Movie Popularity Content Based Database System using Cloud Computing Based on Deep Learning and AI

How to Install VS Studio Code (IDE):-

Step 1: Download VS Code

1. **Go to the Visual Studio Code website**:
   - Open your web browser and navigate to the Visual Studio Code download page.
2. **Download the installer**:
   - Click on the "Download for Windows" button. This will download the VS Code installer (`VSCodeSetup.exe`).

Step 2: Install VS Code

1. **Run the installer**:
   - Locate the downloaded `VSCodeSetup.exe` file (usually in your Downloads folder) and double-click it to start the installation process.
2. **Accept the license agreement**:
   - When the installer opens, read and accept the license agreement, then click "Next".
3. **Select the installation location**:
   - Choose the destination folder where you want to install VS Code. The default location is usually fine. Click "Next".
4. **Select additional tasks**:
   - You will be presented with several options. It's recommended to check the following:
     - "Create a desktop icon" (optional, but useful for quick access)
     - "Add to PATH" (important for running VS Code from the command line)
     - "Register Code as an editor for supported file types" (optional)
     - "Add 'Open with Code' action to Windows Explorer file context menu" (optional, but useful for quickly opening files/folders in VS Code)
     - "Add 'Open with Code' action to Windows Explorer directory context menu" (optional)
5. **Install**:
   - Click "Next" and then "Install" to begin the installation process. This may take a few minutes.
6. **Launch VS Code**:
   - Once the installation is complete, you can choose to launch VS Code immediately by checking the "Launch Visual Studio Code" option, then click "Finish".

Step 3: Launch VS Code

- **From the Start Menu**:
  - Click on the Start menu and type "Visual Studio Code". Click on the Visual Studio Code icon to open the application.

- **From the Desktop Icon**:
    - If you chose to create a desktop icon during installation, you can double-click the VS Code icon on your desktop to launch the application.
- **From the Command Line**:
    - Open Command Prompt or PowerShell and type `code`, then press Enter. This will open VS Code if you added it to the PATH during installation.

Step 4: Install Extensions (Optional but Recommended)

VS Code supports a wide variety of extensions that enhance its functionality. Here are some useful extensions for React.js development:

1. **Open the Extensions View**:
    - Click on the Extensions icon in the Activity Bar on the side of the window or press `Ctrl+Shift+X`.
2. **Search for Extensions**:
    - In the Extensions view, type the name of the extension you want to install in the search bar.
3. **Install Extensions**:
    - Click the Install button next to the extension name.

Recommended Extensions for React.js:

1. **ESLint**:
    - Integrates ESLint JavaScript into VS Code.
    - Search for `ESLint` and click Install.
2. **Prettier - Code formatter**:
    - An opinionated code formatter.
    - Search for `Prettier - Code formatter` and click Install.
3. **JavaScript (ES6) code snippets**:
    - Code snippets for JavaScript in ES6 syntax.
    - Search for `JavaScript (ES6) code snippets` and click Install.
4. **React Native Tools**:
    - Debugging and integrated commands for React Native.
    - Search for `React Native Tools` and click Install.
5. **Bracket Pair Colorizer**:
    - Adds colors to matching brackets.
    - Search for `Bracket Pair Colorizer` and click Install.

Step 5: Configure VS Code for Your Project

Once VS Code is installed and running, you can open your React.js project:

1. **Open a folder**:
    - Go to File > Open Folder... and select your project folder.
2. **Open a file**:
    - Use the File Explorer on the left side to navigate to a file and click to open it.
3. **Terminal**:

- Open an integrated terminal in VS Code by going to View > Terminal or pressing `Ctrl+`` (backtick). This terminal can be used to run your development server or other command-line tools.

Steps to install Node.js :-

Step 1: Download the Node.js Installer

1. **Visit the official Node.js website**:
   - Open your web browser and go to the Node.js download page.
2. **Choose the correct installer**:
   - You will see two versions: the LTS (Long Term Support) version and the Current version. The LTS version is recommended for most users.
   - Click the Windows Installer button to download the `.msi` file. This will be something like `node-v16.x.x-x64.msi` for the 64-bit version.

Step 2: Install Node.js

1. **Run the installer**:
   - Locate the downloaded `.msi` file in your Downloads folder and double-click it to start the installation process.
2. **Follow the setup wizard**:
   - The Node.js Setup wizard will open. Click "Next" to continue.
3. **Accept the license agreement**:
   - Read the license agreement, check the box to accept the terms, and click "Next".
4. **Choose the installation location**:
   - The default location is usually fine. Click "Next" to continue.
5. **Select components**:
   - You can leave all default settings checked, including "Node.js runtime", "npm package manager", "Online documentation shortcuts", and "Add to PATH".
   - Click "Next" to continue.
6. **Install Tools for Native Modules**:
   - You might see an option to install tools for native modules. This includes Python and Visual Studio Build Tools. This step is optional and can be skipped if you don't plan on compiling native modules.
   - Click "Next" to continue.
7. **Begin the installation**:
   - Click the "Install" button to start the installation. You may need to grant administrative privileges to proceed.
8. **Complete the installation**:
   - Once the installation is complete, click the "Finish" button to exit the setup wizard.

Step 3: Verify the Installation

1. **Open Command Prompt**:
   - Press `Win + R`, type `cmd`, and press Enter.
2. **Check the Node.js version**:
   - In the Command Prompt window, type:

     ```
     node -v
     ```

   - You should see the version number of Node.js printed on the screen, which confirms that Node.js is installed correctly.
3. **Check the npm version**:
   - In the Command Prompt window, type:

     ```
     npm -v
     ```

   - You should see the version number of npm (Node Package Manager) printed on the screen, confirming that npm is installed correctly.

Step 4: Update npm (Optional)

npm is frequently updated. To ensure you have the latest version, you can update npm using the following command:

1. **Open Command Prompt**.
2. **Run the update command**:

   ```
   npm install -g npm
   ```

Setup For Project:-

Step 1: Create a Vite Project

1. **Open a terminal**:
   - You can use Command Prompt, PowerShell, or the integrated terminal in Visual Studio Code.
2. **Run the Vite project creation command**:
   - Execute the following command to create a new Vite project:

     ```
     npm create vite@latest
     ```

3. **Follow the prompts**:
   - You will be prompted to enter the project name and choose a template. For example:

```
Project name: » my-vite-project
Select a framework: » react
Select a variant: » react-ts
```

   - Replace `my-vite-project` with your desired project name and select the framework and variant you prefer.
4. **Navigate to your project directory**:

```
cd my-vite-project
```

5. **Install dependencies**:

```
npm install
```

Step 2: Open the Project in Visual Studio Code

1. **Launch Visual Studio Code**:
   - Open Visual Studio Code from the Start menu or by typing `code` in your terminal if it is added to the PATH.
2. **Open the project folder**:
   - In Visual Studio Code, go to `File > Open Folder` and select the folder of your newly created Vite project (e.g., `my-vite-project`).

Step 3: Run the Vite Development Server

1. **Open a terminal in Visual Studio Code**:
   - Go to `View > Terminal` or press `Ctrl+`` (backtick) to open the integrated terminal.
2. **Start the development server**:
   - In the terminal, run:

```
npm run dev
```

   - This will start the Vite development server, and you should see output indicating that the server is running at `http://localhost:3000` (or another port if 3000 is in use).

Step 4: Open the Application in the Browser

1. **Open your browser**:
   - Navigate to `http://localhost:3000` in your web browser.
2. **View your Vite application**:
   - You should see the default Vite application running.

How to install MUI in Project

Step 1: Ensure Your Project is Set Up

Make sure you have already created your Vite project and have it open in Visual Studio Code, as described in the previous instructions. The project should be running correctly with `npm run dev`.

Step 2: Install MUI

1. **Open a terminal in Visual Studio Code**:
   - Go to `View > Terminal` or press `Ctrl+` (backtick) to open the integrated terminal.
2. **Install MUI Core and Icons**:
   - Run the following command to install the MUI core package and the MUI icons package:

```
npm install @mui/material @emotion/react @emotion/styled
npm install @mui/icons-material
```

Step 3: Use MUI in Your Project

To verify that MUI is installed correctly, you can use a simple MUI component in your project.

1. **Open the main component file**:
   - Open the `src/App.jsx` or `src/App.tsx` file, depending on whether you are using JavaScript or TypeScript.
2. **Import MUI components**:
   - At the top of your `App.jsx` or `App.tsx` file, import the necessary MUI components. For example, to use the `Button` component, you would add:

```
import React from 'react';
import Button from '@mui/material/Button';
```

3. **Use MUI components**:
   - Replace the existing code in your `App.jsx` or `App.tsx` file with some MUI components to test the installation. For example:

```
import React from 'react';
import Button from '@mui/material/Button';

function App() {
  return (
    <div style={{ textAlign: 'center', marginTop: '50px' }}>
      <Button variant="contained" color="primary">
        Hello MUI
      </Button>
    </div>
  );
}
```

```
export default App;
```

4. **Save the file**:
    - Save the changes to your `App.jsx` or `App.tsx` file.

Step 4: Verify Your Application

1. **Ensure the development server is running**:
    - If the development server is not running, start it by running:

      ```
      npm run dev
      ```

2. **Open your browser**:
    - Navigate to `http://localhost:3000`.
3. **Check the output**:
    - You should see a button with the text "Hello MUI" styled with Material-UI.

Summary

1. **Open the terminal in your Vite project** in Visual Studio Code.
2. **Install MUI packages** using npm:

   ```
   npm install @mui/material @emotion/react @emotion/styled
   npm install @mui/icons-material
   ```

3. **Import and use MUI components** in your `App.jsx` or `App.tsx` file.
4. **Run the development server** and verify the MUI components are displayed correctly in your application.