getClass().getResource("scene.fxml"));
11
12             Scene scene = new Scene(root, 600, 400);
13
14             primaryStage.setScene(scene);
15             primaryStage.setTitle("JavaFX Application");
16             primaryStage.show();
17         } catch (Exception e) {
18             e.printStackTrace();
19         }
20     }
21
22     public static void main(String[] args) {
23         launch(args);
24     }
25 }
```

Customer

```
1 package application;
2
3 public class Customer {
4     private int customerId;
5     private String name;
6     private String address;
7     //Constructors
8     Customer(){}
9     Customer(int customerId, String name, String address){
10         this.customerId=Math.abs(customerId);
11         this.name=name;
12         this.address=address;
13     }
14     //setters
15     void setCID(int a){
16         customerId=Math.abs(a);
17     }
18     void setname(String b){
19         name=b;
20     }
21     void setaddress(String c){
22         address=c;
23     }
24     //getters
25     int getCID(){
26         return customerId;
27     }
28     String getname(){
29         return name;
30     }
31     String getaddress(){
32         return address;
33     }
34 }
35
36
```

Product

```
1 package application;
2
3 public class Product {
4     private int productId;
5     private String name;
6     private float price;
7
8     //constructors
9     Product(){}
10    Product(int pId, String pname, float pprice){
11        productId = Math.abs(pId);
12        name = pname;
13        price = Math.abs(pprice);
14    }
15    //Setters
16    void setId(int x){
17        productId = Math.abs(x);
18    }
19    void setName(String y){
20        name = y;
21    }
22    void setprice(float z){
23        price = Math.abs(z);
24    }
25    //Getters
26    int getId(){
27        return productId;
28    }
29    String getName(){
30        return name;
31    }
32    float getPrice(){
33        return price;
34    }
35    }
36
```

ElectronicProduct

```
1 package application;
2
3 public class ElectronicProduct extends Product{
4     private String brand;
5     private int warrantyPeriod;
6     //constructors
7     ElectronicProduct(){}
8     ElectronicProduct(int pId, String pname, float pprice, String brand,int warrantyPeriod){
9         super(pId, pname, pprice);
10        this.brand = brand;
11        this.warrantyPeriod=Math.abs(warrantyPeriod);
12    }
13    //Setters
14    void setbrand(String a){
15        brand= a;
16    }
17    void setwarrantyPeriod(int b){
18        warrantyPeriod = Math.abs(b);
19    }
20
21    //Getters
22    int getwarrantyPeriod(){
23        return warrantyPeriod;
24    }
25    String getbrand(){
26        return brand;
27    }
28
29 }
```

ClothingProduct

```
1 package application;
2
3 public class ClothingProduct extends Product{
4     private String size;
5     private String fabric;
6     //constructors
7     ClothingProduct(){}
8     ClothingProduct(int pId, String pname, float pprice, String size,String fabric){
9         super(pId, pname, pprice);
10        this.size = size;
11        this.fabric = fabric;
12    }
13    //Setters
14    void setsize(String a){
15        size= a;
16    }
17    void setfabric(String b){
18        fabric = b;
19    }
20
21    //Getters
22
23    String getsize(){
24        return size;
25    }
26    String getfabric(){
27        return fabric;
28    }
29 }
30
```

BookProduct

```
1 package application;
2
3
4 public class BookProduct extends Product {
5     private String author;
6     private String publisher;
7     //constructors
8     BookProduct(){}
9     BookProduct(int pId, String pname, float pprice, String author,String publisher){
10         super(pId, pname, pprice);
11         this.author = author;
12         this.publisher = publisher;
13     }
14     //Setters
15     void setauthor(String a){
16         author= a;
17     }
18     void setpublisher(String b){
19         publisher = b;
20     }
21
22     //Getters
23
24     String getauthor(){
25         return author;
26     }
27     String getpublisher(){
28         return publisher;
29     }
30 }
31
```

Cart p1

```
1 package application;
2 import java.util.Scanner;
3 public class Cart {
4     private Order order;
5     private int customerId;
6     private int nProducts;
7     private Product[] products = new Product[nProducts];
8     Scanner input = new Scanner(System.in);
9     String orderInfo;
10
11     //Constructors
12     Cart(){
13     }
14     Cart(int customerId, int nProducts, Product products[]){
15         this.customerId=Math.abs(customerId);
16         this.nProducts=Math.abs(nProducts);
17         this.products=products;
18     }
19     //Setters
20     void setCID(int a){
21         customerId=Math.abs(a);
22     }
23     void setnProducts(int b){
24         nProducts = Math.abs(b);
25     }
26     void setArray(Product products[]){
27         this.products=products;
28     }
29
30     //getters
31     int getCID(){
32         return customerId;
33     }
34     int getnProducts(){
35         return nProducts;
36     }
37     Product[] getproducts(){
38         return products;
39     }
40 }
```


Cart p2

```
40
41 //methods
42 void addProduct(Product product) {
43     if (nProducts < products.length) {
44         products[nProducts] = product;
45         nProducts++;
46     }
47 }
48
49 void removeProduct(Product[] products, int index) {
50     if (index < 0 || index >= products.length) {
51         System.out.println("Invalid index!!!");
52         return;
53     }
54     Product[] newProducts = new Product[products.length - 1];
55     int newIndex = 0;
56     for (int i = 0; i < products.length; i++) {
57         if (i == index) {
58             continue;
59         }
60         newProducts[newIndex] = products[i];
61         newIndex++;
62     }
63     this.products = newProducts;
64     nProducts--;
65 }
66
67
68 float calculatePrice(){
69     float totalPrice = 0;
70     for(int i=0; i<this.products.length; i++){
71         totalPrice = totalPrice + this.products[i].getprice();
72     }
73     return (float)totalPrice;
74 }
75
76 void placeOrder() {
77     Order order = new Order(customerId, 121, products, calculatePrice());
78     orderInfo = order.printOrderInfo();
79 }
80
81
82
83 }
```

Order

```
1 package application;
2 public class Order {
3     private int customerId;
4     private int orderId;
5     private Product[] products;
6     private float totalPrice;
7
8     private String orderSummary;
9     //constructors
10    Order(){}
11    Order(int customerId, int orderId, Product[] products, float totalPrice){
12        this.customerId=Math.abs(customerId);
13        this.orderId=Math.abs(orderId);
14        this.products=products;
15        this.totalPrice=totalPrice;
16    }
17
18    public String printOrderInfo() {
19        String orderInfo = "Here's your order summary:\n" +
20            "Order ID: " + orderId + "\n" +
21            "Customer ID: " + customerId + "\n" +
22            "Products: \n";
23        for (Product product : products) {
24            orderInfo += product.getname() + " - $" + product.getprice() + "\n";
25        }
26        orderInfo += "Total Price: $" + totalPrice;
27        System.out.println(orderInfo);
28        return orderInfo;
29    }
30
31 }
32
```

Controller files

Controller1

```
1 package application;
2 import java.io.IOException;
13 public class Controller {
14     private Stage stage;
15     private Scene scene;
16     private Parent root;
17     @FXML
18     private Label myLabel;
19     @FXML
20     private TextField TField1;
21     @FXML
22     private TextField TField2;
23     @FXML
24     private TextField TField3;
25     @FXML
26     private Button button;
27     int customerID;
28     String name;
29     String address;
30     public void submit(ActionEvent event) {
31         try {
32             customerID = Integer.parseInt(TField2.getText().trim());
33             if (customerID <= 0) {
34                 throw new IllegalArgumentException("Customer ID must be a positive integer.");
35             }
36             name = TField1.getText();
37             address = TField3.getText();
38             Customer customer = new Customer(customerID,name,address);
39
40             FXMLLoader loader = new FXMLLoader(getClass().getResource("scene2.fxml"));
41             root=loader.load();
42             Controller2 scene2 = loader.getController();
43             scene2.setAll(customer.getID(), customer.getname());
44             stage = (Stage)((Node)event.getSource()).getScene().getWindow();
45             scene = new Scene(root);
46             stage.setScene(scene);
47             stage.show();
48
49         } catch (NumberFormatException e) {
50             myLabel.setText("Please enter a valid Customer ID.");
51         } catch (IllegalArgumentException e) {
52             myLabel.setText(e.getMessage());
53         } catch (IOException e) {
54             e.printStackTrace();
55         }
56     }
57 }
58
59
60 public void switchToScene1(ActionEvent event) throws IOException{
61     Parent root = FXMLLoader.Load(getClass().getResource("scene.fxml"));
62     stage = (Stage)((Node)event.getSource()).getScene().getWindow();
63     scene = new Scene(root);
64     stage.setScene(scene);
65     stage.show();
66 }
67 }
68 }
```

Controller2

```
1 package application;
2
3 import java.io.IOException;
4 public class Controller2 {
5
6     @FXML
7     Label label;
8     @FXML
9     Label label2;
10    @FXML
11    Button button;
12    @FXML
13    TextField nProductsField;
14    private String name;
15    private int customerID;
16    int nProducts;
17    void setAll(int customerID, String name){
18        this.customerID=customerID;
19        this.name=name;
20        label2.setText("Hello,"+name);
21    }
22
23    public void goToShop(ActionEvent event) {
24        try {nProducts = Integer.parseInt(nProductsField.getText().trim());
25            if (nProducts <= 0) {
26                throw new IllegalArgumentException("The number of products must be a positive integer");
27            }
28            FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene3.fxml"));
29            Parent root = loader.load();
30
31            Controller3 controller3 = loader.getController();
32            controller3.setNProducts(customerID,nProducts);
33            Stage stage = (Stage)((Node)event.getSource()).getScene().getWindow();
34            Scene scene = new Scene(root);
35            stage.setScene(scene);
36            stage.show();
37        }
38        catch (NumberFormatException e) {
39            label.setText("Please enter a positive integer");
40        }
41        catch (IllegalArgumentException e) {
42            label.setText(e.getMessage());
43        }
44        catch (IOException e) {
45            e.printStackTrace();
46        }
47    }
48 }
49
50
51
```

Controller3 pt1

```
1 package application;
2
3 import java.io.IOException;
4
17
18 public class Controller3 {
19     int customerID;
20     int nProducts;
21     Product[] products;
22     int productCount = 0;
23     Cart cart = new Cart();
24
26 Label label;
28 Button add;
30 ImageView smartphone;
32 ImageView tshirt;
34 ImageView oop;
36 RadioButton rButton1;
38 RadioButton rButton2;
40 RadioButton rButton3;
42 ToggleGroup productToggleGroup;
43
45 public void initialize() {}
48
49 public void setNProducts(int customerID, int nProducts) {
50     this.nProducts = nProducts;
51     this.products = new Product[nProducts];
52     this.customerID = customerID;
53     this.cart = new Cart(customerID, nProducts, products);
54 }
55
56 @FXML
57 public void handleRadioButtonAction(ActionEvent event) {
58     RadioButton selectedRadioButton = (RadioButton) productToggleGroup.getSelectedToggle();
59     if (selectedRadioButton != null) {
60         add.setDisable(false);
61     }
62 }
63 }
```

Controller3 pt2

```
63  
64 @FXML  
65 public void addToCart(ActionEvent event) {  
66     RadioButton selectedRadioButton = (RadioButton) productToggleGroup.getSelectedToggle();  
67     if (selectedRadioButton != null) {  
68         int productChoice = Integer.parseInt(selectedRadioButton.getId().substring(7));  
69         if (productCount < nProducts) {  
70             switch (productChoice) {  
71                 case 1:  
72                     products[productCount] = new ElectronicProduct(1, "Smartphone", 599.99f, "Samsung", 1);  
73                     break;  
74                 case 2:  
75                     products[productCount] = new ClothingProduct(2, "T-Shirt", 19.99f, "Medium", "Cotton");  
76                     break;  
77                 case 3:  
78                     products[productCount] = new BookProduct(3, "OOP", 39.99f, "O'Reilly", "X Publications");  
79                     break;  
80             }  
81             cart.addProduct(products[productCount]);  
82             if (productCount == nProducts - 1) {  
83                 switchToScene4();  
84             }  
85             productCount++;  
86         }  
87     }  
88     add.setDisable(true);  
89 }  
90  
91  
92 @FXML  
93 private void switchToScene4() {  
94     try {  
95         FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene4.fxml"));  
96         Parent root = loader.load();  
97  
98         Controller4 controller4 = loader.getController();  
99         controller4.setCart(cart);  
100  
101         Stage stage = (Stage) add.getScene().getWindow();  
102         Scene scene = new Scene(root);  
103         stage.setScene(scene);  
104         stage.show();  
105     } catch (IOException e) {  
106         e.printStackTrace();  
107     }  
108 }  
109 }  
110
```


Controller4 pt1

```
1 package application;
2
3 import java.io.IOException;
4
5 import javafx.event.ActionEvent;
6 import javafx.fxml.FXML;
7 import javafx.fxml.FXMLLoader;
8 import javafx.scene.Parent;
9 import javafx.scene.Scene;
10 import javafx.scene.control.Button;
11 import javafx.scene.control.Label;
12 import javafx.stage.Stage;
13
14 public class Controller4 {
15     private Cart cart;
16
17     public void setCart(Cart cart) {
18         this.cart = cart;
19     }
20
21     @FXML
22     private Label label;
23
24     @FXML
25     private Button yesButton;
26
27     @FXML
28     private Button noButton;
29
30     @FXML
31     public void initialize() {
32         if (label != null) {
33             label.setText("Would you like to place the Order?");
34             label.setStyle("-fx-font-size: 24px;");
35         } else {
36             System.err.println("Error: Label is not properly initialized.");
37         }
38     }
39
40     @FXML
41     private void handleYesButton() {
42         cart.placeOrder();
43         switchToScene5();
44     }
45
46     @FXML
47     private void handleNoButton() {
48         switchToScene6(cart);
49     }
50 }
```

Controller4 pt2

```
50  
51 @FXML  
52 private void switchToScene5() {  
53     try {  
54         FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene5.fxml"));  
55         Parent root = loader.load();  
56  
57         Controller5 controller5 = loader.getController();  
58         if (controller5 != null) {  
59             controller5.setOrderInfo(cart.orderInfo);  
60         } else {  
61             System.err.println("Error: Controller5 is null.");  
62         }  
63  
64         Stage stage = (Stage) yesButton.getScene().getWindow();  
65         Scene scene = new Scene(root);  
66         stage.setScene(scene);  
67         stage.show();  
68     } catch (IOException e) {  
69         e.printStackTrace();  
70     }  
71 }  
72 @FXML  
73  
74 private void switchToScene6(Cart cart) {  
75     //i put cart Cart cuz it makes a compiler error go away.  
76     try {  
77         FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene6.fxml"));  
78         Controller6 controller6 = new Controller6();  
79         controller6.setCart(cart);  
80         loader.setController(controller6);  
81         Parent root = loader.load();  
82  
83         Stage stage = (Stage) noButton.getScene().getWindow();  
84         Scene scene = new Scene(root);  
85         stage.setScene(scene);  
86         stage.show();  
87     } catch (IOException e) {  
88         e.printStackTrace();  
89     }  
90 }  
91  
92 }  
93
```


Controller5

```
1 package application;
2
3 import javafx.fxml.FXML;
4 import javafx.scene.control.Label;
5
6 public class Controller5 {
7     @FXML
8     private Label orderLabel;
9
10    public void setOrderInfo(String orderInfo) {
11        orderLabel.setText(orderInfo);
12    }
13 }
14
```

Controller6 pt1

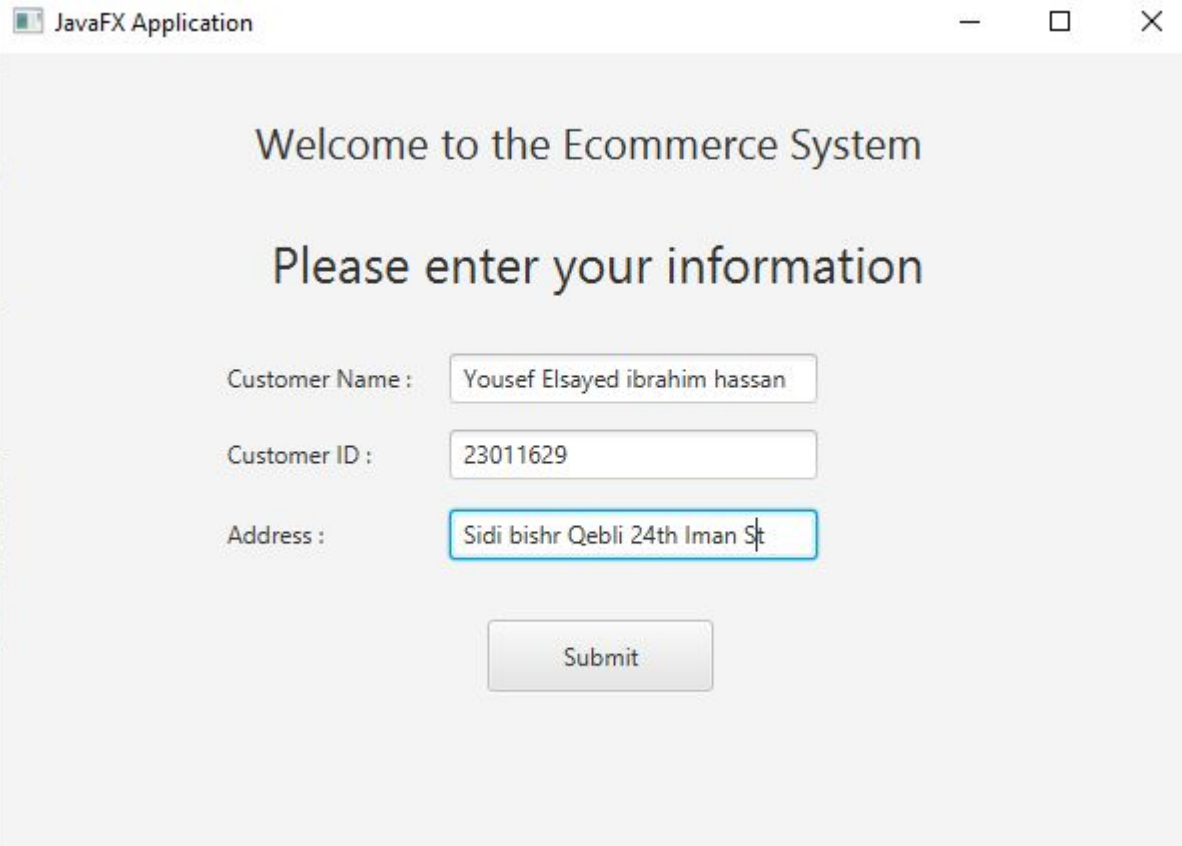
```
1 package application;
2
3 import javafx.event.ActionEvent;
13
14 public class Controller6 {
15
16     private Cart cart;
17
18     public void setCart(Cart cart) {
19         this.cart = cart;
20     }
21
22     @FXML
23     private ListView<String> productList;
24
25     @FXML
26     private Button confirmButton;
27
28     @FXML
29     private Button emptyCartButton;
30
31     @FXML
32     public void initialize() {
33         for (Product product : cart.getproducts()) {
34             productList.getItems().add(product.getname());
35         }
36     }
37
38     @FXML
39     private void handleConfirmButton(ActionEvent event) {
40         int selectedIndex = productList.getSelectionModel().getSelectedIndex();
41         if (selectedIndex >= 0 && selectedIndex < cart.getproducts().length) {
42             cart.removeProduct(cart.getproducts(), selectedIndex);
43         }
44         cart.placeOrder();
45         switchToScene5();
46     }
47
48     @FXML
49     private void handleEmptyCartButton(ActionEvent event) {
50         switchToScene7();
51     }
52 }
```

Controller6 pt2

```
52
53 @FXML
54 private void switchToScene5() {
55     try {
56         FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene5.fxml"));
57         Parent root = loader.load();
58
59         Controller5 controller5 = loader.getController();
60         if (controller5 != null) {
61             controller5.setOrderInfo(cart.orderInfo);
62         } else {
63             System.err.println("Error: Controller5 is null.");
64         }
65
66         Stage stage = (Stage) confirmButton.getScene().getWindow();
67         Scene scene = new Scene(root);
68         stage.setScene(scene);
69         stage.show();
70     } catch (IOException e) {
71         e.printStackTrace();
72     }
73 }
74
75 private void switchToScene7() {
76     try {
77         FXMLLoader loader = new FXMLLoader(getClass().getResource("Scene7.fxml"));
78         Parent root = loader.load();
79
80         Stage stage = (Stage) emptyCartButton.getScene().getWindow();
81         Scene scene = new Scene(root);
82         stage.setScene(scene);
83         stage.show();
84     } catch (IOException e) {
85         e.printStackTrace();
86     }
87 }
88 }
89
```

FXML Files After the output

Output:



The screenshot shows a JavaFX Application window with a light gray background. The window title bar includes the text "JavaFX Application" and standard window control buttons (minimize, maximize, close). The main content area contains the following elements:

- A welcome message: "Welcome to the Ecommerce System"
- A prompt: "Please enter your information"
- Three input fields with labels to their left:
 - "Customer Name :" followed by a text box containing "Yousef Elsayed ibrahim hassan"
 - "Customer ID :" followed by a text box containing "23011629"
 - "Address :" followed by a text box containing "Sidi bishr Qebli 24th Iman St"
- A "Submit" button centered below the input fields.

Hello, Yousef Elsayed ibrahim hass...

1 How many products would you like to order?

Go to shop

2

Please select which product you'd like to order



Smartphone



T-Shirt



OOP

Add to Cart

JavaFX Application



JavaFX Application

3

Please select which product you'd like to order



Smartphone



T-Shirt



OOP

Add to Cart

4

Please select which product you'd like to order



Smartphone



T-Shirt



OOP

Add to Cart

5

Please select which product you'd like to order



Smartphone



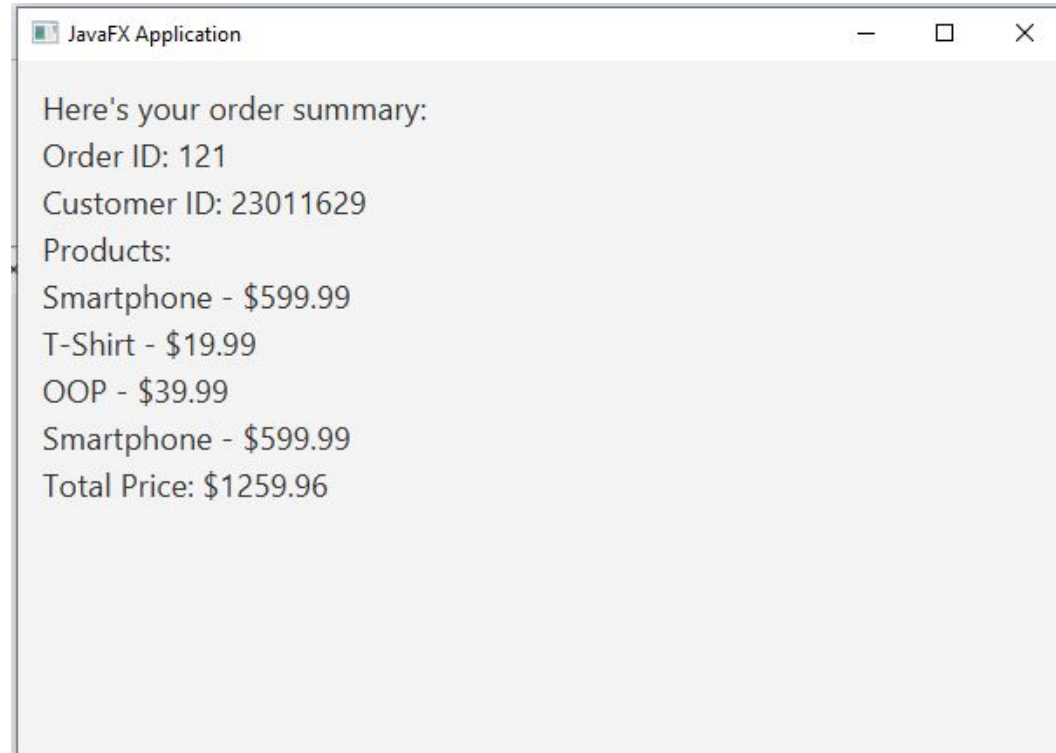
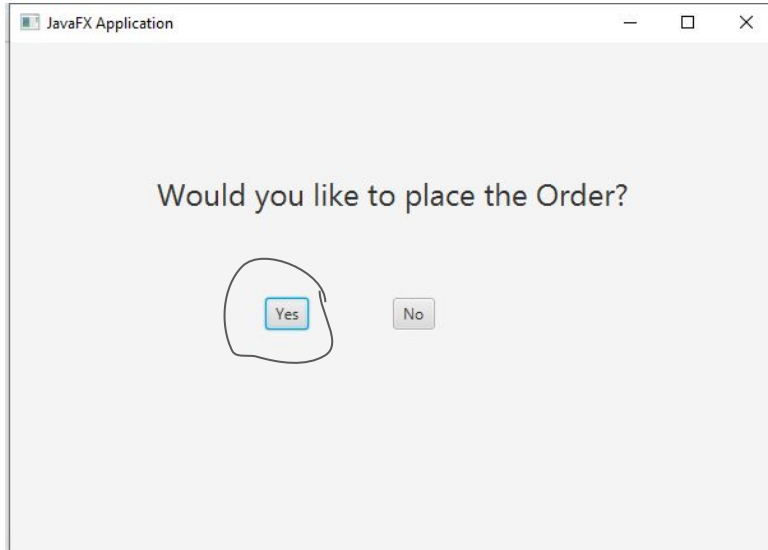
T-Shirt



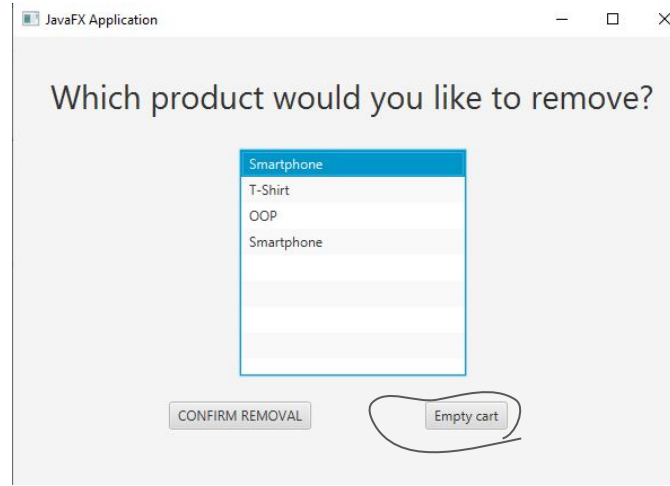
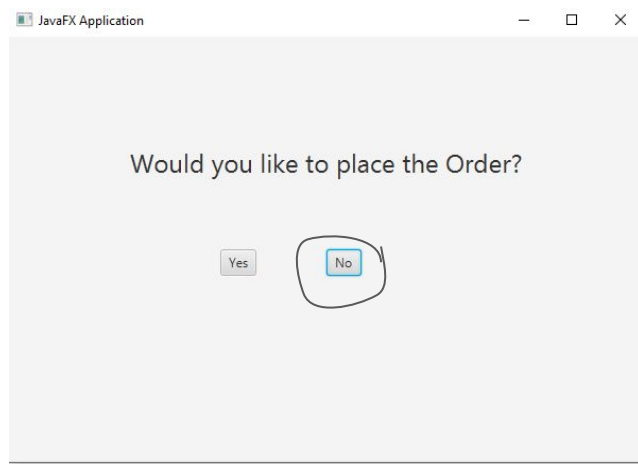
OOP

Add to Cart

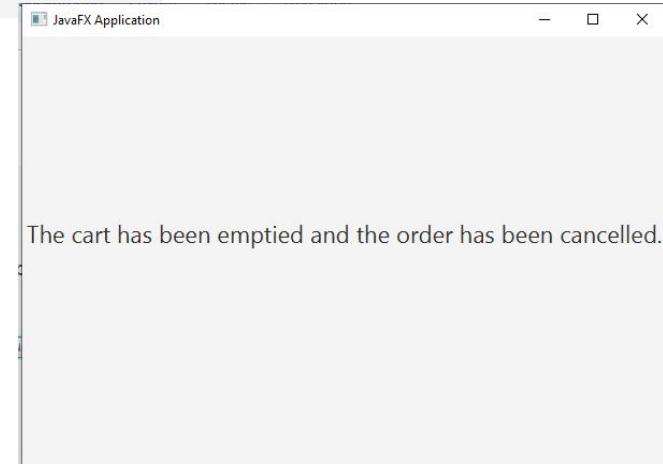
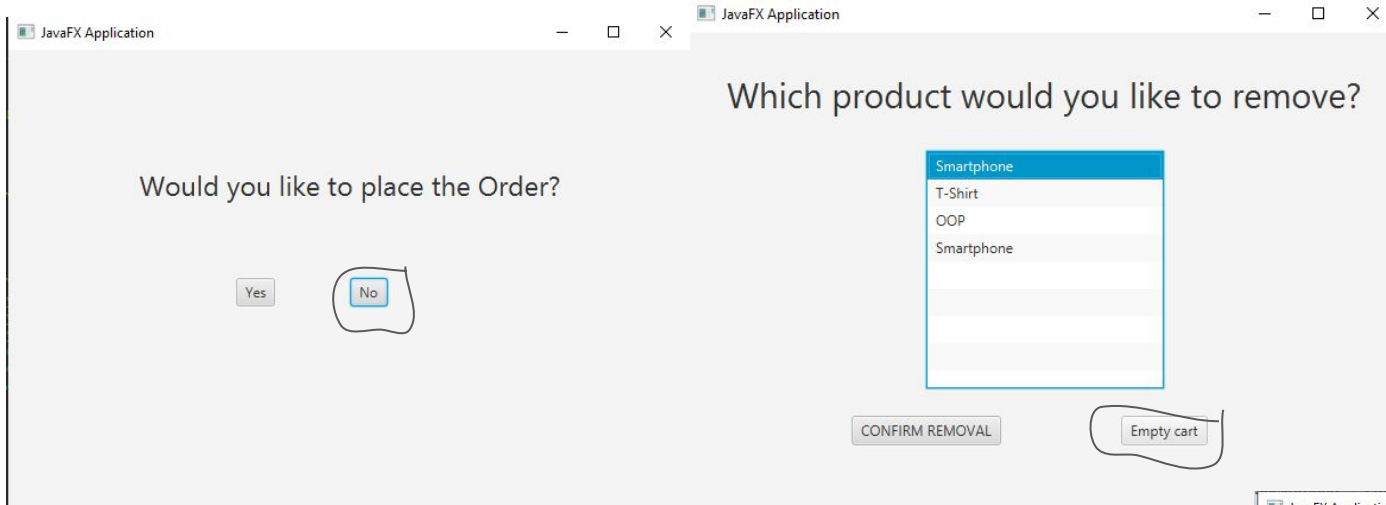
Output end 1 (Yes)



Output end 2(Remove product)



Output End 3 (EmptyCart)



FXML Files

scene.fxml

```
scene.fxml x scene2.fxml Scene3.fxml Scene4.fxml Scene5.fxml Scene6.fxml Scene7.fxml
1 <?xml version="1.0" encoding="UTF-8"?>
2   Ecomn
3   <?import javafx.scene.control.Button?>
4   <?import javafx.scene.control.Label?>
5   <?import javafx.scene.control.TextField?>
6   <?import javafx.scene.layout.AnchorPane?>
7   <?import javafx.scene.text.Font?>
8
9   Bind to grammar/schema...
10  <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0"
11  xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.Controller">
12    <children>
13      <Label layoutX="129.0" layoutY="14.0" prefHeight="62.0" prefWidth="344.0" text="Welcome to the Ecommerce System">
14        <font>
15          <Font size="21.0" />
16        </font>
17      </Label>
18      <Label fx:id="myLabel" layoutX="137.0" layoutY="88.0" text="Please enter your information">
19        <font>
20          <Font size="25.0" />
21        </font>
22      </Label>
23      <Button fx:id="button" layoutX="245.0" layoutY="283.0" mnemonicParsing="false" onAction="#submit" prefHeight="36.0" prefWidth="113.0" text="Submit" />
24      <TextField fx:id="TFieId1" layoutX="226.0" layoutY="150.0" prefHeight="25.0" prefWidth="184.0" promptText="ex: Ahmed Khalil" />
25      <TextField fx:id="TFieId2" layoutX="226.0" layoutY="188.0" prefHeight="25.0" prefWidth="184.0" promptText="ex: 20001292" />
26      <TextField fx:id="TFieId3" layoutX="226.0" layoutY="228.0" prefHeight="25.0" prefWidth="184.0" promptText="ex: Ezbet Saad" />
27      <Label layoutX="115.0" layoutY="154.0" text="Customer Name : " />
28      <Label layoutX="115.0" layoutY="192.0" text="Customer ID : " />
29      <Label layoutX="115.0" layoutY="232.0" text="Address : " />
30    </children>
31  </AnchorPane>
```

scene2.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Button?>
4 <?import javafx.scene.control.Label?>
5 <?import javafx.scene.control.TextField?>
6 <?import javafx.scene.layout.AnchorPane?>
7 <?import javafx.scene.text.Font?>
8
9 Bind to grammar/schema...
10 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0"
11   xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.Controller2">
12   <children>
13     <Label fx:id="Label" layoutX="62.0" layoutY="104.0" text="How many products would you like to order?">
14       <font>
15         <Font size="24.0" />
16       </font>
17     </Label>
18     <TextField fx:id="nProductsField" layoutX="217.0" layoutY="175.0" promptText="ex: 4" />
19     <Button fx:id="button" layoutX="254.0" layoutY="257.0" mnemonicParsing="false" onAction="#goToShop" text="Go to shop" />
20     <Label fx:id="Label2" layoutX="14.0" layoutY="26.0" prefHeight="53.0" prefWidth="571.0" text="Label">
21       <font>
22         <Font size="36.0" />
23       </font>
24     </Label>
25   </children>
26 </AnchorPane>
```

Scene3.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Button?>
4 <?import javafx.scene.control.Label?>
5 <?import javafx.scene.control.RadioButton?>
6 <?import javafx.scene.control.ToggleGroup?>
7 <?import javafx.scene.image.Image?>
8 <?import javafx.scene.image.ImageView?>
9 <?import javafx.scene.layout.AnchorPane?>
10 <?import javafx.scene.text.Font?>
11
12 Bind to grammar/schema...
13 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0"
14   prefWidth="600.0" xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1">
15   <children>
16     <Label fx:id="Label" layoutX="126.0" layoutY="37.0" text="Please select which product you'd like to order">
17       <font>
18         <Font size="17.0" />
19       </font>
20     </Label>
21     <RadioButton fx:id="rButton1" layoutX="14.0" layoutY="146.0" mnemonicParsing="false" onAction="#handleRadioButtonAction">
22       <toggleGroup>
23         <ToggleGroup fx:id="productToggleGroup" />
24       </toggleGroup>
25     </RadioButton>
26     <RadioButton fx:id="rButton2" layoutX="224.0" layoutY="147.0" mnemonicParsing="false"
27       onAction="#handleRadioButtonAction" toggleGroup="$productToggleGroup" />
28     <RadioButton fx:id="rButton3" layoutX="430.0" layoutY="147.0" mnemonicParsing="false"
29       onAction="#handleRadioButtonAction" toggleGroup="$productToggleGroup" />
30     <ImageView fx:id="smartphone" fitHeight="106.0" fitWidth="87.0" layoutX="46.0" layoutY="111.0"
31       pickOnBounds="true" preserveRatio="true">
32       <image>
33         <Image url="@Smartphone.png" />
34       </image>
35     </ImageView>
36     <ImageView fx:id="tshirt" fitHeight="106.0" fitWidth="87.0" layoutX="257.0" layoutY="111.0"
37       pickOnBounds="true" preserveRatio="true">
38       <image>
39         <Image url="@tshirt.png" />
40       </image>
41     </ImageView>
42     <ImageView fx:id="oop" fitHeight="106.0" fitWidth="87.0" layoutX="461.0" layoutY="111.0" pickOnBounds="true"
43       preserveRatio="true">
44       <image>
45         <Image url="@OOP.png" />
46       </image>
47     </ImageView>
48     <Button fx:id="add" layoutX="262.0" layoutY="292.0" mnemonicParsing="false" text="Add to Cart"
49       onAction="#addToCart" />
50   </children>
51 </AnchorPane>
52
```

Scene4.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Button?>
4 <?import javafx.scene.control.Label?>
5 <?import javafx.scene.layout.AnchorPane?>
6 <?import javafx.scene.text.Font?>
7
8 Bind to grammar/schema...
9 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0"
10 xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.Controller4">
11   <children>
12     <Label fx:id="label" layoutX="115.0" layoutY="102.0" text="Would you like to place the Order?">
13       <font>
14         <Font size="24.0" />
15       </font></Label>
16     <Button fx:id="yesButton" layoutX="200.0" layoutY="200.0" mnemonicParsing="false" onAction="#handleYesButton" text="Yes" />
17     <Button fx:id="noButton" layoutX="300.0" layoutY="200.0" mnemonicParsing="false" onAction="#handleNoButton" text="No" />
18   </children>
19 </AnchorPane>
```

Scene5.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Label?>
4 <?import javafx.scene.layout.AnchorPane?>
5 <?import javafx.scene.text.Font?>
6
7
8 Bind to grammar/schema...
9 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0"
10 | xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1" fx:controller="application.Controller5">
11 <children>
12 <Label fx:id="orderLabel" layoutX="14.0" layoutY="14.0" text="Label">
13 <font>
14 <Font size="18.0" />
15 </font>
16 </Label>
17 </children>
18 </AnchorPane>
```


Scene6.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Button?>
4 <?import javafx.scene.control.Label?>
5 <?import javafx.scene.control.ListView?>
6 <?import javafx.scene.layout.AnchorPane?>
7 <?import javafx.scene.text.Font?>
8
9 Bind to grammar/schema...
10 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0" prefWidth="600.0"
11 | xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1">
12 <children>
13   <Label fx:id="label" layoutX="33.0" layoutY="32.0" text="Which product would you like to remove?">
14     <font>
15       <Font size="29.0" />
16     </font>
17   </Label>
18   <ListView fx:id="productList" layoutX="201.0" layoutY="100.0" prefHeight="200.0" prefWidth="200.0" />
19   <Button fx:id="confirmButton" layoutX="138.0" layoutY="323.0" mnemonicParsing="false" onAction="#handleConfirmButton" text="CONFIRM REMOVAL" />
20   <Button fx:id="emptyCartButton" layoutX="365.0" layoutY="323.0" mnemonicParsing="false" onAction="#handleEmptyCartButton" text="Empty cart" />
21 </children>
22 </AnchorPane>
```

Scene7.fxml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <?import javafx.scene.control.Label?>
4 <?import javafx.scene.layout.AnchorPane?>
5 <?import javafx.scene.text.Font?>
6
7
8 Bind to grammar/schema...
9 <AnchorPane maxHeight="-Infinity" maxWidth="-Infinity" minHeight="-Infinity" minWidth="-Infinity" prefHeight="400.0"
10 prefWidth="600.0" xmlns="http://javafx.com/javafx/8.0.171" xmlns:fx="http://javafx.com/fxml/1">
11   <children>
12     <Label layoutX="4.0" layoutY="168.0" text="The cart has been emptied and the order has been cancelled.">
13       <font>
14         <Font size="22.0" />
15       </font>
16     </Label>
17   </children>
18 </AnchorPane>
```