LAB211 Assignment

Type:	**Long Assignment**
Code:	**J1.L.P0028**
LOC:	**500**
Slot(s):	**N/A**

Title

Traditional Feast Order Management.

Background

- A company offering traditional feast services for special events (*such as weddings, anniversaries, and traditional ceremonies*) requires a management program. Key features include registering new customers, managing feast menu options, handling orders with integrated payment processing, updating order details, canceling orders, and saving data into a binary file (*feast_order_service.dat*).
- Students are required to analyze and design the program using an object-oriented programming (OOP) approach. Features such as abstraction, polymorphism, encapsulation, and inheritance must be applied during the development process.

Program Specifications

Build a management system for traditional feast orders with the following functionalities:

1. Register customers.
2. Update customer information.
3. Search for customer information by name.
4. Display feast menus.
5. Place a feast order.
6. Update order information.
7. Save data to file.
8. Display Customer or Order lists.

Others- Quit.

Features:

This system contains the following functions:

- **Function 1: Register customers - 75 LOC**

  Description:
  This function allows the creation of a new customer by collecting necessary details, validating inputs, …

  Input Requirements:
  The function requires the following customer details:

  - **Customer code**: A unique 5-character string. The first character is “C”, “G”or “K”, followed by 4 digits.
  - **Name**: A non-empty string between 2 and 25 characters long.
  - **Phone Number**: A 10-digit number belonging to a network operator in Vietnam.
  - **Email**: A valid email address in standard format.

Validation Rules:

1. **Customer code:**
   - Must start with one of characters in the set (C, G, K).
   - The remaining four characters must be numeric.
   - Must be unique.
2. **Name:**
   - Cannot be empty.
   - Length must be between 2 and 25 characters.
3. **Phone Number:**
   - Must contain exactly 10 digits.
   - Must belong to a valid Vietnamese network operator.
4. **Email:**
   - Must follow standard email formatting (e.g., example@domain.com).

Operation Workflow:

1. Prompt the user to input customer details.
2. Validate each input based on the rules above.
3. Save the registration record if all inputs are valid.
4. Prompt the user to either continue entering new customers or return to the main menu

- **Function 2: Update customer information – 75 LOC**

Description:
This function allows users to update the customer information, which can be updated includes.

- Name
- Phone number
- Email

If the customer code entered does not exist in the profile, the program will notify the user

Input Requirements:

1. **Customer Code**: The unique identifier of the customer (5 characters, as per the validation rules).
2. Fields for update:
   - **Name**: Must adhere to the rules of being non-empty and between 2 and 25 characters long.
   - **Phone Number**: Must contain exactly 10 digits and belong to a Vietnamese network operator.
   - **Email**: Must follow standard email formatting.

Validation Rules:

- Ensure that the customer code exists in the profile database before allowing updates.
- Apply the respective validation rules for each field being updated.

- Keep old information, if not enter new data.

Operation Workflow:

1. Prompt the user to enter the **Customer Code**.
2. Check if the customer exists in the profile:
   - If the customer exists:
     a. Prompt the user to update the desired fields (Name, Phone, Email).
     b. Validate the inputs based on the specified rules.
     c. Save the updated information.
     d. Display a success message.
   - If the customer does not exist:
     - Display the message: **"This customer does not exist ."**
3. Ask the user whether to continue with another update or return to the main menu

- **Function 3: Search for customer information by name – 50 LOC**

Description:

This function allows users to search for customers by their name. The user can enter either the full name or a partial name, and the system will display all matching customer names.

Operation Workflow:

1. **Input the Name**:
   - Prompt the user to enter the name or partial name of the customers they want to search for.
2. **Search Logic:**
   - Perform a search across the customer records to find names that match the input.
   - If multiple customers have the same name, they should all be displayed in alphabetical order.
3. **Handle Cases:**
   - **If matching customers are found**:
     - Display the list of matching customers with details : customer code, name, phone, email.
   - **If no matches are found**:
     - Display the message: **"No one matches the search criteria!"**.
4. **Return to Main Menu:**
   - After displaying the results or the message, prompt the user to return to the main menu.

Sample Output:

Case 1: Matching Customers Found
Matching Customers: An

```
-----------------------------------------------------------------
Code    | Customer Name         | Phone      | Email
-----------------------------------------------------------------
C0102   | An, Hoang Thi To      | 0938232345 | anhtt@gmail.com
K0197   | Anh, Nguyen Do Hoang  | 0909113355 | anhndh@outlook.com
C0901   | Lan, Vo Hoang Minh    | 0389121221 | lanvhm@lovemail.com
-----------------------------------------------------------------
```

Case 2: No Matches Found
No one matches the search criteria!

- **Function 4: Display feast menus – 50 LOC**

Description:

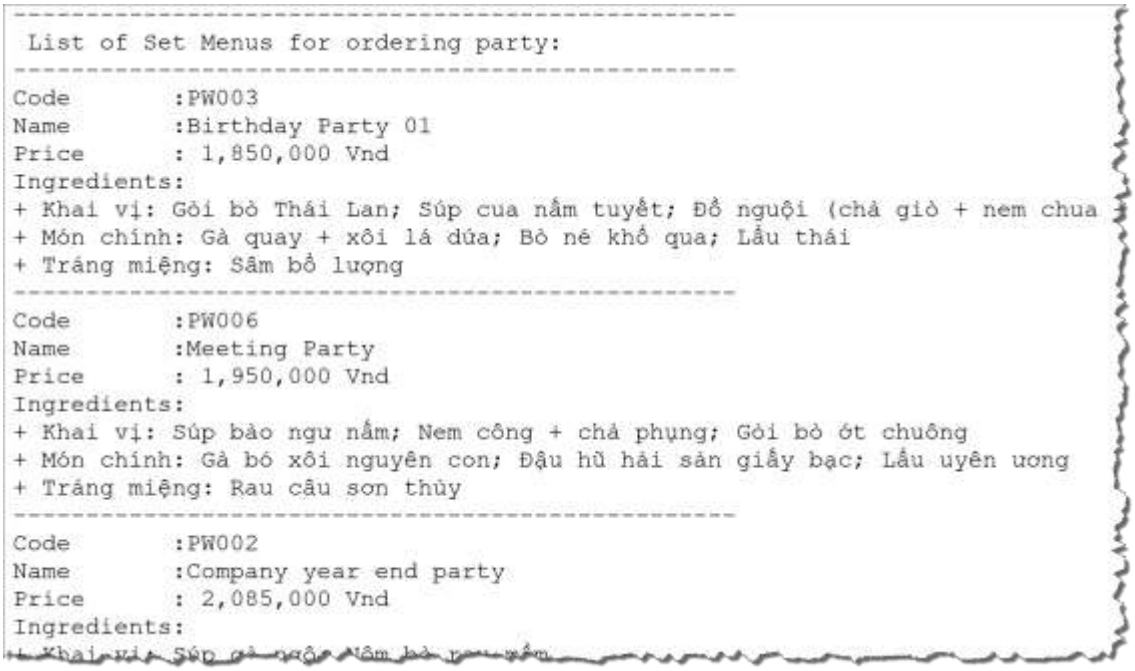
This function allows users to display a list of set menus that can be selected for party booking.

Operation Workflow:

1. **Check the existing "feastMenu.csv" file**:
   - This file contains information about the set menus that users can choose to order for a party. The file 'feastMenu.csv' is provided with the assignment as a requirement for use in customer ordering within the program.
2. **Handle Cases:**
   - **If the "feastMenu.csv" is existed**:
     - Display the menu list and sort it in ascending order based on the price attribute.
   - **If it does not exist**:
     - Display the message: **"Cannot read data from feastMenu.csv. Please check it.**".
3. **Return to Main Menu:**
   - After displaying the results or the message, return to the main menu.

Sample Output:

Case 1: the "feastMenu.csv" is existed



Case 2: the "feastMenu.csv" does not exist
Cannot read data from "feastMenu.csv". Please check it.

- **Function 5: Place a feast order – 100 LOC**

  Description:

  This function allows customers to book a party by selecting the Set Menu, the date they want to hold the event, the number of tables (*Supposing each set menu corresponds to 1 table, for 10 attendees*), ...

  Input Requirements:

  1. **Customer Code**: Customer code is only valid if it is in the list of customers who have registered information.
  2. **Code of SetMenu**: Code of Set menu to be order.
  3. **Number of tables**: integer number
  4. **Preferred event date**: The date on which the customer wants to hold the event

  Validation Rules:

  1. **Customer code:**
     - Must comply with the customer code regulations described earlier (*Start with C,K,G, ...*).
     - Customer code is only valid if it is in the list of customers who have registered information.
     - Cannot be left blank.
  2. **Code of SetMenu:** Each set menu provided by the file 'feastMenu.csv' has a unique identification code in the list. The code must meet the following requirements:
     - The set menu code used when a customer orders a party is only valid if it is in the provided list
     - Cannot be left blank.
  3. **Number of tables**: must be greater than zero
  4. **Preferred event date**: The preferred event date must be in the future.

  Operation Workflow:

  1. **Check order data for duplicates**:
     - A customer can order multiple times on different days. An order information is considered unique based on the simultaneous combination of 3 attributes: Customer code, Set Menu code and event date.
  2. **Handle Cases:**
     - **If the order information already exists**:
       - Display the message: **"Dupplicate data !"**
     - **If not duplicate**:
       - Automatically generate a unique code for the corresponding order
       - Save information to the list of successfully placed orders
       - Calculate the total cost of the order based on selected Set Menu and number of tables
       - Display order information that customers have successfully placed.

3. **Return to Main Menu:**
    - After displaying the message, prompt the user to return to the main menu or continue with place another order.

Sample Output:

Case 1: Order successful

```
---------------------------------------------------------------
Customer order information [Order ID: 12]
---------------------------------------------------------------
 Code          : K0310
 Customer name : Yen, Hoang Minh
 Phone number  : 0351232321
 Email         : yenhm11@gmail.com
---------------------------------------------------------------
 Code of Set Menu: PW002
 Set menu name   : Company year end party
 Event date      : 14/02/2025
 Number of tables: 8
 Price           : 2,085,000 Vnd
 Ingredients:
 + Khai vị: Súp gà ngô; Nộm bò rau mầm
 + Món chính: Tôm hấp bia; Bò sốt tiêu đen + bánh mì; …
 + Tráng miệng: Rau câu dừa
---------------------------------------------------------------
 Total cost      : 16,680,000 Vnd
---------------------------------------------------------------
```

- **Function 6: Update order information– 50 LOC**

Description:
This function allows users to update information about related orders in the system. Information that can be updated includes.

- Code of set menu
- Number of tables
- Preferred event date

Input Requirements:

1. **Order ID**: The unique identifier of the order.
2. Fields for update:
    - **Code of set me**nu: Must comply with the rules of not being blank, being 5 characters long and matching the Set menu code in the list of Set Menus provided by the feastMenu.csv file.
    - **Number of tables**: Must be an integer greater than zero.
    - **Preferred event date**: Must be a valid date in the future.

Validation Rules:

- Ensure that the Order ID exists in the profile database before allowing updates.
- Apply the respective validation rules for each field being updated.
- Keep old information, if not enter new data.

- Do not allow updating an order whose event date occurred before the current date

Operation Workflow:

1. Prompt the user to enter the **Order ID**.
2. Check if the Order exists in the profile:
   - If the order exists:
     a. Prompt the user to update the desired fields (Code of set menu, Number of tables, Preferred event date).
     b. Validate the inputs based on the specified rules.
     c. Update price based on set menu code
     d. Save the updated information.
     e. Display a success message.
   - If the Order does not exist:
     - Display the message: **"This Order does not exist."**
3. Ask the user whether to continue with another update or return to the main menu

- **Function 7: Save data to file – 50 LOC**

  Description:
  This function will allow saving customer list data in the system or successfully registered orders, into the files **customers.dat** or **feast_order_service.dat**.

  Operation Workflow:

  1. **Data Collection**:
     - Gather all current data from the program, including customer details (code, name, phone, email) or Order details (ID, Custmer code, Set menu code, Price, Number of table, …).
  2. **Serialization**:
     - Convert the data into an object format suitable for file storage as a binary object file.
  3. **Save to File**:
     - Write the serialized data to a file. The file should be named appropriately (e.g., **customers.dat or feast_order_service.dat**).
  4. **Confirmation Message**:
     - Display a confirmation message once the data is successfully saved.
  5. **Return to Main Menu**:
     - After saving the data, return to the main menu.

  Sample Confirmation Message:

  - Customer data has been successfully saved to **"customers.dat".**
  - Order data has been successfully saved to **"feast_order_service.dat".**

- **Function 8: Display Customer or Order lists – 50 LOC**

  Description:
  This function will allow to display the list of customers in the system, or successfully registered orders,

which have been saved in the files **customers.dat** or **feast_order_service.dat**. If no registration data is available, the program should notify the user.

Operation Workflow:

1. **View to interact**
   - Provide a menu-based interface for users to choose from
2. **Check the Customers or Order List**:
   - Retrieve the current list of registered customers or orders from the system.
3. **Display Data:**
   - If the list contains entries:
     - With customer data, Display the following details for each customer in a formatted table or list and sort by customer name in alphabetical order:
       - Customer code
       - Name
       - Phone Number
       - Email
     - With order data, Display detailed information of orders in a formatted table or list and sort by event date in ascending order.:
       - Order ID
       - Event date
       - Customer ID
       - Set Menu
       - Price
       - Number of tables
       - Total cost
   - If the registration list is empty:
     - Display the message: **"Does not have any customer information."**
4. **Return to Main Menu:**
   - After handling, return to the main menu.

Sample Output:

Case 1: List Containing Customers Data

Customers information:

```
-------------------------------------------------------------
Code    | Customer Name         | Phone      | Email
-------------------------------------------------------------
C0012   | An, Hoang Thi To      | 0987654321 | anhtt@hotmail.com
G0171   | Binh, Ngo Quoc        | 0902345678 | binhnq@yahoo.com
K0310   | Yen, Hoang Minh       | 0351232321 | yenhm11@gmail.com
-------------------------------------------------------------
```

Case 2: List Containing Orders Data

```
-------------------------------------------------------------------------
ID    | Event date |Customer ID| Set Menu| Price    | Tables |       Cost
-------------------------------------------------------------------------
K0310 | 14/02/2025 | K0310     | PW005   | 2,250,000|       3|  6,750,000
G0171 | 06/03/2025 | G0171     | PW006   | 1,950,000|      20| 39,000,000
C0012 | 08/03/2025 | C0012     | PW003   | 1,850,000|       5|  9,250,000
-------------------------------------------------------------------------
```

Case 3: List is Empty

No data in the system.

The above specifications **provide basic information**. You are required to ***conduct a detailed requirements analysis and build the application based on the real-world requirements***.

The lecturer will explain the **full set of requirements only once during the initial slot of the assignment**.