Full Stack Book Club

Task: Using the provided API, create an Angular app that allows users to track book club members and how many books each member has read. Members are added by filling out a form. Members can be removed by clicking a button. And the books read count can be incremented for each user by clicking a button.

THE API:

Start up the existing API built with Spring Boot. ***NOTE: This API is already made. You only need to make an Angular app for it.*** Follow the setup instructions for Assessment 8. View the API at http://localhost:8080/club-members.

The API stores club member objects that look like this:

```
{
  "id": 8,
  "name": "Wanda",
  "booksRead": 3,
  "foundingMember": true
}
```

The API has the following endpoints:

- **GET /club-members** - Responds with an array of all club members.
- **GET /club-members/{id}** - Responds with a single object matching the given ID.
- **POST /club-members** - Adds a member to the database and generates an ID. The club member is specified as the JSON body of the request.
- **PUT /club-members/{id}** - Updates a member in the database. The club member is specified as the JSON body of the request.
- **DELETE /club-members/{id}** - Removes the specified member from the database.

The Angular App

Setup:
Follow the setup instructions for Assessment 8. Name the Angular project **bookclub-frontend**.

This will act as the front-end of your app and allow users to manage members in the book club API. You will need the following:
Book Club Member Model:
Create a model with properties for the following:

- id (mark as optional)
- name
- booksRead

- foundingMember

Book Club Service:
Create a service with the following methods:

- **getAllMembers() -** grab the full list of members from the API
- **updateMember() -** update the entry in the API at that particular id
- **addMember() -** adds a new member to the database
- **deleteMember() -** deletes the selected member from the database

Member List:
Use the service to grab and display the list of book club members. Additionally, next to each member put a **"Read a Book"** button. You will need the following:

- An injection of the **bookClubService**
- an ngFor to display each member from the database

"Read a Book" Button:
This button will handle increasing the **booksReadCount** for a particular member. You will need the following:

- An update of the **booksReadCount** for the correct user in the database
- An update of the **booksReadCount** in list being displayed

"Kick out a Member" Button:
This button will handle removing any members from the book club. Like the **"Read a Book"** button, a **Kick out Member** button will appear to each member in the listing. Upon clicking it should:

- Remove the selected member from the database
- Display an updated listing of members

Add Member Form:
Below the list of members, place an add member form. On the form put in an input for each property except for id. On clicking submit, the new member should be added to the front and backends. You will need the following:

- An input for each property in the **bookClubMember** except for id (which should always be left blank before adding)
- On submit, the new member should be added to the API backend. Make sure the front end updates to reflect this change as well.