

NODEJS ASSIGNMENT-2

APP USING EXPRESS SERVER

This is an extension on assignment 3, now use a database (MongoDB) to store the users, and create routes to edit and delete a user

Add **PUT** and **DELETE** routes in previous assignments

Background

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

HTTP defines a set of request methods to indicate the desired action to be performed for a given resource. Although they can also be nouns, these request methods are sometimes referred to as **HTTP verbs**. Each of them implements a different semantic, but some common features are shared by a group of them:

GET (READ OPERATION)

The GET method requests a representation of the specified resource. Requests using GET should only retrieve data.

POST (CREATE OPERATION)

The POST method submits an entity to the specified resource, often causing a change in state or side effects on the server.

PUT (UPDATE OPERATION)

The PUT method replaces all current representations of the target resource with the request payload.

PATCH (UPDATE OPERATION)

The PATCH method applies partial modifications to a resource.

DELETE (DELETE OPERATION)

The DELETE method deletes the specified resource.

Instructions

- Create a folder (eg. **Assignment-4**) for storing the files related to the assignment,
- Create a database connection to mongodb on your localhost with the **database name as assignment_4**
- Create a user schema with **name, email (unique), and a boolean field called isPromoted**, default value null

The two additional routes are added below

- **PUT** /user/:id - Edit User Route, toggle the isPromoted field value to true / false
- **DELETE** /user/:id - Delete the user with the id

- The user list / card in the base route should show two buttons (Promote / Demote, and Delete) in every user list / card
- Add the buttons in the format below


```
<form method="POST" action="/users/<%= user._id %>?_method=PUT">  
  <button id="edit-<%= user.name %>" type="submit">Select</button>  
</form>
```

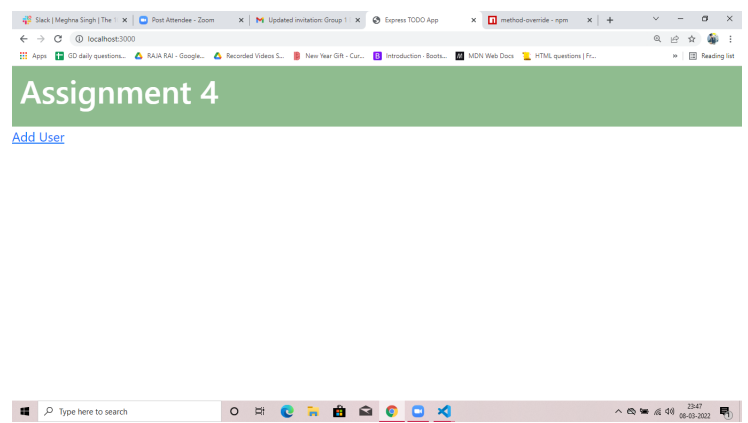
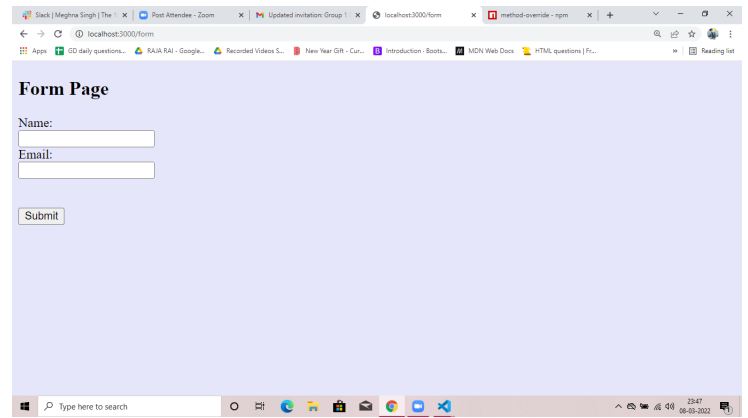
- `<form method="POST" action="/users/<%= user._id %>?_method=DELETE">`
`<button id="delete-<%= user.name %>">Delete</button>`
`</form>`

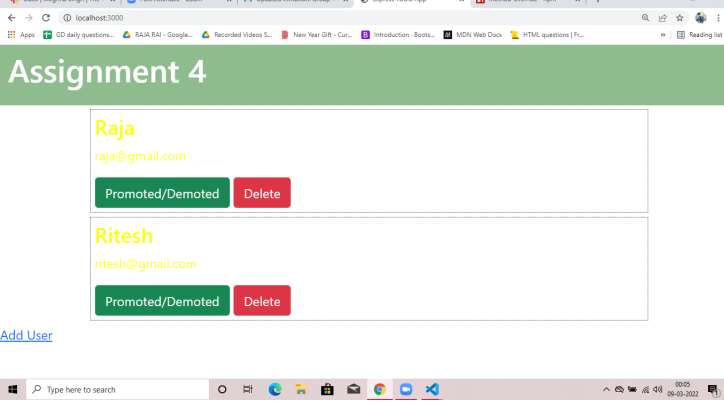
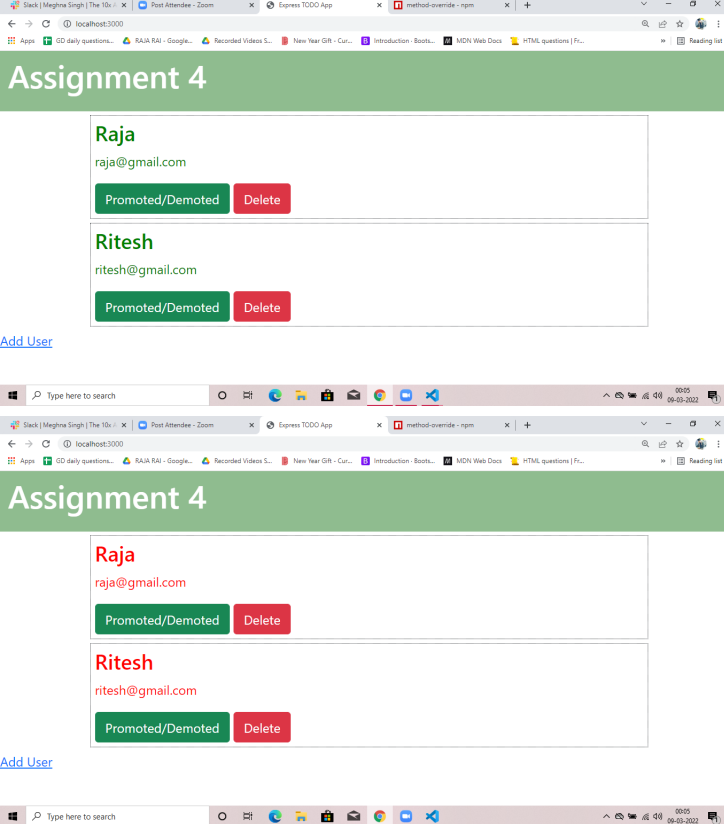
Use npm method-override package to override POST to PUT/DELETE method on backend

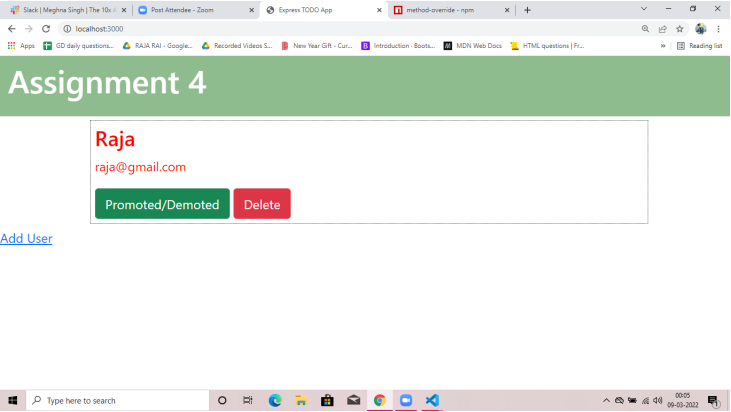
- The Promote / Demote button should call the PUT request with the user id for that user and should change the user isPromoted field to true (if it was null or false) else false
- Show the users with isPromoted value as null in Yellow, isPromoted true value in Green, isPromoted false value in Red
- Add one more button called Delete on every user list / card and on clicking the Delete button, it should call the DELETE route with the user id and the user should get deleted on the backend
- Give the edit button and delete button their unique ids in format (edit-) and (delete-) for their corresponding users
- Redirect to the base route after PUT and DELETE route

Evaluation

Test case number	Steps	Example
1	<ol style="list-style-type: none"> 1. Open command line 2. Do <code>cd</code> to the directory where your folder is present 3. Run the command <code>npm run dev</code> 4. It should print "server is running" 	 <pre> PS C:\Users\Aj\Kumar\Desktop\NodeJs\solutions\3> npm run dev > assignment@1.0.0 dev C:\Users\Aj\Kumar\Desktop\NodeJs\solutions\3 > nodedemon app.js [nodedemon] 2.0.15 [nodedemon] to restart at any time, enter 'rs' [nodedemon] watching path(s): '.' [nodedemon] watching extensions: js,mjs,json [nodedemon] starting 'node app.js index.js' Server is running </pre>

2	<p>1. It should start a server at the mentioned port (http://localhost:3000/), rendering the initial table.</p>	 <p>The screenshot shows a web browser window with the address bar at 'localhost:3000'. The page has a green header with the text 'Assignment 4' and a blue link labeled 'Add User' below it. The browser's taskbar at the bottom shows various open applications and the system clock at 23:47 on 08-03-2022.</p>
3	<p>1. Page should contain a hyperlink for adding new users which should redirect to form page (eg. Add User)</p>	 <p>This screenshot is identical to the one in the first row, showing the 'Assignment 4' page with the 'Add User' link.</p>
4	<p>1. On clicking Add User, it should redirect to '/form' (http://localhost:3000/form) form page. 2. After filling the input fields (i.e. name and email), click the submit button.</p>	 <p>The screenshot shows the browser at 'localhost:3000/form'. The page is titled 'Form Page' and contains two input fields labeled 'Name:' and 'Email:', followed by a 'Submit' button. The browser's taskbar at the bottom is the same as the previous screenshots.</p>

5	<ol style="list-style-type: none"> 1. On clicking Submit button, it should redirect to "/" i.e. home page, 2. Table which you saw earlier should contain the data that you filled on the form page in the input fields. 3. Initially, the data should be in yellow (i.e. isPromoted value as null should appear in Yellow) 	
6	<ol style="list-style-type: none"> 1. On clicking Promoted/Demoted button, color of data should change to green (i.e. isPromoted true value should appear in Green) 2. On clicking Promoted/Demoted button again, color of data should change to red (i.e. isPromoted false value should appear in Red) 	

7	<ol style="list-style-type: none">1. On clicking the Delete button, data should be removed from the database and it should not appear on the screen anymore. Eg. second data got deleted after clicking the Delete button.	
---	---	--

Submission

1. Zip the entire folder
2. Submit the Zip File
3. Submit Github link