Task: Virtual Event Registration & Score Tracker

Description

Build a mini web app for a virtual event or game competition where students can:

1. Register for the event by entering name, email, and age.

2. The system validates the input (age must be ≥12, email must not be empty).

3. After registration, the user can participate in a mini quiz/game:

   o The system generates 3 random questions with multiple-choice answers.

   o The user selects answers, and the system calculates the score using numbers, operators, and if-else.

4. The system generates a result summary:

   o Total marks, percentage, grade (A/B/C/D), and timestamp (Date()).

5. The user data and result are stored in JSON format and displayed.

6. Introduce a Promise to simulate server response delay:

   o After submitting answers, a "Calculating result…" message appears for 2–3 seconds before showing results.

7. Use try-catch to handle errors (empty inputs, invalid age).

Concepts Covered

- Basics, Variables, Data Types → storing name, email, age, answers, marks

- Strings → messages, JSON output

- Numbers & Math → random question selection, score calculation, percentage

- Operators & If-Else → checking answers, assigning grades, age eligibility

- Date() → result timestamp

- Errors → input validation and error handling

- JSON → store/display student info and result

- Promises → simulate asynchronous server delay

Instructions

1. Create an HTML page with sections for:

   o Registration Form → Name, Email, Age

- Quiz/Game Section → 3 questions with multiple-choice answers

- Result Section → Total score, percentage, grade, timestamp, JSON display

2. When the user clicks Register:

   - Validate inputs.

   - Show errors if invalid (age < 12, empty fields).

3. When the user submits answers:

   - Use a Promise with setTimeout (2–3 seconds) to simulate server processing.

   - Randomly assign correct answers and calculate score.

   - Assign grades using if-else.

   - Create a JSON object containing user info, score, grade, and timestamp.

   - Display the results and JSON.

4. Use short CSS to style the form, quiz, and result display.