

Backend: Express

Node JS – Simple Login and Register Endpoints with Local JSON Storage

The following exercise contains the following subjects:

- Express
- User Registration
- User Login

Instructions

Let's create a minimal Express server with two endpoints for user registration and login, using a local JSON file to store user data.

1. Setup Express Server:

- Initialize a new Node.js project.
- Install the necessary dependencies, including `express`.

2. Create Local JSON File:

- Create a local JSON file (e.g., `users.json`) to store user information.
- Define a simple structure for user objects, including fields like `username`, `email`, and `password`.

3. Implement Register Endpoint:

- Create an Express endpoint (`/register`) for user registration.
- Write the logic to handle user registration, including validation and avoiding duplicate users.
- If the registration is successful, add the user to the `users.json` file and respond with a message "User registered successfully."
- If there are validation errors or the user already exists, respond with an appropriate error message.

4. Implement Login Endpoint:

- Create another Express endpoint (`/login`) for user login.
- Write the logic to handle user login, including validating credentials against the `users.json` file and generating a simple login success message.
- If the login is successful, respond with a message "User logged in successfully."
- If the user does not exist or the provided credentials are incorrect, respond with a login error message.

5. Test Endpoints:

- Use tools like Postman or `curl` to test the `/register` and `/login` endpoints.
- Verify that the registration and login functionality work as expected, considering the local JSON file as a simple user data storage.