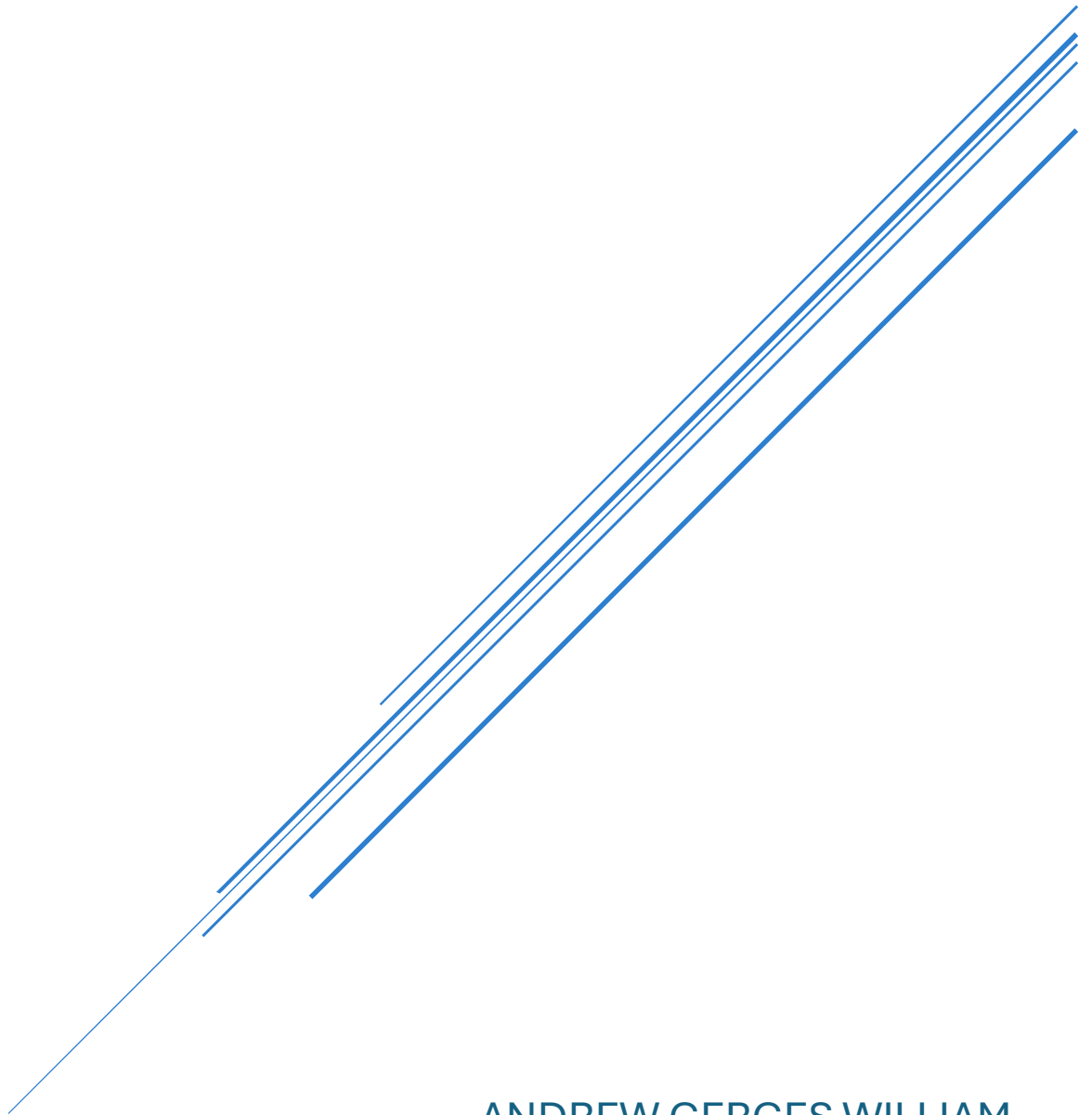


JAVA PROJECT

E-commerce Program



ANDREW GERGES WILLIAM

23010145

```
Product.java x ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2
3 public class Product {
4     protected int productId;
5     protected String name;
6     protected float price;
7
8     public Product(int productId, String name, float price) {
9         this.productId = productId;
10        this.name = name;
11        this.price = price;
12    }
13
14    public int getProductId() { return productId; }
15
16    public void setProductId(int productId) { this.productId = productId; }
17
18    public String getName() { return name; }
19
20    public void setName(String name) { this.name = name; }
21
22    public float getPrice() { return price; }
23
24    public void setPrice(float price) { this.price = price; }
25 }
```

```
Product.java ElectronicProduct.java x ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2 public class ElectronicProduct extends Product {
3     private String brand;
4     private int warrantyPeriod;
5     public ElectronicProduct(int productId, String name, float price, String brand, int warrantyPeriod) {
6         super(productId, name, price);
7         this.brand = brand;
8         this.warrantyPeriod = warrantyPeriod;
9     }
10
11    public String getBrand() {
12        return brand;
13    }
14
15    public void setBrand(String brand) {
16        this.brand = brand;
17    }
18
19    public int getWarrantyPeriod() {
20        return warrantyPeriod;
21    }
22
23    public void setWarrantyPeriod(int warrantyPeriod) {
24        this.warrantyPeriod = warrantyPeriod;
25    }
26 }
```

```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2
3 4 usages
4 public class ClothingProduct extends Product {
5     3 usages
6     private String size;
7     3 usages
8     private String fabric;
9     2 usages
10    public ClothingProduct(int productId, String name, float price, String size, String fabric) {
11        super(productId, name, price);
12    }
13    this.size = size;
14    this.fabric = fabric;
15 }
16
17 no usages
18 public String getSize() {
19     return size;
20 }
21
22 no usages
23 public void setSize(String size) {
24     this.size = size;
25 }
26
27 no usages
28 public String getFabric() {
29     return fabric;
30 }
31
32 no usages
33 public void setFabric(String fabric) {
34     this.fabric = fabric;
35 }
36 }
```

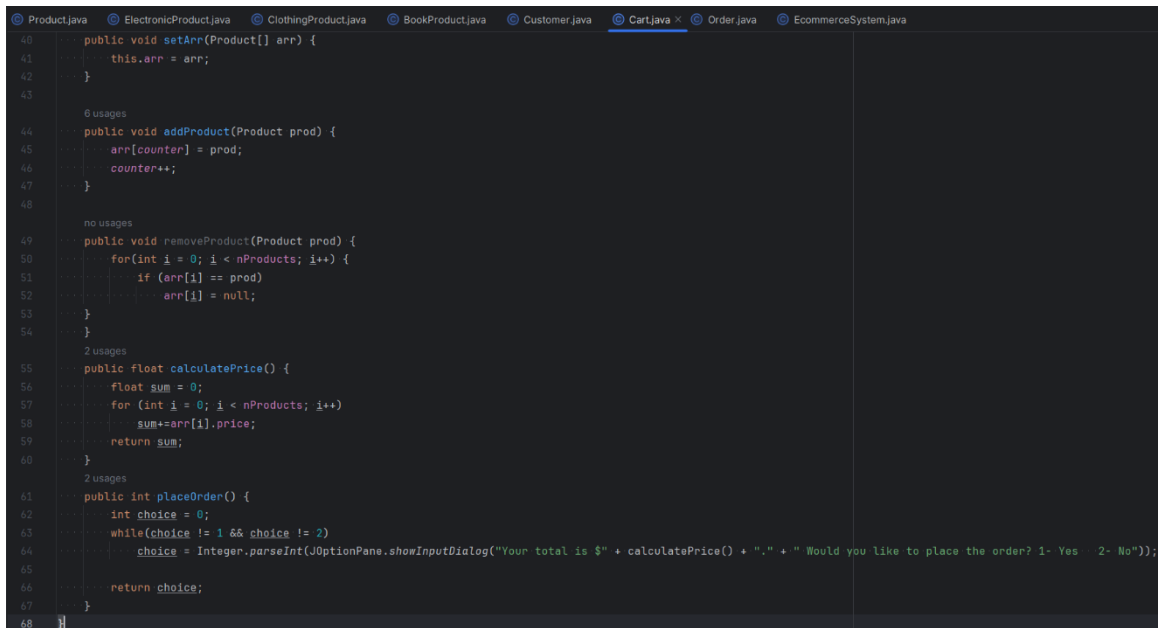
```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2
3 4 usages
4 public class BookProduct extends Product {
5     3 usages
6     private String author;
7     3 usages
8     private String publisher;
9
10    2 usages
11    public BookProduct(int productId, String name, float price, String author, String publisher) {
12        super(productId, name, price);
13    }
14    this.author = author;
15    this.publisher = publisher;
16 }
17
18 no usages
19 public String getAuthor() {
20     return author;
21 }
22
23 no usages
24 public void setAuthor(String author) {
25     this.author = author;
26 }
27
28 no usages
29 public String getPublisher() {
30     return publisher;
31 }
32
33 no usages
34 public void setPublisher(String publisher) {
35     this.publisher = publisher;
36 }
37 }
```

```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java x Cart.java Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2
3 public class Customer {
4     private int customerId;
5     private String name;
6     private String address;
7
8     public Customer(int customerId, String name, String address) {
9         this.customerId = customerId;
10        this.name = name;
11        this.address = address;
12    }
13
14    public int getCustomerId() {
15        return customerId;
16    }
17
18    public void setCustomerId(int customerId) {
19        this.customerId = customerId;
20    }
21
22    public String getName() {
23        return name;
24    }
25
26    public void setName(String name) {
```

```
22    public String getName() {
23        return name;
24    }
25
26    public void setName(String name) {
27        this.name = name;
28    }
29
30    public String getAddress() {
31        return address;
32    }
33
34    public void setAddress(String address) {
35        this.address = address;
36    }
37 }
38
```

```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java × Order.java EcommerceSystem.java
1 package JavaTerm2Project;
2
3 import javax.swing.*;
4 import java.util.Scanner;
5
6 public class Cart {
7     Scanner input = new Scanner(System.in);
8
9     private int customerId;
10    private int nProducts;
11    private Product[] arr = new Product[nProducts];
12    private static int counter = 0;
13    private static int orderId;
14
15    public Cart(int customerId, int nProducts) {
16        this.customerId = customerId;
17        this.nProducts = nProducts;
18    }
19
20    public int getCustomerId() {
21        return customerId;
22    }
23
24    public void setCustomerId(int customerId) {
25        this.customerId = customerId;
26    }
27
28    public int get_nProducts() {
29        return nProducts;
30    }
31
32    public void set_nProducts(int nProducts) {
33        this.nProducts = nProducts;
34    }
35
36    public Product[] getArr() {
37        return arr;
38    }
39
40    public void setArr(Product[] arr) {
41        this.arr = arr;
42    }
43
44    public void addProduct(Product prod) {
45        arr[counter] = prod;
46    }
47
48    public void displayCart() {
49        for (int i = 0; i < arr.length; i++) {
50            System.out.println("Product " + (i + 1) + ": " + arr[i].toString());
51        }
52    }
53
54    public void calculateTotal() {
55        double total = 0;
56        for (int i = 0; i < arr.length; i++) {
57            total += arr[i].getPrice();
58        }
59        System.out.println("Total Price: " + total);
60    }
61
62    public void checkout() {
63        System.out.println("Checking out...");
64        calculateTotal();
65        displayCart();
66    }
67
68    public void run() {
69        while (true) {
70            System.out.println("1. Add Product\n2. Display Cart\n3. Calculate Total\n4. Checkout\n5. Exit\nEnter choice: ");
71            int choice = input.nextInt();
72            switch (choice) {
73                case 1: addProduct(new Product(input.nextInt(), input.next())); break;
74                case 2: displayCart(); break;
75                case 3: calculateTotal(); break;
76                case 4: checkout(); break;
77                case 5: System.exit(0); break;
78                default: System.out.println("Invalid choice");
79            }
80        }
81    }
82
83    public static void main(String[] args) {
84        Cart cart = new Cart(input.nextInt(), input.nextInt());
85        cart.run();
86    }
87
88    public static void main(String[] args) {
89        Cart cart = new Cart(input.nextInt(), input.nextInt());
90        cart.run();
91    }
92
93    public static void main(String[] args) {
94        Cart cart = new Cart(input.nextInt(), input.nextInt());
95        cart.run();
96    }
97
98    public static void main(String[] args) {
99        Cart cart = new Cart(input.nextInt(), input.nextInt());
100       cart.run();
101   }
```

```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java × Order.java EcommerceSystem.java
20 public int getCustomerId() {
21     return customerId;
22 }
23
24 public void setCustomerId(int customerId) {
25     this.customerId = customerId;
26 }
27
28 public int get_nProducts() {
29     return nProducts;
30 }
31
32 public void set_nProducts(int nProducts) {
33     this.nProducts = nProducts;
34 }
35
36 public Product[] getArr() {
37     return arr;
38 }
39
40 public void setArr(Product[] arr) {
41     this.arr = arr;
42 }
43
44 public void addProduct(Product prod) {
45     arr[counter] = prod;
46     counter++;
47 }
48
49 public void displayCart() {
50     for (int i = 0; i < arr.length; i++) {
51         System.out.println("Product " + (i + 1) + ": " + arr[i].toString());
52     }
53 }
54
55 public void calculateTotal() {
56     double total = 0;
57     for (int i = 0; i < arr.length; i++) {
58         total += arr[i].getPrice();
59     }
60     System.out.println("Total Price: " + total);
61 }
62
63 public void checkout() {
64     System.out.println("Checking out...");
65     calculateTotal();
66     displayCart();
67 }
68
69 public void run() {
70     while (true) {
71         System.out.println("1. Add Product\n2. Display Cart\n3. Calculate Total\n4. Checkout\n5. Exit\nEnter choice: ");
72         int choice = input.nextInt();
73         switch (choice) {
74             case 1: addProduct(new Product(input.nextInt(), input.next())); break;
75             case 2: displayCart(); break;
76             case 3: calculateTotal(); break;
77             case 4: checkout(); break;
78             case 5: System.exit(0); break;
79             default: System.out.println("Invalid choice");
80         }
81     }
82 }
83
84 public static void main(String[] args) {
85     Cart cart = new Cart(input.nextInt(), input.nextInt());
86     cart.run();
87 }
88
89 public static void main(String[] args) {
90     Cart cart = new Cart(input.nextInt(), input.nextInt());
91     cart.run();
92 }
93
94 public static void main(String[] args) {
95     Cart cart = new Cart(input.nextInt(), input.nextInt());
96     cart.run();
97 }
98
99 public static void main(String[] args) {
100    Cart cart = new Cart(input.nextInt(), input.nextInt());
101    cart.run();
102 }
```



```
Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java
40 public void setArr(Product[] arr) {
41     this.arr = arr;
42 }
43
44 6 usages
45 public void addProduct(Product prod) {
46     arr[counter] = prod;
47     counter++;
48 }
49
50 no usages
51 public void removeProduct(Product prod) {
52     for(int i = 0; i < nProducts; i++) {
53         if (arr[i] == prod)
54             arr[i] = null;
55     }
56 }
57
58 2 usages
59 public float calculatePrice() {
60     float sum = 0;
61     for (int i = 0; i < nProducts; i++)
62         sum+=arr[i].price;
63     return sum;
64 }
65
66 2 usages
67 public int placeOrder() {
68     int choice = 0;
69     while(choice != 1 && choice != 2)
70         choice = Integer.parseInt(JOptionPane.showInputDialog("Your total is $" + calculatePrice() + "," + " Would you like to place the order? 1- Yes 2- No"));
71     return choice;
72 }
73 }
```

```

Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java x EcommerceSystem.java
1 package JavaTerm2Project;
2
3 import javax.swing.*;
4
5 public class Order {
6     private int customerId;
7     private static int orderId = 0;
8     private int nProducts;
9     private Product[] prod = new Product[nProducts];
10    private float totalPrice;
11
12    public Order(int customerId, Product[] prod) {
13        this.customerId = customerId;
14        this.prod = prod;
15    }
16
17    public static int getOrderId() {
18        return orderId;
19    }
20
21    public void printOrderInfo(Cart cart) {
22        orderId++;
23        // Create JFrame
24        JFrame frame = new JFrame( title: "Order");
25
26        // Create JPanel for order details
27        JPanel panel = new JPanel();
28    }

```

```

Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java x EcommerceSystem.java
19 // Create JFrame
20 JFrame frame = new JFrame( title: "Order");
21
22 // Create JPanel for order details
23 JPanel panel = new JPanel();
24
25 JTextArea orderDetails = new JTextArea( rows: 10, columns: 30); // Order details
26 orderDetails.setEditable(false); // makes it non-editable
27
28 String order = "";
29 for (int i = 0; i < cart.getNProducts(); i++) {
30     order = order + cart.getArr()[i].name + " - $" + cart.getArr()[i].price + '\n';
31 }
32
33 String orderId = "Order ID: " + orderId + '\n';
34 String customerId = "Customer ID: " + cart.getCustomerId() + '\n';
35 String products = "Products: \n";
36 String totalPrice = "Total Price: $" + cart.calculatePrice() + '\n'; // Order details
37
38 orderDetails.setText(orderID + customerId + products + order + totalPrice + "Thanks for shopping!"); // Set order details text
39
40 panel.add(orderDetails); // Add order details text area to the panel
41
42 frame.getContentPane().add(panel); // Add panel to the frame
43 ImageIcon image = new ImageIcon( filename: "C:\\Users\\HP\\Downloads\\icon-Copy-2.jpg");
44 frame.setIconImage(image.getImage());
45 frame.setSize( width: 400, height: 300);
46 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
47 frame.setVisible(true); // Set frame properties
48 }
49 }
50

```

```

Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java x
1 package JavaTerm2Project;
2 import javax.swing.*;
3 import java.util.Scanner;
4 public class EcommerceSystem {
5     public static void main(String[] args) {
6         Scanner input = new Scanner(System.in);
7
8         ElectronicProduct smartphone = new ElectronicProduct( productid: 1, name: "smartphone", price: 599.9f, brand: "Samsung", warrantyPeriod: 1);
9         ClothingProduct tShirt = new ClothingProduct( productid: 2, name: "T-Shirt", price: 19.99f, size: "Medium", fabric: "Cotton");
10        BookProduct OOP = new BookProduct( productid: 3, name: "OOP", price: 39.99f, author: "O'Reilly", publisher: "X Publications"); //Test products
11
12        int id;
13        String name, address;
14
15        id = Math.abs(Integer.parseInt(JOptionPane.showInputDialog("Welcome to the store! Please enter your ID")));
16        name = JOptionPane.showInputDialog("Please enter your name");
17        address = JOptionPane.showInputDialog("Please enter your address");
18
19        Customer cust = new Customer(id, name, address);
20
21        int count = Integer.parseInt(JOptionPane.showInputDialog("How many products do you want to add to your cart?"));
22
23        Cart cart = new Cart(id, count);
24        cart.set_nProducts(count);
25        Product[] products = new Product[count];
26        cart.setArr(products); // Setting up the cart of products
27
28        for (int i = 0; i < count; i++) {
29            int choice = 0;
30
31            while (choice > 3 || choice < 1)
32                choice = Integer.parseInt(JOptionPane.showInputDialog("Which product to add? 1- Smartphone 2- T-Shirt 3-OOP"));
33            // Picking what products to add to cart

```

```

Product.java ElectronicProduct.java ClothingProduct.java BookProduct.java Customer.java Cart.java Order.java EcommerceSystem.java x
31            while (choice > 3 || choice < 1)
32                choice = Integer.parseInt(JOptionPane.showInputDialog("Which product to add? 1- Smartphone 2- T-Shirt 3-OOP"));
33            // Picking what products to add to cart
34            if (choice == 1)
35                cart.addProduct(smartphone);
36            else if (choice == 2)
37                cart.addProduct(tShirt);
38            else
39                cart.addProduct(OOP); // Adding the product to cart
40        }
41
42        Order order = new Order(id, cart.getArr()); // Setting up the order
43        int placeOrder = cart.placeOrder();
44
45        JFrame frame = new JFrame( title: "Thank you");
46        JPanel panel = new JPanel();
47
48        frame.setSize( width: 400, height: 300);
49        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
50        ImageIcon image = new ImageIcon( filename: "C:\\Users\\HP\\Downloads\\icon-Copy-2.jpg");
51        JTextArea thanks = new JTextArea( rows: 10, columns: 30);
52        thanks.setEditable(false);
53        thanks.setText("Thanks for shopping!");
54        panel.add(thanks);
55        frame.getContentPane().add(panel);
56        frame.setIconImage(image.getImage()); // Text to show in case customer does not order
57
58        if (placeOrder == 1)
59            order.printOrderInfo(cart); // Final receipt of the added products
60        else
61            frame.setVisible(true); // Frame properties
62    }
63 }

```