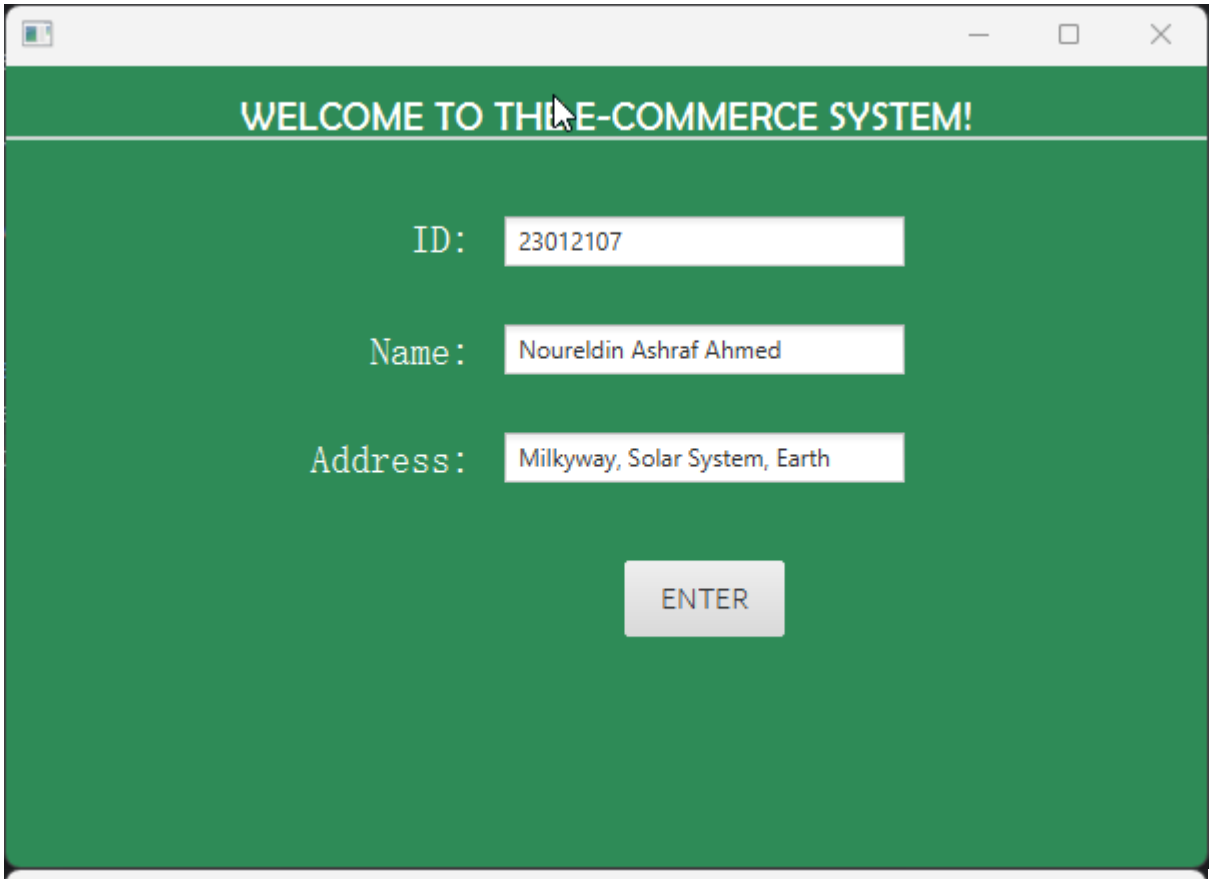
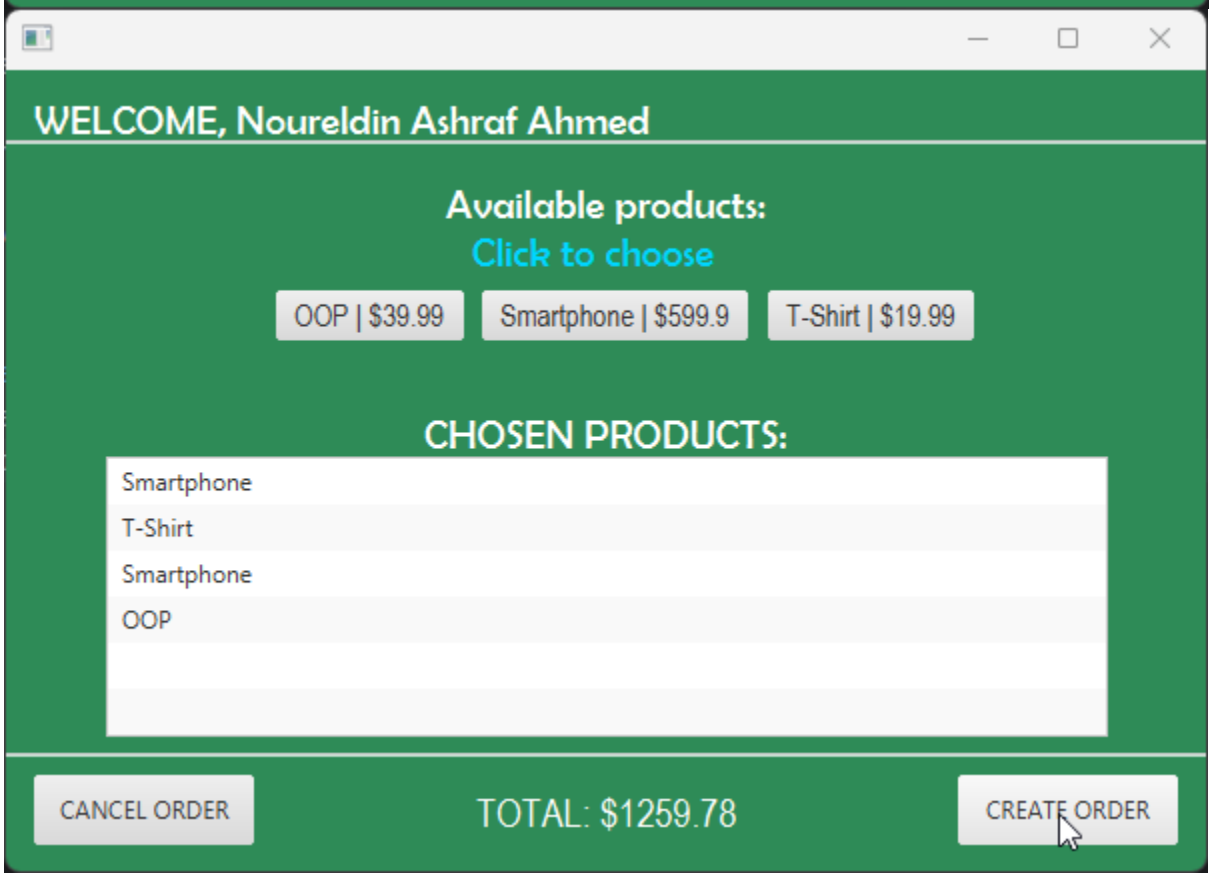
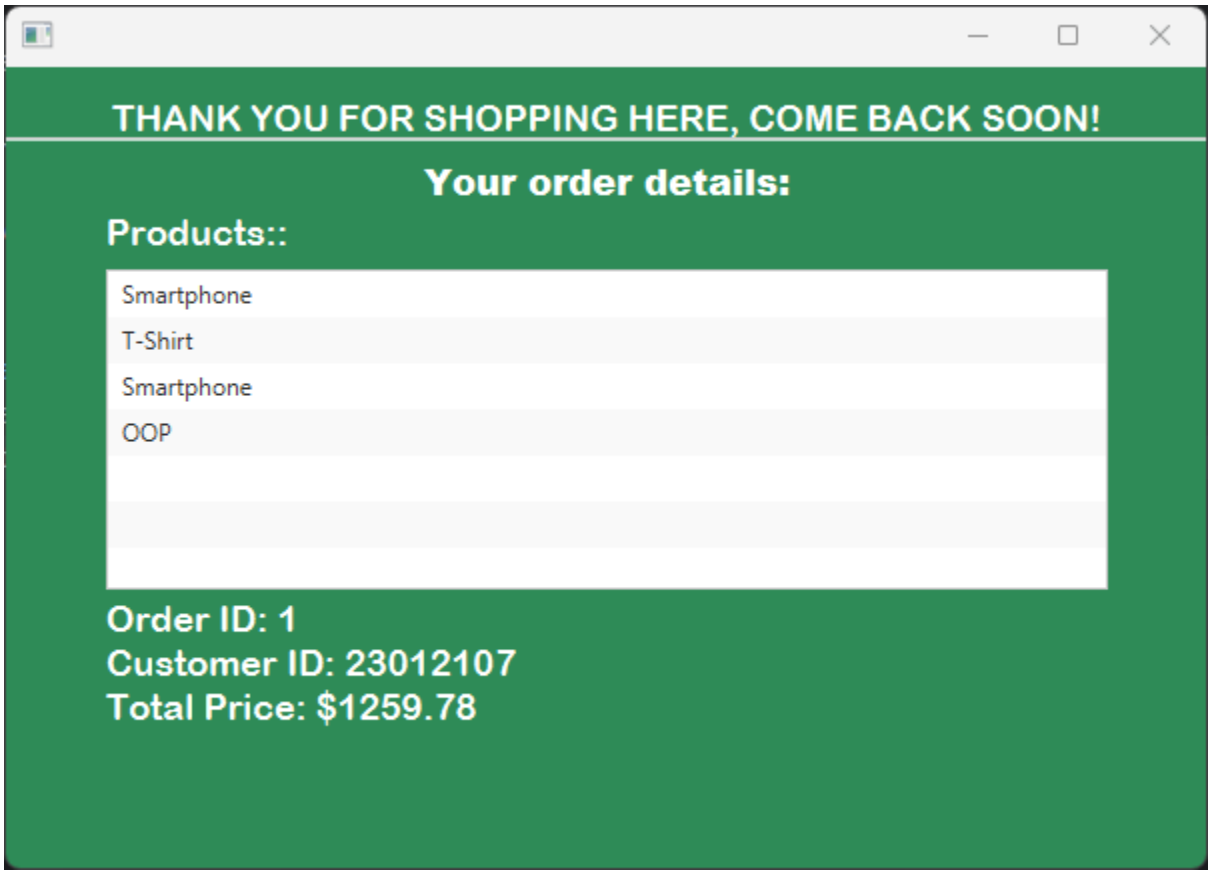
TEST CASE


WELCOME TO THE E-COMMERCE SYSTEM!
ID:
23012107
Name:
Noureldin Ashraf Ahmed
Address:
Milkyway, Solar System, Earth
ENTER


WELCOME, Noureldin Ashraf Ahmed
Available products:
Click to choose
OOP | $39.99
Smartphone | $599.9
T-Shirt | $19.99
CHOSEN PRODUCTS:
Smartphone
T-Shirt
Smartphone
OOP
CANCEL ORDER
TOTAL: $1259.78
CREATE ORDER


THANK YOU FOR SHOPPING HERE, COME BACK SOON!
Your order details:
Products::
Smartphone
T-Shirt
Smartphone
OOP
Order ID: 1
Customer ID: 23012107
Total Price: $1259.78

PRODUCT CLASS

```java
public class Product {
    private int productId;
    private String name;
    private double price;

    public Product() {}
    public Product(int productId, String name, double price) {
        if (productId < 0)
            this.productId = (-1 * productId);
        else
            this.productId = productId;

        this.name = name;

        if (price < 0)
            this.price = (-1 * price);
        else
            this.price = price;
    }

    public int getProductId() { return productId; }
    public void setProductId(int productId) {
        if (productId < 0)
            this.productId = (-1 * productId);
        else
            this.productId = productId;
    }

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

    public double getPrice() { return price; }
    public void setPrice(double price) {
        if (price < 0)
            this.price = (-1 * price);
        else
            this.price = price;
    }
}
```

ELECTRONIC PRODUCT CLASS

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

    public ElectronicProduct() {}
    public ElectronicProduct(int productId, String name, double price, String brand, int warrantyPeriod) {
        super(productId, name, price);
        this.brand = brand;

        if (warrantyPeriod <0)
            this.warrantyPeriod = (-1 * warrantyPeriod);
        else
            this.warrantyPeriod = warrantyPeriod;
    }

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

    public int getWarrantyPeriod() { return warrantyPeriod; }
    public void setWarrantyPeriod(int warrantyPeriod) {
        if (warrantyPeriod <0)
            this.warrantyPeriod = (-1 * warrantyPeriod);
        else
            this.warrantyPeriod = warrantyPeriod;
    }
}
```

CLOTHING PRODUCT CLASS

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

    public ClothingProduct() {}
    public ClothingProduct(int productId, String name, double price, String size, String fabric) {
        super(productId, name, price);
        this.size = size;
        this.fabric = fabric;
    }

    public String getSize() { return size; }
    public void setSize(String size) { this.size = size; }

    public String getFabric() { return fabric; }
    public void setFabric(String fabric) { this.fabric = fabric; }
}
```

BOOK PRODUCT CLASS

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

    public BookProduct() {}
    public BookProduct(int productId, String name, double price, String author, String publisher) {
        super(productId, name, price);
        this.author = author;
        this.publisher = publisher;
    }

    public String getAuthor() { return author; }
    public void setAuthor(String author) { this.author = author; }

    public String getPublisher() { return publisher; }
    public void setPublisher(String publisher) { this.publisher = publisher; }
}
```

CUSTOMER CLASS

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

    public Customer(){}
    public Customer(int customerId, String name, String address) {
        if (customerId < 0)
            this.customerId = (-1 * customerId);
        else
            this.customerId = customerId;

        this.name = name;
        this.address = address;
    }

    public int getCustomerId() { return customerId; }
    public void setCustomerId(int customerId) {
        if (customerId < 0)
            this.customerId = (-1 * customerId);
        else
            this.customerId = customerId;
    }

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

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

CART CLASS

```java
public class Cart {
    private int customerId;
    private ArrayList<Product> products;

    public Cart(int customerId) {
        if (customerId < 0)
            this.customerId = (-1 * customerId);
        else
            this.customerId = customerId;

        products = new ArrayList<>();
    }

    public void addProduct(Product product) { products.add(product); }
    public void removeProduct(Product product) { products.remove(product); }
    public void removeAllProducts() { products.clear(); }
    public int getProductsAmount(){return products.size();}

    public double calculatePrice(){
        double totalPrice = 0;
        for (Product i : products){
            totalPrice+=i.getPrice();
        }
        return totalPrice;
    }

    public Order placeOrder(){
        Order order = new Order(customerId, orderId: 1, products, calculatePrice());
        return order;
    }

    public int getCustomerId() { return customerId; }
    public void setCustomerId(int customerId) {
        if (customerId < 0)
            this.customerId = (-1 * customerId);
        else
            this.customerId = customerId;
    }
}
```

ORDER CLASS

```java
public class Order {
    private int customerId;
    private int orderId;
    private ArrayList<Product> products;
    private double totalPrice;

    public Order(int customerId, int orderId, ArrayList<Product> products, double totalPrice){
        if (customerId < 0)
            this.customerId = (-1 * customerId);
        else
            this.customerId = customerId;

        if (orderId < 0)
            this.orderId = (-1 * orderId);
        else
            this.orderId = orderId;

        this.products = products;

        if (totalPrice < 0)
            this.totalPrice = (-1 * totalPrice);
        else
            this.totalPrice = totalPrice;
    }

    public String printOrderInfo(){
        String s = "Order ID: " + orderId+
        "\nCustomer ID: " + customerId+
        "\nTotal Price: $" + totalPrice;
        return s;
    }
    public ArrayList<String> getProducts(){
        ArrayList<String> productNames = new ArrayList<>();
        for (Product i : products){
            productNames.add(i.getName());
        }
        return productNames;
    }
}
```

ECOMMERCE SYSTEM CLASS

```java
public class EcommerceSystem extends Application {
    public static ElectronicProduct smartphone;
    public static ClothingProduct tShirt;
    public static BookProduct OOP;
    public static Customer customer;
    public static Cart cart;
    public static Order order;
    private static Scene scene;

    public static void main(String[] args) { launch(); }


    @Override
    public void start(Stage stage) throws IOException {
        scene = new Scene(loadFXML("primary"));
        stage.setScene(scene);
        stage.show();
    }

    public static void setRoot(String fxml) throws IOException {
        scene.setRoot(loadFXML(fxml));
    }

    private static Parent loadFXML(String fxml) throws IOException {
        FXMLLoader fxmlLoader = new FXMLLoader(EcommerceSystem.class.getResource( name: fxml + ".fxml"));
        return fxmlLoader.load();
    }

}
```

CLASSES REQUIRED FOR JAVAFX:

PRIMARYCONTROLLER CLASS

```
27  public class PrimaryController {
28      @FXML
29  </>     private Label errorMessage;
30      @FXML
31      private Button enterButton;
32      @FXML
33  </>     private TextField addressTextfield;
34      @FXML
35  </>     private TextField idTextfield;
36      @FXML
37  </>     private TextField nameTextfield;
38      @FXML
39      public void initialize() throws IOException {
40          smartphone = new ElectronicProduct( productId: 1,  name: "Smartphone",  price: 599.9,  brand: "Samsung",  warrantyPeriod: 1);
41          tShirt = new ClothingProduct( productId: 2,  name: "T-Shirt",  price: 19.99,  size: "Medium",  fabric: "Cotton");
42          OOP = new BookProduct( productId: 3,  name: "OOP",  price: 39.99,  author: "O'Reilly",  publisher: "X Publications");
43      }
44      @FXML
45      void onClickEnter(ActionEvent event) throws IOException {
46
47          if (!hasErrors()){
48              int id = Integer.parseInt(idTextfield.getText());
49              String name = nameTextfield.getText();
50              String address = addressTextfield.getText();
51              customer = new Customer(id, name, address);
52              cart = new Cart(id);
53              switchToSecondary();
54          }
```

CONTINUE THIS CLASS IN NEXT PAGE...

```
    private boolean hasErrors(){
        //CHECK IF EMPTY
        String id = idTextfield.getText();
        String address = addressTextfield.getText();
        String name = nameTextfield.getText();
        if (name.isEmpty() || address.isEmpty() || id.isEmpty()){
            errorMessage.setText("ERROR: FILL IN YOUR INFORMATION!");
            return true;
        }
        if(!isNumeric(id)){
            errorMessage.setText("ERROR: ENTER AN APPROPRIATE NUMERIC ID!");
            return true;
        }
        return false;
    }
    private boolean isNumeric(String str) {
        if (str == null) {
            return false;
        }
        try {
            double d = Integer.parseInt(str);
        } catch (NumberFormatException nfe) {
            return false;
        }
        return true;
    }
    private void switchToSecondary() throws IOException {
        EcommerceSystem.setRoot("secondary");
    }
}
```

SECONDARYCONTROLLER CLASS

```java
public class SecondaryController {
    @FXML
    private ListView<String> productListView;
    @FXML
    private Label welcomeMessage;
    @FXML
    private Label totalMessage;

    @FXML
    public void initialize() throws IOException {
        String welcomemsg = welcomeMessage.getText().replaceAll( regex: "replace", customer.getName());
        welcomeMessage.setText(welcomemsg);
    }
    @FXML
    void onClickOOP(ActionEvent event) {
        cart.addProduct(OOP);
        addProductToViewList(OOP.getName());
        totalMessage.setText("TOTAL: $"+cart.calculatePrice());
    }
    @FXML
    void onClickSmartphone(ActionEvent event) {
        cart.addProduct(smartphone);
        addProductToViewList(smartphone.getName());
        totalMessage.setText("TOTAL: $"+cart.calculatePrice());
    }
    @FXML
    void onClickTshirt(ActionEvent event) {
        cart.addProduct(tShirt);
        addProductToViewList(tShirt.getName());
        totalMessage.setText("TOTAL: $"+cart.calculatePrice());
    }
    @FXML
    void onClickCancelOrder(ActionEvent event) {
        cart.removeAllProducts();
        productListView.getItems().clear();
        totalMessage.setText("TOTAL: $"+cart.calculatePrice());
    }
    @FXML
    void onClickCreateOrder(ActionEvent event) throws IOException {
        order = cart.placeOrder();
        switchToShow();
    }
    private void addProductToViewList(String productName){
        productListView.getItems().add(productName);
    }
    private void switchToShow() throws IOException {
        EcommerceSystem.setRoot("show");
    }
}
```

SHOWCONTROLLER CLASS

```java
public class ShowController {
    @FXML
    private ListView<String> productListView;
    @FXML
    private Label detailsMessage;

    @FXML
    public void initialize() throws IOException {
        productListView.getItems().addAll(order.getProducts());
        detailsMessage.setText(order.printOrderInfo());
    }
}
```