Programs to practice in the Lab

Topics Covered: Node JS Modules (http, fs)

Question 1: Create a simple routing system within the Node.js http module that handles different paths (e.g., /home, /about, /contact). Send a different response for each route, such as "Welcome to Home," "About Us," etc.

Question 2: Write a Node.js server that parses query parameters from a GET request. For example, if the request is /greet?name=John, the server should respond with "Hello, John!".

Question 3: Implement a file server using the http module that serves a static HTML file when a client requests a GET/index.html.

Question 4: Extend the HTTP server to respond with a JSON object when a request is made to /api/data. The server should send a JSON object with name, age, and city properties. Make sure the correct Content-Type header is set to application/json.

Question 5: Create a simple server that displays a registration form and validates user data based only on GET requests. The server should:

- Serve an HTML form when the user accesses the root (/), with fields like username, email, and password.
- If a user requests the form with query parameters (e.g., /register?username=test&email=test@example.com&password=abc123), validate the data in the query string.
    - The username should be at least 3 characters.
    - The email should be in a valid email format.
    - The password should be at least 6 characters.
- Respond with a success message if the data is valid.
- Respond with an error message if any field is invalid.

@@@