CMPS 350 · Web Development Fundamentals
Assignment 02 · JavaScript

Instructions

1. The deadline for completing your work is Saturday, March 16, 2024 at 23:59:59. **No late submission will be accepted**.

2. As you make progress, push your code to your GitHub repository under assignments/assignment02

3. Any form of plagiarism will be treated seriously and reported. The use AI tools such as ChatGPT , Gemini etc .. is strictly forbidden.

4. **You are not allowed to use any explicit loop statement such as while, do, or for throughout this assignment**.

Event Management App   [80 points]

Create an **event-app** directory under Assignment 2. The app will be used to manage events and track participants and event details. The system is contains the following classes:

1. **Participant**: Represents the individuals participating in the event .This class includes the following attributes:
   - **participantId**: Unique identifier for each participant.
   - **firstName**: First name of the participant.
   - **lastName**: Last name of the participant.
   - **gender** : The gender of the participant
   - **email**: Email address of the participant.

2. **Event**: Represents an event that participants can register for. This class includes the following

   Attributes:
   - **eventId**: A unique identifier for each event.
   - **eventName**: The name of the event.
   - **date**: The date of the event.
   - **location**: The location where the event will take place.
   - **participants**: List of participants attending the event.

   Methods:
   - **addParticipant(participant):** Adds a participant to the event.
   - **removeParticipant(participantID):** Removes a participant from the event.
   - **getParticipants():** Returns a list of participants attending the event.
   - **updateParticipant(participant):** updates the information of the participant having the matching participantId.

3. The app should have **events-repo.js** to maintain the list of events and allow adding, updating, and deleting events. This module is expected to provide a set of functions for these operations and maintain a singular object that holds the event list.
   - **addEvent(event):** adds an event to the list of events. eventID should be auto-assigned a random number.
   - **updateEvent(event):** updates the event having the matching eventID.
   - **deleteEvent(eventID):** delete the event with eventID from the list of events.

- o **getEventsByParticipant(participantId):** returns the list of events registered by a particular participant.
- o **findUpcomingEvents(date)** : This method filters and returns a list of all events scheduled after a specific date.
- o **getEventParticipantsSummary(eventId)** : Generates a summary of participants for a specific event, including
  - Event Info
  - Total participant count
  - The number of female and male
4. Create an **app.js file** and test the app functionality using a few events and participants.

Unit Testing   [20 points]

Create a file **events-repo.spec.js** under **event-app/test** folder and write the necessary test cases for **at least 5 of the methods** of the event-repo.js module. You should use Mocha and Chai to write the unit tests.