Task 4: File System Module

Objective:

Read from and write to a file using Node.js `fs` module. Ensure the program successfully creates, reads, and writes to a file.

Prerequisites:

- Basic understanding of JavaScript.
- Node.js installation.

Concepts:

- **Reading from a File:**
  - The `fs` module provides methods to interact with the file system, such as reading from and writing to files.
  - The `fs.readFile` method reads the content of a file asynchronously.

  Example:

```
// readFileExample.js
const fs = require('fs');

fs.readFile('example.txt', 'utf8', (err, data) => {
  if (err) throw err;
  console.log(data); // Output the content of example.txt
});
```

- **Writing to a File:**
  - The `fs.writeFile` method writes data to a file asynchronously.

  Example:

```
// writeFileExample.js
const fs = require('fs');

const content = 'This is some content to write to the file';

fs.writeFile('example.txt', content, (err) => {
  if (err) throw err;
  console.log('File has been saved!');
});
```

Setup:

1. **Install Node.js:**

   Ensure Node.js is installed on your machine. You can access the instructions here:

- 

  Detailed Instructions for Installing Node.js and NPM and integration...
  For Windows: — Download Node.js Installer: — Visit the official Node.js website:...
  w3o NFThing Last Edited 7/3/2024

Tasks:

1. **Create and Write to a File:**
   - **Task:**
     - Create a file named `writeToFile.js` and write your own code that:
       - Uses the `fs` module to create and write a string of your choice to a file named `output.txt`.
   - **Outcome:**
     - Ensure the file `output.txt` is created and contains the written string.
2. **Read from a File:**
   - **Task:**
     - Create a file named `readFromFile.js` and write your own code that:
       - Uses the `fs` module to read the content of `output.txt` and logs it to the console.
   - **Outcome:**
     - Ensure the content of `output.txt` is correctly read and logged to the console.
3. **Append to a File:**
   - **Task:**
     - Create a file named `appendToFile.js` and write your own code that:
       - Uses the `fs` module to append a new string to `output.txt`.
   - **Outcome:**
     - Ensure the new string is correctly appended to the existing content in `output.txt`.

Instructions:

- **Perform the following tasks:**
  - Write the required code in separate files (`writeToFile.js`, `readFromFile.js`, `appendToFile.js`).
  - Run each file using Node.js to ensure the code executes without errors and demonstrates the use of the `fs` module.

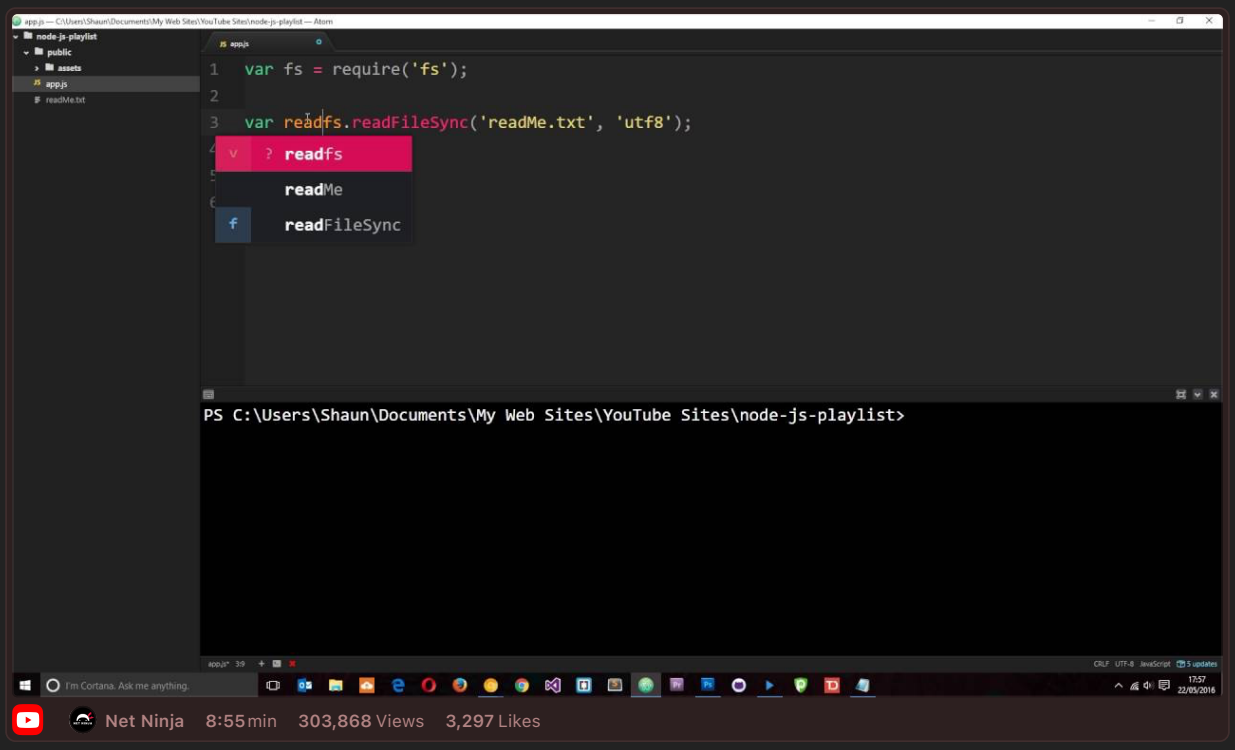
Resources:

- File system | Node.js v22.4.0 Documentation
  The node:fs module enables interacting with the file system in a — way modeled on standard POSIX f...
  https://nodejs.org/api/fs.html

- 
  Index | Node.js v22.3.0 Documentation

  https://nodejs.org/en/docs/

Videos:



GitHub Instructions:

1. **Open in Visual Studio Code:**
   - After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, Visual Studio Code (VSCode) will open the repository directly.
   - If prompted, select "Open" or "Allow" to open the repository in VSCode.
2. **Complete the Task:**
   - Write your solution in the respective files (`writeToFile.js` , `readFromFile.js` , `appendToFile.js`).
3. **Run and Test Your Code:**
   - Run your code to ensure it works correctly. Use the following command for each file:

     ```
     node <filename>.js
     ```

4. **Commit Your Changes:**
   - Commit your changes with a meaningful message:

```
git commit -m "Completed File System Module task"
```

5. **Push Your Changes to Your Forked Repository:**
   - Push your changes to your forked repository:

```
git push origin main
```

6. **Create a Pull Request:**
   - Go to your forked repository on GitHub.
   - Click on the "Pull Requests" tab.
   - Click the "New Pull Request" button.
   - Ensure the base repository is the original template repository and the base branch is `main`.
   - Ensure the head repository is your forked repository and the compare branch is `main`.
   - Click "Create Pull Request".
   - Add a title and description for your pull request and submit it.

Summary of Commands:

```
# Fork the repository on GitHub

# Clone the forked repository
git clone https://github.com/your-github-username/repository-name.git
cd repository-name

# Complete the task by writing and running the code in the specified files

# Add, commit, and push your changes
git commit -m "Completed File System Module task"
git push origin main

# Create a pull request on GitHub
```