



Assignment3

Part1: Simple CRUD Operations Using HTTP (7 Grades):

⚠ For all the following APIs, you must use the fs module to read and write data from a JSON file (e.g., users.json). Do not store or manage data using arrays (2 Grades).

1. Create an API that adds a new user to your users stored in a JSON file. (ensure that the email of the new user doesn't exist before) (1 Grade)

- URL: POST /user

input	output
<pre>{ "name": "User 1", "age": 27, "email": "user@email.com" }</pre>	<pre>{ "message": "User added successfully." }</pre>
<pre>{ "name": "User 2", "age": 30, "email": "user@email.com" }</pre>	<pre>{ "message": "Email already exists." }</pre>

2. Create an API that updates an existing user's name, age, or email by their ID. The user ID should be retrieved from the URL (1 Grade)
Note: Remember to update the corresponding values in the JSON file

- URL: PATCH /user/id

input	output
<pre>{ "age": 30 }</pre>	<pre>{ "message": "User age updated successfully." }</pre>
<pre>/user/99</pre>	<pre>{ "message": "User ID not found." }</pre>

3. Create an API that deletes a User by ID. The user id should be retrieved from the URL (1 Grade)

Note: Remember to delete the user from the file

- URL: DELETE /user/id

input	output
<pre>/user/1</pre>	<pre>{ "message": "User deleted successfully." }</pre>
<pre>/user/99</pre>	<pre>{ "message": "User ID not found." }</pre>

4. Create an API that gets all users from the JSON file. (1 Grade)

- URL: GET /user

input	output
<pre>—</pre>	<pre>[{ "id": 1, "name": "User 1", "age": 27, "email": "user@email.com" }]</pre>

5. Create an API that gets User by ID. (1 Grade)

- URL: GET /user/:id

- Output:

input	output
<pre>/user/1</pre>	<pre>{ "id": 1, "name": "User 1", "age": 27, "email": "user@email.com" }</pre>
<pre>/user/99</pre>	<pre>{ "message": "User not found." }</pre>

Part2: Node Internals (3 Grades):

1. What is the Node.js Event Loop? (0.5 Grade)
2. What is Libuv and What Role Does It Play in Node.js? (0.5 Grade)



Assignment3

3. **How Does Node.js Handle Asynchronous Operations Under the Hood? (0.5 Grade)**
4. **What is the Difference Between the Call Stack, Event Queue, and Event Loop in Node.js? (0.5 Grade)**
5. **What is the Node.js Thread Pool and How to Set the Thread Pool Size? (0.5 Grade)**
6. **How Does Node.js Handle Blocking and Non-Blocking Code Execution? (0.5 Grade)**

! Important Notes about postman

1. Name the endpoint with a meaningful name like 'Add User', not dummy names.
2. Save your changes on each request(ctrl+s).
3. Include the Postman collection link (export your Postman collection) in the email with your assignment link

Bonus (2 Grades)

How to deliver the bonus?

- 1- Solve the problem [Majority Element](#) on **LeetCode**
- 2- Inside your assignment folder, create a **SEPARATE FILE** and name it "bonus.js"
- 3- Copy the code that you have submitted on the website inside "bonus.js" file