LAB211 Assignment

Type:	**Long Assignment**
Code:	**J1.L.P0033**
LOC:	**500**
Slot(s):	**N/A**

Title

KHOE Group Sports Booking System.

Background

- KHOE Group is a sports complex offering recreational and fitness facilities to residents within a residential community, promoting health and wellness through physical activity. The organization seeks to develop a software system to manage reservations for amenities such as badminton courts, football fields, and swimming pools. The system will support ten core functions, as outlined in the technical specification table below.
- 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 Sports Booking System with the following functionalities:

1. **Import Facility from CSV file**
2. **Update Facility Information**
3. **View Facilities & Services**
4. **Book a Facility / Service**
5. **View Today's Bookings**
6. **Cancel a Booking**
7. **Monthly Revenue Report**
8. **Service Usage Statistics**
9. **Save All Data**

Others- Quit.

Features:

Software Requirements Specification (**SRS**): **KHOE Group Sports Booking System**

This system contains the following functions:

- **Function 1: Import Facility from CSV file - 50 LOC**

  Description:
  The current module is provided with a list of sports facilities exported by another module; therefore, this function is responsible for reading data from a plain text file named "***facility_schedule.csv***".

  Input Requirements:

  - File path to the *"Active_Room_List.txt"* file.
  - **File format**: .txt, each line represents one room.
  - **Line format**:
    `Id, Facility Name, Facility Type, Location, Capacity, Availability Start, Availability End`

- **Example**:
  `BC-01, Badminton Court 1, Badminton, Zone A - 2nd Floor, 4, 2025-10-10 08:00, 2025-10-10 09:00`

Validation Rules:

1. File must exist
2. All six fields must be provided per line.
3. Facility Name must be unique.
4. Capacity must be a positive integer

Operation Workflow:

1. Check file existence.
2. File is read line-by-line.
3. Each line is validated and parsed.
4. Valid entries are saved; invalid ones are logged

Sample Output

10 rooms successfully loaded.
2 entries failed.

- **Function 2: Update Facility Information – 50 LOC**

Description:

Only allow updates to the following information: ***Location***, ***Capacity***, and ***Availability Schedule***.

Input Requirements

- Id (*or Facility Name*)
- Updated fields (e.g., *Location, Capacity, Availability schedule*)

Validation Rules

- Id of Facility Name must exist in the system.
- Updated data must follow original format validations.

Operation Workflow:

1. User selects "***Update facility Information***":
2. Retrieves and displays current info
3. **Search guest by Facility Name :**
   - **If the "Facility Name" or Id is existed**:
     - Retrieves and displays current info.
     - User submits changes
     - System validates and saves updates if valid
     - Display success message
   - **If it does not exist**:
     - Display the message: **"No Facility or Service found in database !"**.
4. **Return to Main Menu:**
   - After displaying the results or the message, return to the main menu.

Sample Output:

Case 1: the "Facility Name" is existed

Facility information updated for Name: '**Tennis Court 2**'

Case 2: the "Facility Name" does not exist

No Facility or Service found in database !.

- **Function 3: View Facilities & Services – 50 LOC**

Description:
Display all available data related to training facilities or services in a tabular format. If no data is available, display an appropriate notification message..

Input Requirements:

- No external input required.

Validation Rules:

- Facilities & Services list must not be empty.

Operation Workflow:

2. User selects "**View Facilities & Services** ".
3. System retrieves all Facilities & Services records
   - If the data exists:
     - Data is formatted and presented to the user
   - If the data does not exist:
     - Display the message: **" Facilities & Services list is currently empty, not loaded yet."**
4. Data is formatted and presented to the user

Sample Output

Facilities & Services `List`

```
Facility Name      | Type         |Location           | Capacity |
Availability
-------------------+--------------+-------------------+----------+----------
--
Tennis Court 1     | Tennis       | Zone A - 2nd Floor|        4 |
11/10/2025
                                                                    12:00-
13:00
Mini Football Field| Football     | Zone C - Ground   |       10 |
21/10/2025
                                                                    09:00-
14:00
Badminton Court 4  | Badminton    | Zone D - 1st Floor|        6 |
12/10/2025
                                                                    06:00-
17:00
......
```

- **Function 4: Book a Facility / Service – 50 LOC**

  Description:

  Allow users to book their desired facility or service based on predefined time slots.

  Input Requirements:

  - Player name
  - Facility name
  - Date & Time (reserved)
  - Duration (hours)

  Validation Rules:

  - Player name: Must be a string between 2 and 18 characters.
  - Facility name: Must correspond to a facility or service defined within the system.
  - Date & time: Must match an available time slot for the corresponding facility
  - Duration: Must be an integer in the range of 1 to 5

  Operation Workflow:

  1. User selects "**Book a Facility / Service**"
  2. System prompts for input requirements
  3. Generate a unique code for the corresponding booking; this code must uniquely identify the booking in the system
  4. Validate data and save to file
     (*You can use the file in either text or object format, depending on your design*)
  5. Return to Main Menu:
     - After displaying the message, prompt the user to return to the main menu or continue with place another order.

- **Function 5: View Today's Bookings – 50 LOC**

  Description:
  Display a list of bookings (*courts and services*) for today or for a user-specified date.

  Input Requirements:

  - Date (*if left blank, defaults to today*).

  Validation Rules:

  - Valid date format (YYYY-MM-DD)

  Operation Workflow:

  1. User selects or uses today's date
  2. System fetches bookings for that date.
  3. Displays list sorted by time:

- If the booking list contains data, display it in a table as shown in the “Sample Output” section.
- If the booking list is empty:
  - Display the message: **" There are currently no courts or services booked !."**

4. Ask the user whether to continue or return to the main menu

Sample Output:

Case 1: If bookings for courts and services are available on 31/08/2025

```
------------------------------------------------------------------
Bookings on 31/08/2025
------------------------------------------------------------------
 Time | Facility           | User                | Duration
------------------------------------------------------------------
 07:00| Tennis Court 3     | James Anderson      |        5
 09:00| Badminton Court 1  | Daniel Wilson       |        2
 09:00| Table Tennis Area 2| Emma Anderson       |        2
 10:00| Mini Football Field| Christopher Taylor  |        1
 10:00| Table Tennis Area 1| James Anderson      |        3
------------------------------------------------------------------
```

Case 1: If bookings for courts and services are not available on 31/08/2025

```
There are currently no courts or services booked !.
```

- **Function 6: Cancel a Booking – 50 LOC**

Description:
Allows users to cancel an existing booking.

Input Requirements

- Booking ID (or **Player name + Date + Facility**)

Validation Rules

- Booking must exist.
- Cannot cancel if the time has already passed

Operation Workflow:

1. User selects booking to cancel
2. System verifies the booking
3. Deletes or marks as canceled
4. Return to Main Menu:
   - After saving the data, return to the main menu.

Sample Output

Case 1: If the ID exists and the availability start time is in the future

```
Booking inforamtion:
 + Booking ID    : 071521060920025
 + Player name   : Sophia Brown
 + Facility name : Badminton Court 1 [Id: BC-01]
 + Date          : 09/10/2025
 + Time          : 07:00
 + Duration      : 3
------------------------------------------------------------------
```

```
Do you really want to cancel this court booking? [Y/N]:Y
... System message ...
"The booking ID 071521060920025 has been successfully canceled."
```

Case 2: If the ID does not exist in the guest list

```
Booking for ID '071521060920025' could not be found
```

Case 3: If the ID exists in the booking list but the time is in the past

```
This booking (ID: 071521060920025) cannot be canceled
```

Function 7: Monthly Revenue Report – 50 LOC

Description:

Generates a revenue summary for a given month.

Input Requirements

- Target month (MM/YYYY)

Validation Rules

- Month must be valid and within system's data range.

Operation Workflow:

1. User selects "**Monthly Revenue Report**"
2. Inputs target month:
   - Verify that the month and year provided for the statistics are valid.
3. **System aggregates total revenue per facility and display data**
   - If the month and year provided for the statistics are valid:
     - The system calculates total revenue and displays the results with room attributes:
       - No.
       - Facility
       - Amount
   - If the list does not contain entries:
     - Display the message: **" No data available in the Monthly Revenue Report"**
4. **Return to Main Menu:**
   - After handling, return to the main menu.

Sample Output:

Case 1: Containing Data

Monthly Revenue Report - 'MM/YYYY'

```
-------------------------------------
No.  | Facility        | Amount
-------------------------------------
1    | Badminton       |    30000000
2    | Football Field  |    25000000
3    | Table tennis    |    70000000
4    | Swimming Pool   |    45000000
...
-------------------------------------
Total                   170000000
-------------------------------------
```

Case 2: does not contain entries

```
No data available in the Monthly Revenue Report
```

- **Function 8: Service Usage Statistics – 50 LOC**

Description:
Shows how many users used each type of facility/service

Input Requirements

- Enter a time range to generate statistics, or leave it blank to include all.

Validation Rules

- Valid date format if time range provided.

Operation Workflow

1. User selects time range (*optional*).
2. System counts unique users per service
3. Displays results

Sample Output

```
Revenue Report by Room Type
-----------------------------------
  Facility type  | No. of Players
-----------------------------------
  Badminton      |        150
  Football       |         65
  Table tennis   |         58
  Swimming       |        182
```

- **Function 9: Save All Data – 50 LOC**

Description:
This function will allow saving data in the system, into the files **BookingInfo.dat**.

Operation Workflow:

1. **Data Collection:**
   - Gather all current data from the program, including booking details (Booking ID, Player name, date, time, …).
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., **BookingInfo.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:

- Booking court has been successfully saved to **"BookingInfo.dat"** file.

- **Function 10: Quit – 50 LOC**

  Description:

  Allow users to quit the application. If booking data is unsaved when the user tries to exit, prompt the user to confirm saving before exiting

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.