LAB211 Assignment	Type:	Long Assignment
	Code:	J1.L.P0021
	LOC:	400+
	Slot(s):	N/A

Title

Event Management – Read and Write File dat(binary file)

Background

A local event organizing company requires a program to manage their events efficiently. The program should include functionalities such as creating new events, checking existing events, searching event information, updating events, saving events to a file, and printing the list of events from the file. All event information will be **stored in a binary file (events.dat).**

Students are required to analyze and design the program using the object-oriented programming model. It is expected that features such as abstraction, polymorphism, encapsulation, and inheritance are employed during the analysis and design process. This requirement is mandatory and must be fulfilled by all students.

The ArrayList data structure must be used to represent the list.

Program Specifications

Develop an event management program with the following basic functionalities:

1. Create a new event.
2. Check if an event exists.
3. Search for event information by location.
4. Update event:

   4.1. Update event details.

   4.2. Delete event.
5. Save events to a file.
6. Print the list of events from the file.
7. Others - Quit.

Features:

This system contains the following functions:

- **Function 1: Create an Event - 50 LOC**
  - **Input Event Information:** Prompt the user to input information about the new event, including the name, date, location, number of attendees, and status. Automatically generate unique event ID for the new event to avoid duplicates.
  - **Check for valid data:**
    - Ensure that the event name is at least five characters long and does not contain spaces.
    - Ensure that the event date is valid and in the correct format (YYYY-MM-DD).

- Ensure that the location is provided.
- Ensure that the number of attendees must be greater than 0. (the number of people who are present or participate in event)
- Ensure that the status is either "Available" or "Not Available".

- **Create New Event:**
  - If all information is valid, create a new event with the details provided.
- **Ask to Continue:**
  - Ask the user if they want to create another event or go back to the main menu.

- **Function 2: Check the Existence of Event - 50 LOC**
  - **Check the Event Existence:**
    - Read the event ID from the user input.
    - Search for the event ID in the event database file (e.g., events.dat).
  - **Display Message:**
    - If the event ID exists in the file, the message "Exist Event" will be displayed.
    - Otherwise, display the message "No Event Found!".
  - **Ask to Continue:**
    - Ask the user if they want to perform another check or go back to the main menu.

- **Function 3: Search Event Information by Location- 50 LOC**
  - **Input Search String:**
    - Prompt the user to enter a search string (part of the event location).
  - **Search Events:**
    - Read the search string from the user input.
    - Search for events in the event database file (e.g., events.dat) that contain the search string in their location.
  - **Display Search Results:**
    - If there are events matching the search criteria:

      Print the list of event information in order of event attendance.
    - Otherwise, if the list of events is empty:

      Print the notification "No Event Found".
  - **Ask to Continue:**
    - Ask the user if they want to perform another search or go back to the main menu.

- **Function 4: Update and Delete Event**

- **Function 4.1: Update Event Information - 50 LOC**

  1. **Input Event ID:**
     - Prompt the user to enter the event ID they want to update.
  2. **Check Event Existence:**
     - Search for the event ID in the event database file (e.g., events.dat).
     - If the event does not exist, display the notification "Event does not exist".
  3. **Update Event Information:**
     - If the event exists, allow the user to edit the remaining information.
     - **If any information is left blank, do not change the old information**.
  4. **Show Update Result:**
     - **Display the result of the update**: success or failure.
  5. **Ask to Continue:**
     - Ask the user if they want to go back to the main menu.

- **Function 4.2: Delete Event Information - 50 LOC**

  1. **Input Event ID:**
     - Prompt the user to enter the event ID they want to delete.
  2. **Check Event Existence:**
     - Search for the event ID in the event database file.
     - If the event does not exist, display the notification "Event does not exist".
  3. **Confirm Deletion:**
     - Ask the user to confirm the deletion of the event.
  4. **Delete Event:**
     - If confirmed, delete the event from the event database file.
  5. **Show Delete Result:**

- **Display the result of the deletion** success or failure.

6. **Ask to Continue:**

- Ask the user if they want to go back to the main menu.

- **Function 5: Save to File - 50 LOC**
  - **Save Events to File:**
    - Write the list of events' information to the event database file (e.g., events.dat).
  - **Show Save Result:**
    - Display a message indicating that the events have been successfully saved to the file.
  - **Ask to Continue:**
    - Ask the user if they want to go back to the main menu.

- **Function 6: Print All Events from File - 50 LOC**
  - **Load Events from File:**
    - Read the event information from the event database file (e.g., events.dat) into a collection.
  - **Display Events Information:**
    - Display the list of event information sorted by event date in ascending order.
    - If two events have the same date, sort them by event name in ascending order.
  - **Ask to Continue:**
    - Ask the user if they want to go back to the main menu.

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

- **Bonus 50 LOC (maximum 500 LOC for total project points) 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 the 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.