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 the Product information, and updating information, ... The 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. Print
2. Create a Product
3. Check exits Product.
4. Search Product' information by Name
5. Update Product:
    5.1. Update Product.
    5.2. Delete Product.
6. Save Products to file.
7. Print list Products from the file.

Others- Quit.

Features:

This system contains the following functions:

- **Function 1: Create a Product - 50 LOC**
    o The user is required to input a piece of **Product information** including ProductID, ProductName, UnitPrice, Quantity, Status

    o The application should check the valid data with the following conditions:

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

    o After that, the system creates new a Product.

    o The application asks user to go back to the main menu.

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

    o A message "Exist Product" should be displayed in the case the ProductName exists in the Product.dat file.

    o Otherwise, the message "No Product Found!" will be displayed.

o   The application asks user to go back to the main menu.

- **Function 3: Search Product information by name – 50 LOC**
  o   The user is required to enter a search string (a part of product name). Then, the application should return a list of Product information containing the search string.

  o   If the list Product is null, the notification "Have no any Product" will be shown. Otherwise, the application will print the list Product information that is ordered by the Product Name.

  o   The application asks user to go back to the main menu.

- **Function 4: Update Product**
  - **Function 4.1: Update Product information- 50 LOC**
  o   The user enters the ProductID from the keyboard.

  o   If it does not exist, the notification "Productname does not exist" will be shown. Otherwise, the Product can be edited the remaining information. If the inputted information is blank, old information will not be changed.

  o   The result of the update should be shown as success or failure status.

  o   The application asks user to go back to the main menu.

  - **Function 4.2: Delete Product information – 50 LOC**
  o   The user enters the ProductID from the keyboard

  o   If it does not exist, the notification "Productname does not exist" will be shown. Otherwise, the Product can be deleted

  o   The result of the delete action should be shown as success or fail status.

  o   The application asks user to go back to the main menu.

- **Function 5: Save to file - 50 LOC**
  o   The application allows the user write a list of the Product's information to the Product.dat file.

  o   The application asks user to go back to the main menu.

- **Function 6: Print all lists from file – 50 LOC**
  o   The application can allow to load Product information list from the file into the Collection.

  o   The application shows the list of Product information order by Quantity descending. If the products have same quantity, the list will be sorted with the ascending UnitPrice field.

  o   The application asks user to go back to the main menu.

- **Function 7: Create a layout – 50 LOC**
  o   The program is organized in the form of a function menu.

  o   The support function will be asked 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.