Title

Product Management – Read and Write File.

Background

A store needs a Product information management program. With basic requirements such as creating a Product, displaying Product information, and updating information, ... Product information is stored in a text or binary file (Product.dat).

Program Specifications

Build a Product management program. With the following basic functions:

1. Create a Product
2. Check exits Product.
3. Search Product' information by Name
4. Update Product:
   4.1. Update Product.
   4.2. Delete Product.
5. Save Products to file.
6. Print list Products from the file.
Others- Quit.

Features:

This system contains the following functions:

- **Function 1: Create a Product - 50 LOC**
  - Require to input a piece of Product information including ProductID, ProductName, UnitPrice, Quantity, Status

  - Check the valid data with the following conditions:

    - Product Name must be at least five characters and no spaces.
    - UnitPrice is a real number and ranges from 0 to 10000
    - Quantity is an integer number and ranges from 0 to 1000
    - Status is a string and has values: Available or Not Available

  - Create new a Product.

  - Ask to go back to the main menu.

- **Function 2: Check to exist Product– 50 LOC**
  - The system will check the product ID that is stored in the file Product.dat.

  - A message "Exist Product" should be displayed in the case the product ID exists in the Product.dat file.

  - Otherwise, the message "No Product Found!" will display.

  - Ask to go back to the main menu.

- **Function 3: Search Product information by name – 50 LOC**
  - Require to enter a search string (a part of product name), and return a list of Product information containing the search string.
  - If the list Product is null, the notification "Have no any Product", else print the list Product information order by the Product Name.
  - Ask to go back to the main menu.

- **Function 4: Update Product**
  - **Function 4.1: Update Product information- 50 LOC**
    - The user enters the ProductID from the keyboard
    - If it does not exist, the notification "Product does not exist". Otherwise, the Product can edit the remaining information. If Information is blank, then do not change old information
    - Show the result of the update: success or failure.
    - Ask to go back to the main menu.

  - **Function 4.2: Delete Product information – 50 LOC**
    - The user enters the ProductID from the keyboard
    - If it does not exist, the notification "Product does not exist". Otherwise, the Product can be deleted
    - Must show confirm message
    - Show the result of the delete: success or fail
    - Ask to go back to the main menu.

- **Function 5: Save to file - 50 LOC**
  - Write a list of the Product's information to the file (Product.dat).
  - Ask to go back to the main menu.

- **Function 6: Print all lists from file – 50 LOC**
  - Loading list Product information from the file into Collection
  - Displaying list Product information order by Quantity descending. If the same Quantity then order by UnitPrice ascending
  - Ask to go back to the main menu.

- **Function 7: Create a layout – 50 LOC**
  - The program is organized in the form of a function menu.
  - The support function asks if the user wants to continue or not.

- **Bonus 50 LOC (maximum 500 LOC) if the student applies one of the Design Patterns (such as DAO pattern, Factory pattern, Repository pattern, and so on) in this project. More references for the design pattern:** https://www.tutorialspoint.com/design_pattern/index.htm
- The above specifications are only basic information; you must perform a requirements analysis step and build the application according to real requirements.
- The lecturer will explain the requirement only once on the first slot of the assignment.