LAB 211 Assignment

Type:	Long Assignment
Code:	J1.L.P0020
LOC:	400+
Slot(s):	N/A

Title

The Vehicle Management - Read and Write File.

Background

- Your family wants to develop a program to manage the vehicles at your showroom. Vehicles have categorizes and properties. Every vehicles has properties as: id, name, color, price, brand, type, productYear..

- This management program needs to have primary functions such as: displaying Vehicles information, searchVehicle, addVehicle, editVehicle…

- Vehicle information is stored in a text or binary file (vehicle.dat).

- Let's build your idea based OOP model.

Program Specifications

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

0. Adding new vehicle.
1. Checking exits Vehicle.
2. Updating vehicle.
3. Deleting vehicle.
4. Searching vehicle.
   4.1 Searching by id.
   4.2 Searching by name.
5. Displaying vehicle list.
   5.1 Displaying all.
   5.2 Displaying all (descending by price)
6. Saving Vehicle to file.
7. Printing list Vehicles the file.

Others Quit

Features:

This system contains the following functions:

Display a menu and ask users to select an option.

- **Function 0: Add new vehicle - 50 LOC**
  - Require to input vehicle information including: ID_Vehicle, Name_Vehicle, color_Vehicle, price_Vehicle, brand_Vehicle, type,productYear.
  - Remember to the constraints must be checked.
  - Add the new vehicle and show the result of the add: success or fail.
  - Ask to continuous create new vehicle or go back to the main menu.

- **Function 1: Check to exist Vehicle– 50 LOC**
  - The system will check the ID_Vehicle that is stored in the file Vehicle.dat.
  - A message “Exist Vehicle” should be displayed in the case the ID_Vehicle exists in the Vehicle.dat file.
  - Otherwise, the message “No Vehicle Found!” will display.
  - Ask to go back to the main menu.

- **Function 2: Update vehicle – 50 LOC**
  - Require enter the vehicle’s id_Vehicle.
  - If vehicle does not exist, the notification “Vehicle does not exist”. Otherwise, user can start input new information of vehicle and update.
  - If new information is blank, then not change old information.
  - Then system must print out the result of the updating.
  - After updating, the program returns to the main screen.

- **Function 3: Delete vehicle – 50 LOC**
  - User can delete any vehicle in the showroom by id_vehicle.
  - Before the delete system must show confirm message.
  - Show the result of the delete: success or fail.
  - After delete, the program returns to the main screen
- **Function 4: Search vehicle**
  - Create a submenu that allows the user to select way to search: search by name_vehicle or by id_vehicle.

✓ **S.4.1: Search by Name_vehicle – 50 LOC**
  - User input the text want to search.
  - The system will search in the show room, and return all vehicle that has name contain the search string.
  - Show result list: all information of vehicle(descending).

✓ **S.4.2: Search by id_vehicle – 50 LOC**
  - The user enters the vehicle id_vehicle.
  - The system searches the show room, and returns the vehicle that has id_vehicle same with the search string.
  - Show result: all information of vehicle.
- **Function 5: Display vehicle list**
  - Create a submenu that allows the user to select way to show: show all or by type.

✓ **D.5.1: Show all – 50 LOC**
  - The system will show this list of vehicle in the show room.
  - Show result list: all information of vehicle.

✓ **D.5.2: Show all (descending by price_vehicle)– 50 LOC**
  - The system will show this list of vehicle in the show room.
  - Show result list: all information of vehicle (descending by price_vehicle).
- **Function 6: Save data to file-50LOC**
  - Write a list of the Vehicle’s information to the file (Vehicle.dat).
- **Function 7: Print vehicle list**

Create a submenu that allows the user to select way to show: show all or by type.

✓ **P.7.1: Print all – 50 LOC**
  - The system will show this list of vehicle in the show room.

✓ **P.7.2: Print all (descending by price_vehicle)– 50 LOC**

  - The system will show this list of vehicle in the show room.
  - Show result list: all information of vehicle (descending by price_vehicle)

- 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