



CSC212 E-COMMERCE INVENTORY & ORDER MANAGEMENT SYSTEM

Nader Alzahrani 44510196



PHASE (1)
UML DIAGRAM IS ATTACHED IN .ZIP FOLDER

Group Members & Tasks

Name	Id	tasks
نادر احمد الزهراني	445170196	UML, Orders, Review, Main
سلمان ياد الجرдан	445102518	UML, CSV Reading, Main
سعود عبدالرحمن آل زيد	445101786	Products, Customers, Exceptions

For this project, we set to use GitHub for the first time for most of us, mainly because it's a skill we knew we needed to learn, and because it made working together so much easier.

Overview

We didn't know how much Customers/Products/Orders/Reviews, is it what is only in the dataset file? Or what the user added? So, for simplicity, we will assume:

- **C = number of customers**
- **P = number of products**
- **O = number of orders**
- **R = number of reviews across all products (or per-product where noted)**
- **K = number of products inside a single order**

So, we can compute the space & time complexity

Class: ECommerceService (Main)

Fields

- `private LinkedList<Customers> allCustomers;`
- `private LinkedList<Products> allProducts;`
- `private LinkedList<Orders> allOrders;`

Methods

- `public static void main(String[] args)`
- `public void loadData()`
- `private void loadCustomers(String filePath) throws FileNotFoundException`
- `private void loadProducts(String filePath) throws FileNotFoundException`
- `private void loadOrders(String filePath) throws FileNotFoundException`
- `private void loadReviews(String filePath) throws FileNotFoundException`
- `private Customers findCustomerById(int customerId)`
- `private Products findProductById(int productId)`
- `public void runMainMenu()`
- `private void handleTop3Products()`
- `private void handleOrdersBetweenDates(Scanner scanner)`
- `private void handleCommonProducts(Scanner scanner)`
- `private void handleAddProduct(Scanner scanner)`
- `private void handleAddCustomer(Scanner scanner)`
- `private void handlePlaceOrder(Scanner scanner)`
- `private void handleAddReview(Scanner scanner)`
- `private void handleViewCustomerReviews(Scanner scanner)`
- `public void displayTop3ProductsByRating()`
- `public void displayOrdersBetweenDates(LocalDate startDate, LocalDate endDate)`
- `public void displayCommonProducts(int custId1, int custId2)`

Complexity Analysis (key operations)

- **loadCustomers:** Time O(C), Space O(C).
- **loadProducts:** Time O(P), Space O(P).
- **loadOrders:** Time O(O + O·K), Space O(O + O·K).
- **loadReviews:** Time O(R), Space O(R).
- **findCustomerById:** Time O(C), Space O(1).
- **findProductById:** Time O(P), Space O(1).
- **displayTop3ProductsByRating:** Time O(P·R(Average)), Space O(1).
- **displayOrdersBetweenDates:** Time O(O), Space O(1).
- **displayCommonProducts:** Time O(P·R(Average)), Space O(1).

Class: Customers

Fields

- `private int customerId;`
- `private String name;`
- `private String email;`
- `private LinkedList<Orders> orders;`

Methods

- `public boolean addOrder(Orders order)` — append order.
- `public String getOrderHistory()` — iterate orders and build text.
- `public int getOrderCount()`
- `public Orders getOrderById(int orderId)` — linear search.
- `public LinkedList<Orders> getOrders()`
- `public int getCustomerId()`
- `public String getName()`
- `public String getEmail()`

Complexity

- `addOrder`: Time O(1), Space O(1).
- `getOrderHistory`: Time O(n), Space O(L).
- `getOrderById`: Time O(n), Space O(1).
- `Getters`: Time O(1), Space O(1).

Class: Products

Fields

- `private int productId;`
- `private String name;`
- `private double price;`
- `private int stock;`
- `private LinkedList<Reviews> reviews;`

Methods

- `public void addReview(int reviewId, int customerId, int rating, String comment)`
- `public double getAverageRating()`
- `public void updateProductsStock(int quantity)`
- `public boolean isOutOfStock()`
- `public void updateProduct(String name, double price, int stock)`
- `public int getProductId()`
- `public String getName()`
- `public double getPrice()`
- `public int getStock()`
- `public LinkedList<Reviews> getReviews()`
- `public void setProductId(int productId)`
- `public void setName(String name)`
- `public void setPrice(double price)`
- `public void setStock(int stock)`
- `public void setReviews(LinkedList<Reviews> reviews)`
- `public String toString()`

Complexity

- `addReview`: Time O(1), Space O(1).
- `getAverageRating`: Time O(n), Space O(1).
- `updateProductsStock / isOutOfStock / updateProduct / getters/setters / toString`: Time O(1), Space O(1).

Class: Orders

Fields

- `private int orderId;`
- `private int customerId;`
- `private LinkedList<Products> products;`
- `private int productCount;`
- `private double totalPrice;`
- `private OrderStatus status;`
- `private LocalDate orderDate;`

Methods

- `public void addProduct(Products product)`
- `public void cancelOrder()`
- `public void updateStatus(OrderStatus newStatus)`
- `public OrderStatus getStatus()`
- `public int getOrderId()`
- `public int getCustomerId()`
- `public double getTotalPrice()`
- `public LocalDate getOrderDate()`
- `public int getProductCount()`
- `public Products[] getProducts()`
- `public String toString()`

Complexity

- `addProduct: Time O(1), Space O(1).`
- `getProducts: Time O(K), Space O(K).`
- `toString: Time O(K), Space O(L).`
- `cancelOrder / updateStatus / getters: Time O(1), Space O(1).`

Class: Reviews

Fields

- **private int reviewId;**
- **private int productId;**
- **private int customerId;**
- **private int rating;**
- **private String comment;**

Methods

- **public int getRating()**
- **public int getReviewId()**
- **public int getProductId()**
- **public int getCustomerId()**
- **public String getComment()**
- **public String toString()**

Complexity

- **Getters / `toString`: Time O(1), Space O(1).**