

Project Overview

1. Project Tree

2. Explanation of Directories

1. **src** : Contains your Java source code files organized by packages.
 - **com.ecommerce** : Includes `Customer.java` and `Product.java`, which define the e-commerce logic.
 - **com.ecommerce.orders** : Contains `Order.java`, responsible for managing customer orders.
 - **Main.java** : The entry point of the application, where all program operations are executed.
 - **module-info.java** : Contains module definitions if your project is modularized (optional for basic projects).
 2. **bin** : Contains compiled `.class` files organized into the same package structure as **src**. You execute the program from here using the `java com.Main` command.
-

Explanation of Classes

Each class plays a specific role in the e-commerce application:

1. Product Class

- Represents a product available in the store.
 - Includes attributes: `productID`, `name`, and `price`.
 - Provides getters to access product information and overrides `toString` for easy display.
-

2. Customer Class

- Represents a customer with a `name`, `customerID`, and a `shoppingCart` (list of products).
 - Methods:
 - `addToCart(Product product)` : Adds a product to the shopping cart.
 - `removeFromCart(Product product)` : Removes a product from the shopping cart.
 - `calculateTotalCost()` : Calculates the total price of all items in the shopping cart.
 - The shopping cart is managed as a `List<Product>`.
-

3. Order Class

- Represents an order placed by a customer.
 - Attributes:
 - `orderId` : Unique identifier for the order.
 - `customer` : The customer placing the order.
 - `products` : List of products in the shopping cart.
 - `orderTotal` : Total cost of the order.
 - Methods:
 - `generateOrderSummary()` : Returns a formatted string with the order details.
-

4. Main Class

This is the entry point of the application. The program uses this class to interact with the user via a menu system.

Program Flow

Here's how the program runs:

1. Initialization:

- The program starts by creating a list of sample products in the store.
- It prompts the user for their name and creates a `Customer` instance to represent them.

2. Main Menu:

- A menu is displayed, allowing the user to:
 1. Browse available products.

2. View items in their shopping cart.
3. Add a product to their cart.
4. Remove a product from their cart.
5. Place an order.
6. Exit the program.

3. Interactivity:

- Based on the user's choice:
 - **Option 1:** Displays all products with their details.
 - **Option 2:** Displays the contents of the shopping cart and the total cost.
 - **Option 3:** Allows the user to enter a product ID to add to the cart.
 - **Option 4:** Allows the user to enter a product ID to remove from the cart.
 - **Option 5:** Places an order and displays the order summary.
 - **Option 6:** Exits the program.

4. Order Placement:

- When the user chooses to place an order:
 - An `Order` object is created using the customer's shopping cart.
 - The order summary is displayed, including the order ID, customer name, products purchased, and total cost.

Running the Program

Compilation

to compile the code

1. Navigate to the project directory.
2. Compile the source code: `javac -cp src src/com/Main.java -d bin`
3. The compiled java byte codes will be in the bin folder.

to run the code

1. Navigate to the project directory.
2. Run the code: `java -cp bin com.Main`

The Screenshots

```
└─┐» java -cp bin com.Main
```

Welcome to the Online Store!

Enter your name: AMM

MENU:

1. Browse Products
2. View Shopping Cart
3. Add Product to Cart
4. Remove Product from Cart
5. Place Order
6. Exit

Choose an option: 1

Available Products:

Product [ID=P001, Name=Laptop, Price=\$899.99]

Product [ID=P002, Name=Smartphone, Price=\$499.99]

Product [ID=P003, Name=Headphones, Price=\$59.99]

Product [ID=P004, Name=Smartwatch, Price=\$199.99]

MENU:

1. Browse Products
2. View Shopping Cart
3. Add Product to Cart
4. Remove Product from Cart
5. Place Order
6. Exit

Choose an option: 3

Enter the Product ID to add to your cart:

P001

Laptop added to cart.

MENU:

1. Browse Products
2. View Shopping Cart
3. Add Product to Cart
4. Remove Product from Cart
5. Place Order
6. Exit

Choose an option: 3

Enter the Product ID to add to your cart:

P002

Smartphone added to cart.