**React Js**

**Name:** React Js Tutorials in Hindi (CodeWithHarry)

**Tutorial Url:** https://youtube.com/playlist?list=PLu0W\_9lII9agx66oZnT6IyhcMIbUMNMdt&si=etlVrzs3EPV\_bDNk

|  |  |
| --- | --- |
| **1. Introduction to React Js + Installation (Completed)** | **Date: 02/12/24** |

**Installation**

A screenshot of a computer

Description automatically generated

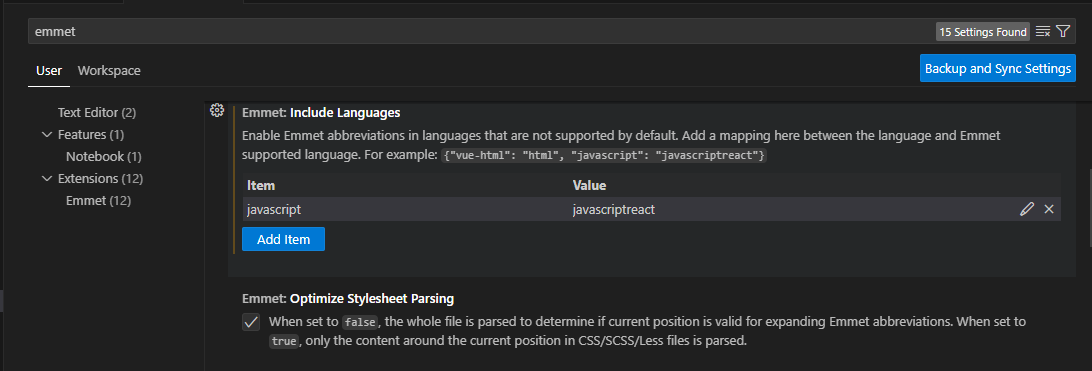
* Install Node.js
* Install VS Code
* Install React Dev Tools

A white background with black and white clouds

Description automatically generated

**Extension For VS Code**

* Live Server
* Auto save (afterDelay)
* Thunder Client
* ES7 React/Redux/GraphQL/React-Native snippets
* ES7+ React/Redux/React-Native snippets
* Bracket Pair Colorizer
* Auto Rename Tag
* Prettier - Code formatter
* Manage -> Command Palette -> Word Wrap (ALT+Z)



|  |  |
| --- | --- |
| **2. Creating our first react app using create-react-app (Completed)** | **Date: 03/12/24** |

|  |
| --- |
| **Create File on C: Drive**  **Creating First React App**  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\my-app> **node -v**  v20.18.1  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\my-app> **npx -v**  10.9.2  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\my-app> **npm -v**  10.9.2  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> **npx create-react-app textutils**  Creating a new React app in C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils.  Installing packages. This might take a couple of minutes.  Installing react, react-dom, and react-scripts with cra-template...  added 1315 packages in 2m  260 packages are looking for funding  run `npm fund` for details  Initialized a git repository.  Installing template dependencies using npm...  added 46 packages, and changed 1 package in 11s  264 packages are looking for funding  run `npm fund` for details  Removing template package using npm...  removed 1 package, and audited 1361 packages in 5s  264 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details.  Created git commit.  Success! Created textutils at C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils  Inside that directory, you can run several commands:  npm start  Starts the development server.  npm run build  Bundles the app into static files for production.  npm test  Starts the test runner.  npm run eject  Removes this tool and copies build dependencies, configuration files  and scripts into the app directory. If you do this, you can’t go back!  cd textutils  npm start  Happy hacking!  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> **cd "textutils"**  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> **npm start**  > textutils@0.1.0 start  > react-scripts start  Starting the development server...  One of your dependencies, babel-preset-react-app, is importing the  "@babel/plugin-proposal-private-property-in-object" package without  declaring it in its dependencies. This is currently working because  "@babel/plugin-proposal-private-property-in-object" is already in your  node\_modules folder for unrelated reasons, but it may break at any time.  Compiled successfully!  You can now view textutils in the browser.  Local: http://localhost:3000  On Your Network: http://192.168.240.19:3000  Note that the development build is not optimized.  To create a production build, use npm run build.  webpack compiled successfully |

|  |
| --- |
| **Error:** installing template dependencies using npm... npm error code eresolve npm error eresolve unable to resolve dependency tree npm error npm error while resolving  **Solution:** PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> **npm config set legacy-peer-deps true**  **Error:** cannot find module 'ajv/dist/compile/codegen'  **Solution:** PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\my-app> **npm install --save-dev ajv@^7**  **Error:** one of your dependencies, babel-preset-react-app, is importing the "@babel/plugin-proposal-private-property-in-object" package without declaring it in its dependencies. this is currently working because "@babel/plugin-proposal-private-property-in-object" is already in your node\_modules folder for unrelated reasons, but it may break at any time.  **Solution:** PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\my-app> **npm install --save-dev @babel/plugin-proposal-private-property-in-object --force** |

|  |  |
| --- | --- |
| **3. JavaScript Refresher** | **Date: 11/12/24** |

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Document</title>      <script>          "use strict";          // let a = 34;          // let a = 'This is me'          // const a = 45; (Uncaught TypeError: Assignment to constant variable.)          // a = 4554 + 4          let a = 45;          a = 4554 + 4;          console.log(a);          // Number          // Object          // Booleans - true or False          // String          // undefined          // null          // function hello() {          //     console.log("Hey Harry!");          // }          // let obja = {          //     r: 34,          //     m: 64,          //     func: function myfunc(number) {          //         console.log("The number is " + number);          //     }          // }          // document.addEventListener("click", function click() {          //     console.log("Clicked");          //     // alert("hello!");          //     let conf = confirm("Are you sure?");          //     console.log(conf);          // });          // for (let index = 0; index < array.length; index++) {          //     const element = array[index];          // }          let arr = [34, 21, 23];          console.log(this);          // function myfunc(number) {          //     console.log("The number is " + this);          // }          // let mobj = {          //     a: this,          //     myfunc : (number) => {          //         console.log("The number is " + this);          //     }          // }          let mobj = {              a: this,              myfunc : function (number) {                  console.log(this);              }          }          mobj.myfunc();          console.log(mobj.a);            setTimeout(() => {              console.log("yayyy");          }, 3000);          console.log("mai neeche wali script hoon");          // arr.forEach(element => {            // });        </script>  </head>  <body>      <h1>I am learning js</h1>  </body>  </html> |

|  |  |
| --- | --- |
| **4. Understanding JSX** | **Date: 11/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import logo from './logo.svg';  import './App.css';  function App() {    return (      <> {/\* Fragments \*/}        <h1> This is me </h1>        <div className="App">          <header className="App-header">            <img src={logo} className="App-logo" alt="logo" />            <p>              Edit <code>src/App.js</code> and save to reload.            </p>            <a              className="App-link"              href="https://reactjs.org"              target="\_blank"              rel="noopener noreferrer"            >              Learn React With Amandeep            </a>          </header>        </div>      </>    );  }  export default App; |

|  |
| --- |
| Remove everythings and create own JSX Components  File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  let name = "Amandeep";  function App() {    return (      <>        <nav>          <li>Home</li>          <li>About</li>          <li>Contact</li>        </nav>        <div className="container">          <h1>Hello {name}</h1>          <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Consequuntur commodi dolorem ex perferendis illum sunt quasi autem reiciendis a quos? Praesentium corrupti aspernatur eligendi est numquam aut corporis non suscipit, ut obcaecati ipsum optio!</p>        </div>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.css  .App {    text-align: center;  }  .App-logo {    height: 40vmin;    pointer-events: none;  }  @media (prefers-reduced-motion: no-preference) {    .App-logo {      animation: App-logo-spin infinite 20s linear;    }  }  .App-header {    background-color: #282c34;    min-height: 100vh;    display: flex;    flex-direction: column;    align-items: center;    justify-content: center;    font-size: calc(10px + 2vmin);    color: white;  }  .App-link {    color: #61dafb;  }  @keyframes App-logo-spin {    from {      transform: rotate(0deg);    }    to {      transform: rotate(360deg);    }  }  .blank {    color: red;  } |

|  |  |
| --- | --- |
| **5. Project 1: Setup + Adding Bootstrap to React** | **Date: 11/12/24** |

|  |
| --- |
| **Important point**   1. When we give react app to anyone, we **delete “textutils\node\_modules”** folder. Because here are all dependencies are there. If we needed, we create it by using this **command “npm install”.** 2. In “public” folder, we couldn’t store the **security file or Secret file.** 3. Note that the development build is not optimized. To create a production build, use command “**npm run build”**. |

|  |
| --- |
| **Clean all these things.**   1. Apply Bootstrap library in “index.html” file. 2. Empty the “index.css” file. 3. Remove the components inside the fragments in “App.js” file. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description" content="Web site created using create-react-app" />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>React App</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\index.css  **Empty it.** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  let name = "Amandeep";  function App() {    return (      <>      </>    );  }  export default App; |

**Start the project**

|  |
| --- |
| **When you create new components apply these things**   1. Open and closing tag <tag> </tag>, <tag /> 2. class -> className 3. for -> htmlFor 4. tabindex -> tabIndex 5. herf=“#” -> herf=“/” |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description" content="Web site created using create-react-app" />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>React App</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\index.css  **Empty file.** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  function App() {    return (      <>        <nav className="navbar navbar-expand-lg bg-body-tertiary">          <div className="container-fluid">            <a className="navbar-brand" href="/">Textutils</a>            <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">              <span className="navbar-toggler-icon"></span>            </button>            <div className="collapse navbar-collapse" id="navbarSupportedContent">              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                <li className="nav-item">                  <a className="nav-link active" aria-current="page" href="/">Home</a>                </li>                <li className="nav-item">                  <a className="nav-link" href="/">About</a>                </li>              </ul>              <form className="d-flex" role="search">                <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                  <button className="btn btn-outline-success" type="submit">Search</button>              </form>            </div>          </div>        </nav>      </>    );  }  export default App; |

|  |  |
| --- | --- |
| **6. Understanding Props and PropTypes in React** | **Date: 11/12/24** |

|  |
| --- |
| **“ES7 React/Redux/GraphQL/React-Native snippets”**  **This Extension give the ability to create format with shortcut. Go through the website if you want to know more.**  [**https://marketplace.visualstudio.com/items?itemName=rodrigovallades.es7-react-js-snippets**](https://marketplace.visualstudio.com/items?itemName=rodrigovallades.es7-react-js-snippets) |

**Export and Import Module (Explanation)**

A screenshot of a computer program

Description automatically generated

**Single Export and Import**

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\module1.mjs  const a = "Amandeep";  const b = "Harshal";  const c = "Neelam";  const d = "Nidhi";  export default b; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\module2.mjs  import ui from './module1.mjs';  console.log(ui);  terminal:  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> node .\module2.mjs  Neelam  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> node .\module2.mjs  Harshal |

**Multiple Export and Import**

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\module1.mjs  const a = "Amandeep";  const b = "Harshal";  const c = "Neelam";  const d = "Nidhi";  export default c;  export {a};  export {b};  export {d}; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\module2.mjs  import ui, {a, b, d} from './module1.mjs';  console.log(ui);  console.log(d);  console.log(a);  console.log(b);  Terminal:  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> **node .\module2.mjs**  Neelam  Nidhi  Amandeep  Harshal |

|  |
| --- |
| **Important Point**  Creating “components” folder. The file names in the components folder start with a capital letter, such as 'Navbar.js'. |

|  |
| --- |
| **rfc (React Function-base Components) shortcut is used to create structure**  File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  export default function Navbar(props) {      return (          <nav className="navbar navbar-expand-lg bg-body-tertiary">              <div className="container-fluid">                  <a className="navbar-brand" href="/">{props.title}</a>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <a className="nav-link active" aria-current="page" href="/">Home</a>                          </li>                          <li className="nav-item">                              <a className="nav-link" href="/">{props.aboutText}</a>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| **Import Navbar.js in App.js**  File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import Navbar from './components/Navbar';  function App() {    return (      <>        <Navbar title="textutils" aboutText="About Us" />        {/\*<Navbar />\*/}  {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        {/\* <Navbar /> \*/}      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description" content="Web site created using create-react-app" />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>React App</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |  |
| --- | --- |
| **7. Understanding State & Handling Events in React** | **Date: 12/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  function App() {    return (      <>        <Navbar title="textutils" aboutText="About Us" />        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <div className="container my-3">          <TextForm heading="Enter the text for analyze below" />        </div>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {      const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();   // 3. "text" convert to Uppercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);    // 5. now "setText(event.target.value)" change the value to uppercase in textarea.      }      const [text, setText] = useState('Enter text here');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <div>              <h1>{props.heading}</h1>    {/\* 2 \*/}              <div className="mb-3">                  <textarea className="form-control" value={text} onChange={handleOnChange} id="myBox" rows="10"></textarea>                  {/\* 1. "text" is written here. (Inside "text" value sent) \*/}                </div>              <button className="btn btn-primary" onClick={handleUpClick}>Convert to Uppercase</button>              {/\* 2. When we click on "handleUpClick". Is gone to "handleUpClick" function \*/}          </div>      );  } |

|  |  |
| --- | --- |
| **8. Adding more Logic to TextUtils**  **9. Exercise 1: Enhancing TextUtils**  **10. Creating "Enable Dark Mode" Button Using useState Hook** | **Date: 14/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  // import TextForm from './components/TextForm';  function App() {    return (      <>        <Navbar title="textutils" aboutText="About Us" />        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <div className="container my-3">          {/\* <TextForm heading="Enter the text for analyze below" /> \*/}          <About />        </div>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  export default function Navbar(props) {      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <a className="navbar-brand" href="/">{props.title}</a>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <a className="nav-link active" aria-current="page" href="/">Home</a>                          </li>                          <li className="nav-item">                              <a className="nav-link" href="/">{props.aboutText}</a>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {      const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();   // 3. "text" convert to Uppercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleLoClick = () => {          // console.log("Lowercase was clicked :" + text);          let newText = text.toLowerCase();   // 3. "text" convert to Lowercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleClearClick = () => {          // console.log("Clear was clicked :" + text);          let newText = "";   // 3. "text" convert to Clear and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);    // 5. now "setText(event.target.value)" change the value to uppercase in textarea.      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container">                  <h1>{props.heading}</h1>    {/\* 2 \*/}                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} id="myBox" rows="10"></textarea>                      {/\* 1. "text" is written here. (Inside "text" value sent) \*/}                  </div>                  <button className="btn btn-primary mx-1" onClick={handleUpClick}>Convert to Uppercase</button>                  {/\* 2. When we click on "handleUpClick". Is gone to "handleUpClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleLoClick}>Convert to Lowercase</button>                  {/\* 2. When we click on "handleLoClick". Is gone to "handleLoClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleClearClick}>Clear Text</button>                  {/\* 2. When we click on "handleClearClick". Is gone to "handleClearClick" function \*/}              </div>              <div className="container my-3">                  <h1>Your text summary</h1>                  <p>{text.split(" ").length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text}</p>              </div>          </>      );  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\About.js  import React, { useState } from 'react'  export default function About() {      // 1. I have create this 'myStyle' variable apply on the component like ("container", "accordionExample", "accordion-button", "accordion-body")      // let myStyle = {      //     color: 'white',      //     backgroundColor: 'black'      // }      // 2. Then apply on the useState()      const [myStyle, setMyStyle] = useState({          color: 'black',          backgroundColor: 'white',          border: '2px solid white'      })      // 5. Create "useState" for btnText      const [btnText, setBtnText] = useState("Enable Dark mode")      // 4. Apply "toggleStyle" on here.      const toggleStyle = () => {          if (myStyle.color === 'black') {              setMyStyle({                  color: 'white',                  backgroundColor: 'black',                  border: '2px solid white'              })              setBtnText("Enable Light mode")          }          else {              setMyStyle({                  color: 'black',                  backgroundColor: 'white'              })              setBtnText("Enable Dark mode")          }      }      return (          <div className='container' style={myStyle}>              <h1 className='my-3'>About Us</h1>              <div className="accordion" id="accordionExample" style={myStyle}>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button" type="button" style={myStyle} data-bs-toggle="collapse" data-bs-target="#collapseOne" aria-expanded="true" aria-controls="collapseOne">                              Accordion Item #1                          </button>                      </h2>                      <div id="collapseOne" className="accordion-collapse collapse show" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              <strong>This is the first item's accordion body.</strong> It is shown by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.                          </div>                      </div>                  </div>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button collapsed" style={myStyle} type="button" data-bs-toggle="collapse" data-bs-target="#collapseTwo" aria-expanded="false" aria-controls="collapseTwo">                              Accordion Item #2                          </button>                      </h2>                      <div id="collapseTwo" className="accordion-collapse collapse" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              <strong>This is the second item's accordion body.</strong> It is hidden by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.                          </div>                      </div>                  </div>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button collapsed" style={myStyle} type="button" data-bs-toggle="collapse" data-bs-target="#collapseThree" aria-expanded="false" aria-controls="collapseThree">                              Accordion Item #3                          </button>                      </h2>                      <div id="collapseThree" className="accordion-collapse collapse" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              <strong>This is the third item's accordion body.</strong> It is hidden by default, until the collapse plugin adds the appropriate classes that we use to style each element. These classes control the overall appearance, as well as the showing and hiding via CSS transitions. You can modify any of this with custom CSS or overriding our default variables. It's also worth noting that just about any HTML can go within the <code>.accordion-body</code>, though the transition does limit overflow.                          </div>                      </div>                  </div>              </div>              <div className="container my-3">                  {/\* 3. Create the "onClick={toggleStyle}" \*/}                  <button onClick={toggleStyle} type='button' className='btn btn-primary my-3'>{btnText}</button>              </div>          </div>      )  } |

|  |  |
| --- | --- |
| **11. Exercise 1: Solutions + Shoutouts** | **Date: 14/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {      const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();   // 3. "text" convert to Uppercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleLoClick = () => {          // console.log("Lowercase was clicked :" + text);          let newText = text.toLowerCase();   // 3. "text" convert to Lowercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleClearClick = () => {          // console.log("Clear was clicked :" + text);          let newText = "";   // 3. "text" convert to Clear and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);    // 5. now "setText(event.target.value)" change the value to uppercase in textarea.      }      const handleCopy = () => {          // console.log("i am copy");          var text = document.getElementById("myBox");          text.select();          text.setSelectionRange(0, 9999);          navigator.clipboard.writeText(text.value);  // 6. now "navigator.clipboard.writeText(text.value)" copy the textarea value.      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container">                  <h1>{props.heading}</h1>    {/\* 2 \*/}                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} id="myBox" rows="10"></textarea>                      {/\* 1. "text" is written here. (Inside "text" value sent) \*/}                  </div>                  <button className="btn btn-primary mx-1" onClick={handleUpClick}>Convert to Uppercase</button>                  {/\* 2. When we click on "handleUpClick". Is gone to "handleUpClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleLoClick}>Convert to Lowercase</button>                  {/\* 2. When we click on "handleLoClick". Is gone to "handleLoClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleClearClick}>Clear Text</button>                  {/\* 2. When we click on "handleClearClick". Is gone to "handleClearClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleCopy}>Copy Text</button>                  {/\* 2. When we click on "handleCopy". Is gone to "handleCopy" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleExtraSpaces}>Extra Spaces</button>                  {/\* 2. When we click on "handleExtraSpaces". Is gone to "handleExtraSpaces" function \*/}              </div>              <div className="container my-3">                  <h1>Your text summary</h1>                  <p>{text.split(" ").length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text}</p>              </div>          </>      );  } |

|  |  |
| --- | --- |
| **12. Improving Dark Mode & Refactoring App Component** | **Date: 16/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  // import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import React, { useState } from 'react';  function App() {    const [mode, setMode] = useState('light'); // 1. Whether dark mode is enabled or not by default "light" mode    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743'; // use this for change background      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';      }    }    return (      <>        <Navbar title="textutils" mode={mode} toggleMode={toggleMode} /> {/\* 2. props sent \*/}        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <div className="container my-3">          <TextForm heading="Enter the text for analyze below" mode={mode} /> {/\* 6. props sent \*/}          {/\* <About /> \*/}        </div>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}> {/\* 3. props receive \*/}              <div className="container-fluid">                  <a className="navbar-brand" href="/">{props.title}</a>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <a className="nav-link active" aria-current="page" href="/">Home</a>                          </li>                          <li className="nav-item">                              <a className="nav-link" href="/">{props.aboutText}</a>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                      <div className={`form-check form-switch text-${props.mode==='light'?'dark':'light'}`}>                          {/\* 5. props receive, Using ternary operator \*/}                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" /> {/\* 4. props receive \*/}                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {      const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();   // 3. "text" convert to Uppercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleLoClick = () => {          // console.log("Lowercase was clicked :" + text);          let newText = text.toLowerCase();   // 3. "text" convert to Lowercase and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleClearClick = () => {          // console.log("Clear was clicked :" + text);          let newText = "";   // 3. "text" convert to Clear and append to "newText".          setText(newText)    // 4. "newText" goes inside the setText() function.      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);    // 5. now "setText(event.target.value)" change the value to uppercase in textarea.      }      const handleCopy = () => {          // console.log("i am copy");          var text = document.getElementById("myBox");          text.select();          text.setSelectionRange(0, 9999);          navigator.clipboard.writeText(text.value);  // 6. now "navigator.clipboard.writeText(text.value)" copy the textarea value.      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  {/\* 8. style={{color: props.mode==='dark'?'white':'#042743'}} \*/}                  <h1>{props.heading}</h1>    {/\* 2 \*/}                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} style={{ backgroundColor: props.mode === 'light'?'white':'grey', color: props.mode === 'dark'?'white':'#042743' }} id="myBox" rows="10"></textarea>                      {/\* 1. "text" is written here. (Inside "text" value sent), "style={{}}" meaning of it is Creating object inside the JavaScript. 7. {{backgroundColor: props.mode==='light'?'white':'grey'}}} \*/}                  </div>                  <button className="btn btn-primary mx-1" onClick={handleUpClick}>Convert to Uppercase</button>                  {/\* 2. When we click on "handleUpClick". Is gone to "handleUpClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleLoClick}>Convert to Lowercase</button>                  {/\* 2. When we click on "handleLoClick". Is gone to "handleLoClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleClearClick}>Clear Text</button>                  {/\* 2. When we click on "handleClearClick". Is gone to "handleClearClick" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleCopy}>Copy Text</button>                  {/\* 2. When we click on "handleCopy". Is gone to "handleCopy" function \*/}                  <button className="btn btn-primary mx-1" onClick={handleExtraSpaces}>Extra Spaces</button>                  {/\* 2. When we click on "handleExtraSpaces". Is gone to "handleExtraSpaces" function \*/}              </div>              <div className="container my-3" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  {/\* 8. style={{color: props.mode==='dark'?'white':'#042743'}} \*/}                  <h1>Your text summary</h1>                  <p>{text.split(" ").length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text.length>0?text:"Enter something in the textbox above to preview it here."}</p>              </div>          </>      );  } |

|  |  |
| --- | --- |
| **13. Adding + Auto Dismissing Alert Messages** | **Date: 20/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  // import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Navbar title="textutils" mode={mode} toggleMode={toggleMode} />        <Alert alert={alert} />        <div className="container my-3">          <TextForm showAlert={showAlert} heading="Enter the text for analyze below" mode={mode} />          {/\* <About /> \*/}        </div>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Alert.js  import React from 'react'  function Alert(props) {      const capitalize = (word) => {          const lower = word.toLowerCase();          return lower.charAt(0).toUpperCase() + lower.slice(1);      }      return (          <div>              { props.alert && (<div className={`alert alert-${props.alert.type} alert-dismissible fade show`} role="alert">                  <strong>{capitalize(props.alert.type)}</strong>: {props.alert.msg}              </div>              )}          </div>      );  }  export default Alert |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {      const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();          setText(newText)          props.showAlert("Converted to uppercase!", "success");      }      const handleLoClick = () => {          // console.log("Lowercase was clicked :" + text);          let newText = text.toLowerCase();          setText(newText)          props.showAlert("Converted to lowercase!", "success");      }      const handleClearClick = () => {          // console.log("Clear was clicked :" + text);          let newText = "";          setText(newText)          props.showAlert("Text cleared!", "success");      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);      }      const handleCopy = () => {          // console.log("i am copy");          var text = document.getElementById("myBox");          text.select();          text.setSelectionRange(0, 9999);          navigator.clipboard.writeText(text.value);          props.showAlert("Copied to clipboard!", "success");      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));          props.showAlert("Extra Spaces removed!", "success");      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h1>{props.heading}</h1>                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} style={{ backgroundColor: props.mode === 'light'?'white':'grey', color: props.mode === 'dark'?'white':'#042743' }} id="myBox" rows="10"></textarea>                  </div>                  <button className="btn btn-primary mx-1" onClick={handleUpClick}>Convert to Uppercase</button>                  <button className="btn btn-primary mx-1" onClick={handleLoClick}>Convert to Lowercase</button>                  <button className="btn btn-primary mx-1" onClick={handleClearClick}>Clear Text</button>                  <button className="btn btn-primary mx-1" onClick={handleCopy}>Copy Text</button>                  <button className="btn btn-primary mx-1" onClick={handleExtraSpaces}>Extra Spaces</button>              </div>              <div className="container my-3" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h1>Your text summary</h1>                  <p>{text.split(" ").length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text.length>0?text:"Enter something in the textbox above to preview it here."}</p>              </div>          </>      );  } |

|  |  |
| --- | --- |
| **14. Exercise 2: Adding Custom Color Theme to TextUtils**  **15. Changing title dynamically & adding favicons to TextUtils** | **Date: 20/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description" content="Textutils is a utility which can be used to manipulate your text in the way you want." />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>Textutils - Home</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  // import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");        document.title = 'TextUtils - Dark Mode';        // setInterval(() => {        //   document.title = 'TextUtils is Amazing Mode';        // }, 2000);        // setInterval(() => {        //   document.title = 'Install TextUtils Now';        // }, 1500);      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Navbar title="textutils" mode={mode} toggleMode={toggleMode} />        <Alert alert={alert} />        <div className="container my-3">          <TextForm showAlert={showAlert} heading="Enter the text for analyze below" mode={mode} />          {/\* <About /> \*/}        </div>      </>    );  }  export default App; |

|  |  |
| --- | --- |
| **16. React Router Setup + Usage** | **Date: 20/12/24** |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> npm install react-router-dom |

|  |
| --- |
| **Important Topic:**   * props * useState * Router |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  import {    BrowserRouter as Router,    Routes,    Route,    Link  } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");        document.title = 'TextUtils - Dark Mode';        // setInterval(() => {        //   document.title = 'TextUtils is Amazing Mode';        // }, 2000);        // setInterval(() => {        //   document.title = 'Install TextUtils Now';        // }, 1500);      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Router>          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">            <Routes>              {/\*              use "exact", Otherwise it match first component and show and we couldn't find why it do.              1. /users --> Component 1              2. /users/home --> Component 2               \*/}              <Route exact path="/about" element={<About />} />              <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Enter the text to analyze below" mode={mode} />} />            </Routes>          </div>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <Link className="navbar-brand" to="/">{props.title}</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link active" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">{props.aboutText}</Link>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                      <div className={`form-check form-switch text-${props.mode==='light'?'dark':'light'}`}>                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description" content="Textutils is a utility which can be used to manipulate your text in the way you want." />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>Textutils - Home</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |  |
| --- | --- |
| **17. [Free] Building + Hosting React App For free on Github Pages** | **Date: 21/12/24** |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> npm run build  > textutils-ch6@0.1.0 build  > react-scripts build  Creating an optimized production build...  One of your dependencies, babel-preset-react-app, is importing the  "@babel/plugin-proposal-private-property-in-object" package without  declaring it in its dependencies. This is currently working because  "@babel/plugin-proposal-private-property-in-object" is already in your  node\_modules folder for unrelated reasons, but it may break at any time.  babel-preset-react-app is part of the create-react-app project, which  is not maintianed anymore. It is thus unlikely that this bug will  ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to  your devDependencies to work around this error. This will make this message  go away.  Compiled with warnings.  [eslint]  src\App.js  Line 11:3: 'Link' is defined but never used no-unused-vars  Search for the keywords to learn more about each warning.  To ignore, add // eslint-disable-next-line to the line before.  File sizes after gzip:  73.97 kB build\static\js\main.315c6f3b.js  1.78 kB build\static\js\453.0d72e233.chunk.js  337 B build\static\css\main.dc4d6ff2.css  The project was built assuming it is hosted at /.  You can control this with the homepage field in your package.json.  The build folder is ready to be deployed.  You may serve it with a static server:  npm install -g serve  serve -s build  Find out more about deployment here:  <https://cra.link/deployment> |

|  |
| --- |
| * **Remove Router (Route, switch, Link, to)** * **"homepage": "https://myusername.github.io/my-app", (put this in the top of “package.json”)**   {    "homepage": "https://Amandeep0026.github.io/my-app",    "name": "textutils-ch6",    "version": "0.1.0",    "private": true,    "dependencies": {      "@testing-library/jest-dom": "^5.17.0",      "@testing-library/react": "^13.4.0",      "@testing-library/user-event": "^13.5.0",      "react": "^19.0.0",      "react-dom": "^19.0.0",      "react-scripts": "5.0.1",      "web-vitals": "^2.1.4"    },    "scripts": {      "start": "react-scripts start",      "build": "react-scripts build",      "test": "react-scripts test",      "eject": "react-scripts eject"    },    "eslintConfig": {      "extends": [        "react-app",        "react-app/jest"      ]    },    "browserslist": {      "production": [        ">0.2%",        "not dead",        "not op\_mini all"      ],      "development": [        "last 1 chrome version",        "last 1 firefox version",        "last 1 safari version"      ]    },    "devDependencies": {      "ajv": "^7.2.4"    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  // import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <a className="navbar-brand" href="#">{props.title}</a>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <a className="nav-link active" aria-current="page" href="#">Home</a>                          </li>                          <li className="nav-item">                              <a className="nav-link" href="#">{props.aboutText}</a>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                      <div className={`form-check form-switch text-${props.mode === 'light' ? 'dark' : 'light'}`}>                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  // import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  // import {  //   BrowserRouter as Router,  //   Routes,  //   Route  // } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");        document.title = 'TextUtils - Dark Mode';        // setInterval(() => {        //   document.title = 'TextUtils is Amazing Mode';        // }, 2000);        // setInterval(() => {        //   document.title = 'Install TextUtils Now';        // }, 1500);      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        {/\* <Router> \*/}          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">            {/\* <Routes> \*/}              {/\*              use "exact", Otherwise it match first component and show and we couldn't find why it do.              1. /users --> Component 1              2. /users/home --> Component 2               \*/}              {/\* <Route exact path="/about" element={<About />} /> \*/}              {/\* <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Enter the text to analyze below" mode={mode} />} /> \*/}            {/\* </Routes> \*/}            <TextForm showAlert={showAlert} heading="Enter the text to analyze below" mode={mode} />          </div>        {/\* </Router> \*/}      </>    );  }  export default App; |

|  |
| --- |
| Push code to Github  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git init  Reinitialized existing Git repository in C:/wamp64/www/MERN\_STACK/Code\_With\_Harry/React\_Js/textutils/.git/  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git add .  warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'README.md', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'public/index.html', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'public/manifest.json', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'public/robots.txt', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'src/App.css', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'src/App.js', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'src/index.js', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'src/reportWebVitals.js', LF will be replaced by CRLF the next time Git touches it  warning: in the working copy of 'src/setupTests.js', LF will be replaced by CRLF the next time Git touches it  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git commit -am "initial commit"  [master e342808] initial commit  11 files changed, 491 insertions(+), 94 deletions(-)  create mode 100644 module1.mjs  create mode 100644 module2.mjs  create mode 100644 src/components/TextForm.js  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git remote add origin https://github.com/Amandeep0026/React-Project.giPS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git remote -v  origin https://github.com/Amandeep0026/React-Project.git (fetch)  origin https://github.com/Amandeep0026/React-Project.git (push)  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> git push origin master  Enumerating objects: 38, done.  Counting objects: 100% (38/38), done.  Delta compression using up to 4 threads  Compressing objects: 100% (37/37), done.  Writing objects: 100% (38/38), 183.99 KiB | 2.59 MiB/s, done.  Total 38 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)  remote: Resolving deltas: 100% (4/4), done.  remote:  remote: Create a pull request for 'master' on GitHub by visiting:  remote: https://github.com/Amandeep0026/React-Project/pull/new/master  remote:  To https://github.com/Amandeep0026/React-Project.git  \* [new branch] master -> master |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> npm install --save gh-pages  added 10 packages, and audited 1385 packages in 9s  271 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\package.json |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> npm run deploy  > textutils-ch6@0.1.0 predeploy  > npm run build  > textutils-ch6@0.1.0 build  > react-scripts build  Creating an optimized production build...  One of your dependencies, babel-preset-react-app, is importing the  "@babel/plugin-proposal-private-property-in-object" package without  declaring it in its dependencies. This is currently working because  "@babel/plugin-proposal-private-property-in-object" is already in your  node\_modules folder for unrelated reasons, but it may break at any time.  babel-preset-react-app is part of the create-react-app project, which  is not maintianed anymore. It is thus unlikely that this bug will  ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to  your devDependencies to work around this error. This will make this message  go away.  Compiled with warnings.  [eslint]  src\components\Navbar.js  Line 9:17: The href attribute requires a valid value to be accessible. Provide a valid, navigable address as the href value. If you cannot provide a valid href, but still need the element to resemble a link, use a button and change it with appropriate styles. Learn more: https://github.com/jsx-eslint/eslint-plugin-jsx-a11y/blob/HEAD/docs/rules/anchor-is-valid.md jsx-a11y/anchor-is-valid  Line 16:29: The href attribute requires a valid value to be accessible. Provide a valid, navigable address as the href value. If you cannot provide a valid href, but still need the element to resemble a link, use a button and change it with appropriate styles. Learn more: https://github.com/jsx-eslint/eslint-plugin-jsx-a11y/blob/HEAD/docs/rules/anchor-is-valid.md jsx-a11y/anchor-is-valid  Line 19:29: The href attribute requires a valid value to be accessible. Provide a valid, navigable address as the href value. If you cannot provide a valid href, but still need the element to resemble a link, use a button and change it with appropriate styles. Learn more: https://github.com/jsx-eslint/eslint-plugin-jsx-a11y/blob/HEAD/docs/rules/anchor-is-valid.md jsx-a11y/anchor-is-valid  Search for the keywords to learn more about each warning.  To ignore, add // eslint-disable-next-line to the line before.  File sizes after gzip:  60.09 kB (-13.88 kB) build\static\js\main.cb36e95e.js  1.78 kB build\static\js\453.0d72e233.chunk.js  337 B build\static\css\main.dc4d6ff2.css  The project was built assuming it is hosted at /React-Project/.  You can control this with the homepage field in your package.json.  The build folder is ready to be deployed.  Find out more about deployment here: |

|  |
| --- |
| <https://amandeep0026.github.io/React-Project/> “here your website is published” |

|  |  |
| --- | --- |
| **18. Purchasing a Domain + Hosting TextUtils on a VPS** | **Date: 21/12/24** |

|  |
| --- |
| * [**https://v5.reactrouter.com/web/guides/quick-start**](https://v5.reactrouter.com/web/guides/quick-start) **“React route”** * [**https://create-react-app.dev/docs/deployment/**](https://create-react-app.dev/docs/deployment/) **“React deployment”** |

|  |
| --- |
| Removing GitHub hosting code  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils> npm uninstall gh-pages  removed 10 packages, and audited 1375 packages in 6s  270 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\package.json  {    "name": "textutils-ch6",    "version": "0.1.0",    "private": true,    "dependencies": {      "@testing-library/jest-dom": "^5.17.0",      "@testing-library/react": "^13.4.0",      "@testing-library/user-event": "^13.5.0",      "gh-pages": "^6.2.0",      "react": "^19.0.0",      "react-dom": "^19.0.0",      "react-scripts": "5.0.1",      "web-vitals": "^2.1.4"    },    "scripts": {      "predeploy": "npm run build",      "deploy": "gh-pages -d build",      "start": "react-scripts start",      "build": "react-scripts build",      "test": "react-scripts test",      "eject": "react-scripts eject"    },    "eslintConfig": {      "extends": [        "react-app",        "react-app/jest"      ]    },    "browserslist": {      "production": [        ">0.2%",        "not dead",        "not op\_mini all"      ],      "development": [        "last 1 chrome version",        "last 1 firefox version",        "last 1 safari version"      ]    },    "devDependencies": {      "ajv": "^7.2.4"    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");        document.title = 'TextUtils - Dark Mode';        // setInterval(() => {        //   document.title = 'TextUtils is Amazing Mode';        // }, 2000);        // setInterval(() => {        //   document.title = 'Install TextUtils Now';        // }, 1500);      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Router>          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">              <Routes>              {/\*              use "exact", Otherwise it match first component and show and we couldn't find why it do.              1. /users --> Component 1              2. /users/home --> Component 2               \*/}              <Route exact path="/about" element={<About />} />              <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Enter the text to analyze below" mode={mode} />} />            </Routes>          </div>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <Link className="navbar-brand" to="/">{props.title}</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link active" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">{props.aboutText}</Link>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                      <div className={`form-check form-switch text-${props.mode === 'light' ? 'dark' : 'light'}`}>                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |  |
| --- | --- |
| **19. Fixing Issues & Wrapping up TextUtils** | **Date: 21/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\public\index.html  <!DOCTYPE html>  <html lang="en">  <head>    <meta charset="utf-8" />    <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />    <meta name="viewport" content="width=device-width, initial-scale=1" />    <meta name="theme-color" content="#000000" />    <meta name="description"    content="Textutils is a word counter and a  character counting utility which can be used to manipulate your text in the way you want. You can remove extra spaces, copy the manipulated text as well as convert your text from Uppercase to lowercase and lowercase to Uppercase" />    <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />    <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"      integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    <title>Textutils - Word counter | character counter | lowercase to uppercase | uppercase to lowercase</title>  </head>  <body>    <noscript>You need to enable JavaScript to run this app.</noscript>    <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"      integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz"      crossorigin="anonymous"></script>  </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");        // document.title = 'TextUtils - Dark Mode';        // setInterval(() => {        //   document.title = 'TextUtils is Amazing Mode';        // }, 2000);        // setInterval(() => {        //   document.title = 'Install TextUtils Now';        // }, 1500);      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        // document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Router>          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">            <Routes>              {/\*              use "exact", Otherwise it match first component and show and we couldn't find why it do.              1. /users --> Component 1              2. /users/home --> Component 2               \*/}              <Route exact path="/about" element={<About mode={mode} />} />              <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Try Textutils - Word Counter, Character Counter, Remove extra spaces" mode={mode} />} />            </Routes>          </div>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {        const handleUpClick = () => {          // console.log("Uppercase was clicked :" + text);          let newText = text.toUpperCase();          setText(newText)          props.showAlert("Converted to uppercase!", "success");      }      const handleLoClick = () => {          // console.log("Lowercase was clicked :" + text);          let newText = text.toLowerCase();          setText(newText)          props.showAlert("Converted to lowercase!", "success");      }      const handleClearClick = () => {          // console.log("Clear was clicked :" + text);          let newText = "";          setText(newText)          props.showAlert("Text cleared!", "success");      }      const handleOnChange = (event) => {          // console.log("On change");          setText(event.target.value);      }      const handleCopy = () => {          // console.log("i am copy");          var text = document.getElementById("myBox");          text.select();          navigator.clipboard.writeText(text.value);          document.getSelection().removeAllRanges();          props.showAlert("Copied to clipboard!", "success");      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));          props.showAlert("Extra Spaces removed!", "success");      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h1 className='mb-4'>{props.heading}</h1>                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} style={{ backgroundColor: props.mode === 'dark'?'#13466e':'white', color: props.mode === 'dark'?'white':'#042743' }} id="myBox" rows="10"></textarea>                  </div>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleUpClick}>Convert to Uppercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleLoClick}>Convert to Lowercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleClearClick}>Clear Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleCopy}>Copy Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleExtraSpaces}>Extra Spaces</button>              </div>              <div className="container my-3" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h2>Your text summary</h2>                  <p>{text.split(" ").filter((element)=>{return element.length!==0}).length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").filter((element)=>{return element.length!==0}).length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text.length>0?text:"Nothing to preview!"}</p>              </div>          </>      );  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <Link className="navbar-brand" to="/">{props.title}</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link active" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">{props.aboutText}</Link>                          </li>                      </ul>                      {/\* <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-primary" type="submit">Search</button>                      </form> \*/}                      <div className={`form-check form-switch text-${props.mode === 'light' ? 'dark' : 'light'}`}>                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  // Navbar.defaultProps = {  //     title: 'Set title here',  //     aboutText: 'About text here'  // }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Alert.js  import React from 'react'  function Alert(props) {      const capitalize = (word) => {          const lower = word.toLowerCase();          return lower.charAt(0).toUpperCase() + lower.slice(1);      }      return (          <div style={{height: '50px'}}>              {props.alert && (<div className={`alert alert-${props.alert.type}              alert-dismissible fade show`} role="alert">                  <strong>{capitalize(props.alert.type)}</strong>: {props.alert.msg}              </div>              )}          </div>      );  }  export default Alert |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\About.js  import React, { useState } from 'react'  export default function About(props) {      // const [myStyle, setMyStyle] = useState({      //     color: 'black',      //     backgroundColor: 'white',      //     border: '2px solid white'      // })      let myStyle = {          color: props.mode === 'dark'?'white':'#042743',          backgroundColor: props.mode === 'dark'?'rgb(36 74 104)':'white',      }      // const [btnText, setBtnText] = useState("Enable Dark mode")      // const toggleStyle = () => {      //     if (myStyle.color === 'black') {      //         setMyStyle({      //             color: 'white',      //             backgroundColor: 'black',      //             border: '2px solid white'      //         })      //         setBtnText("Enable Light mode")      //     }      //     else {      //         setMyStyle({      //             color: 'black',      //             backgroundColor: 'white'      //         })      //         setBtnText("Enable Dark mode")      //     }      // }      return (          <div className='container' style={{color: props.mode === 'dark'?'white':'#042743'}}>              <h1 className='my-3'>About Us</h1>              <div className="accordion" id="accordionExample" style={myStyle}>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button" type="button" style={myStyle} data-bs-toggle="collapse" data-bs-target="#collapseOne" aria-expanded="true" aria-controls="collapseOne">                              <strong>Analyze Your Text</strong>                          </button>                      </h2>                      <div id="collapseOne" className="accordion-collapse collapse show" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              Textutils gives you a way to analyze your text quickly and efficiently. Be it word count, character count or.                          </div>                      </div>                  </div>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button collapsed" style={myStyle} type="button" data-bs-toggle="collapse" data-bs-target="#collapseTwo" aria-expanded="false" aria-controls="collapseTwo">                              <strong>Free to use</strong>                          </button>                      </h2>                      <div id="collapseTwo" className="accordion-collapse collapse" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              TextUtils is a free character counter tool that provides instant character count & word count statistics for a given text. TextUtils reports the number of words and characters. Thus it is suitable for writing text with word/ character limit.                          </div>                      </div>                  </div>                  <div className="accordion-item">                      <h2 className="accordion-header">                          <button className="accordion-button collapsed" style={myStyle} type="button" data-bs-toggle="collapse" data-bs-target="#collapseThree" aria-expanded="false" aria-controls="collapseThree">                              <strong>Browser Compatible</strong>                          </button>                      </h2>                      <div id="collapseThree" className="accordion-collapse collapse" data-bs-parent="#accordionExample">                          <div className="accordion-body" style={myStyle}>                              This word counter software works in any web browsers such as Chrome, Firefox, Internet Explorer, Safari, Opera. It suits to count characters in facebook, blog, books, excel document, pdf document, essays, etc.                          </div>                      </div>                  </div>              </div>              {/\* <div className="container my-3">                  <button onClick={toggleStyle} type='button' className='btn btn-primary my-3'>{btnText}</button>              </div> \*/}          </div>      )  } |

|  |  |
| --- | --- |
| **20. Exercise 2: Solutions + Shoutouts** | **Date: 23/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const removeBodyClasses = ()=>{      document.body.classList.remove('bg-light')      document.body.classList.remove('bg-dark')      document.body.classList.remove('bg-warning')      document.body.classList.remove('bg-danger')      document.body.classList.remove('bg-success')      document.body.classList.remove('bg-primary')    }    const toggleMode = (cls) => {      removeBodyClasses();      console.log(cls)      document.body.classList.add('bg-'+cls)      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");        // document.title = 'TextUtils - Light Mode';      }    }    return (      <>        {/\* <Navbar title="textutils" aboutText="About Us" /> \*/}        {/\*<Navbar />\*/}        {/\* Navbar.defaultProps functionality is remove in react 18v \*/}        <Router>          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">            <Routes>              {/\*              use "exact", Otherwise it match first component and show and we couldn't find why it do.              1. /users --> Component 1              2. /users/home --> Component 2               \*/}              <Route exact path="/about" element={<About mode={mode} />} />              <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Try Textutils - Word Counter, Character Counter, Remove extra spaces" mode={mode} />} />            </Routes>          </div>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {        const handleUpClick = () => {          let newText = text.toUpperCase();          setText(newText)          props.showAlert("Converted to uppercase!", "success");      }      const handleLoClick = () => {          let newText = text.toLowerCase();          setText(newText)          props.showAlert("Converted to lowercase!", "success");      }      const handleClearClick = () => {          let newText = "";          setText(newText)          props.showAlert("Text cleared!", "success");      }      const handleOnChange = (event) => {          setText(event.target.value);      }      const handleCopy = () => {          var text = document.getElementById("myBox");          text.select();          navigator.clipboard.writeText(text.value);          document.getSelection().removeAllRanges();          props.showAlert("Copied to clipboard!", "success");      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));          props.showAlert("Extra Spaces removed!", "success");      }      const [text, setText] = useState('');      return (          <>              <div className="container" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h1 className='mb-4'>{props.heading}</h1>                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} style={{ backgroundColor: props.mode === 'dark'?'#13466e':'white', color: props.mode === 'dark'?'white':'#042743' }} id="myBox" rows="10"></textarea>                  </div>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleUpClick}>Convert to Uppercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleLoClick}>Convert to Lowercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleClearClick}>Clear Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleCopy}>Copy Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleExtraSpaces}>Extra Spaces</button>              </div>              <div className="container my-3" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h2>Your text summary</h2>                  <p>{text.split(" ").filter((element)=>{return element.length!==0}).length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").filter((element)=>{return element.length!==0}).length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text.length>0?text:"Nothing to preview!"}</p>              </div>          </>      );  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <Link className="navbar-brand" to="/">{props.title}</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link active" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">{props.aboutText}</Link>                          </li>                      </ul>                      <div className="d-flex">                          <div className="bg-primary rounded mx-2" onClick={() => { props.toggleMode('primary') }} style={{ height: '30px', width: '30px', cursor: 'pointer' }}></div>                          <div className="bg-danger rounded mx-2" onClick={() => { props.toggleMode('danger') }} style={{ height: '30px', width: '30px', cursor: 'pointer' }}></div>                          <div className="bg-success rounded mx-2" onClick={() => { props.toggleMode('success') }} style={{ height: '30px', width: '30px', cursor: 'pointer' }}></div>                          <div className="bg-warning rounded mx-2" onClick={() => { props.toggleMode('warning') }} style={{ height: '30px', width: '30px', cursor: 'pointer' }}></div>                          <div className="bg-light rounded mx-2" onClick={() => { props.toggleMode('light') }} style={{ height: '30px', width: '30px', cursor: 'pointer', border: '2px solid black' }}></div>                          <div className="bg-dark rounded mx-2" onClick={() => { props.toggleMode('dark') }} style={{ height: '30px', width: '30px', cursor: 'pointer', border: '2px solid white' }}></div>                      </div>                      {/\* <div className={`form-check form-switch text-${props.mode === 'light' ? 'dark' : 'light'}`}>                          <input className="form-check-input" onClick={() => { props.toggleMode(null) }} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Toggle Mode</label>                      </div> \*/}                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  } |

|  |  |
| --- | --- |
| **21. Fixing few more Textutils Issues** | **Date: 23/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\App.js  import './App.css';  import About from './components/About';  import Navbar from './components/Navbar';  import TextForm from './components/TextForm';  import Alert from './components/Alert';  import React, { useState } from 'react';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  function App() {    const [mode, setMode] = useState('light');    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    const toggleMode = () => {      if (mode === 'light') {        setMode('dark');        document.body.style.backgroundColor = '#042743';        showAlert("Dark mode has been enabled", "success");      }      else {        setMode('light');        document.body.style.backgroundColor = 'white';        showAlert("Light mode has been enabled", "success");      }    }    return (      <>        <Router>          <Navbar title="textutils" mode={mode} toggleMode={toggleMode} aboutText="About Us" />          <Alert alert={alert} />          <div className="container my-3">            <Routes>              <Route exact path="/about" element={<About mode={mode} />} />              <Route exact path="/" element={<TextForm showAlert={showAlert} heading="Try Textutils - Word Counter, Character Counter, Remove extra spaces" mode={mode} />} />            </Routes>          </div>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\TextForm.js  import React, { useState } from 'react';  export default function TextForm(props) {        const handleUpClick = () => {          let newText = text.toUpperCase();          setText(newText)          props.showAlert("Converted to uppercase!", "success");      }      const handleLoClick = () => {          let newText = text.toLowerCase();          setText(newText)          props.showAlert("Converted to lowercase!", "success");      }      const handleClearClick = () => {          let newText = "";          setText(newText)          props.showAlert("Text cleared!", "success");      }      const handleOnChange = (event) => {          setText(event.target.value);      }      const handleCopy = () => {          navigator.clipboard.writeText(text);          props.showAlert("Copied to clipboard!", "success");      }      const handleExtraSpaces = () => {          let newText = text.split(/[ ]+/);          setText(newText.join(" "));          props.showAlert("Extra Spaces removed!", "success");      }      const [text, setText] = useState('');    // useState function      //text = "new text"; //Wrong way to change the state      //setText("new text"); //Correct way to change the state      return (          <>              <div className="container" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h1 className='mb-4'>{props.heading}</h1>                  <div className="mb-3">                      <textarea className="form-control" value={text} onChange={handleOnChange} style={{ backgroundColor: props.mode === 'dark'?'#13466e':'white', color: props.mode === 'dark'?'white':'#042743' }} id="myBox" rows="10"></textarea>                  </div>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleUpClick}>Convert to Uppercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleLoClick}>Convert to Lowercase</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleClearClick}>Clear Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleCopy}>Copy Text</button>                  <button disabled={text.length===0} className="btn btn-primary mx-1 my-1" onClick={handleExtraSpaces}>Extra Spaces</button>              </div>              <div className="container my-3" style={{ color: props.mode === 'dark' ? 'white' : '#042743' }}>                  <h2>Your text summary</h2>                  <p>{text.split(/\s+/).filter((element)=>{return element.length!==0}).length} words and {text.length} characters</p>                  <p>{0.008 \* text.split(" ").filter((element)=>{return element.length!==0}).length} Minutes read</p>                  <h2>Preview</h2>                  <p>{text.length>0?text:"Nothing to preview!"}</p>              </div>          </>      );  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\components\Navbar.js  import React from 'react';  import PropTypes from 'prop-types';  import { Link } from 'react-router-dom';  export default function Navbar(props) {      return (          <nav className={`navbar navbar-expand-lg navbar-${props.mode} bg-${props.mode}`}>              <div className="container-fluid">                  <Link className="navbar-brand" to="/">{props.title}</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">{props.aboutText}</Link>                          </li>                      </ul>                      <div className={`form-check form-switch text-${props.mode === 'light' ? 'dark' : 'light'}`}>                          <input className="form-check-input" onClick={props.toggleMode} type="checkbox" role="switch" id="flexSwitchCheckDefault" />                          <label className="form-check-label" htmlFor="flexSwitchCheckDefault">Enable DarkMode</label>                      </div>                  </div>              </div>          </nav>      );  }  Navbar.propTypes = {      title: PropTypes.string.isRequired,        //PropTypes.string using because we are pass string only      aboutText: PropTypes.string.isRequired,     //PropTypes.string using because we are pass string only  }  // This part is not working, this functionality is removed by react 18v  Navbar.defaultProps = {      title: 'Set title here',      aboutText: 'About text here'  }; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\textutils\src\index.css  .accordion-button::after {      filter: invert(1);  } |

|  |  |
| --- | --- |
| **22. Project 2 Setup + Introduction to Class based components** | **Date: 23/12/24** |

**What is the difference between class base component and function base component?**

**Syntax**:

* **Class**: Uses class syntax and requires a render() method.
* **Function**: Uses a simple function to return JSX.

**State Management**:

* **Class**: Uses this.state and this.setState().
* **Function**: Uses the useState hook.

**Lifecycle**:

* **Class**: Uses lifecycle methods like componentDidMount.
* **Function**: Uses the useEffect hook.

**Complexity**:

* **Class**: More boilerplate needs this.
* **Function**: Cleaner and simpler.

Function-based components are now the standard in React development due to their simplicity, performance, and support for hooks. Class-based components are still used in some legacy applications but are becoming less common in modern React codebases.

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  export default class App extends Component {    c = 'John';    render() {      return (        <div>          Hello my first class based Component {this.c}        </div>      )    }  } |

|  |  |
| --- | --- |
| **23. Component Structure of our NewsMonkey React App** | **Date: 25/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\public\index.html  <!DOCTYPE html>  <html lang="en">    <head>      <meta charset="utf-8" />      <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />      <meta name="viewport" content="width=device-width, initial-scale=1" />      <meta name="theme-color" content="#000000" />      <meta        name="description"        content="NewsMonkey is a news app which can be used to grab quick daily news bites. If you are interested in news, weather, politics and sports news, NewsMonkey is for you!"      />      <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />      <!--        manifest.json provides metadata used when your web app is installed on a        user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/      -->      <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">      <!--        Notice the use of %PUBLIC\_URL% in the tags above.        It will be replaced with the URL of the `public` folder during the build.        Only files inside the `public` folder can be referenced from the HTML.        Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC\_URL%/favicon.ico" will        work correctly both with client-side routing and a non-root public URL.        Learn how to configure a non-root public URL by running `npm run build`.      -->      <title>NewsMonkey - Get your daily dose of news for free!</title>    </head>    <body>      <noscript>You need to enable JavaScript to run this app.</noscript>      <div id="root"></div>      <!--        This HTML file is a template.        If you open it directly in the browser, you will see an empty page.        You can add webfonts, meta tags, or analytics to this file.        The build step will place the bundled scripts into the <body> tag.        To begin the development, run `npm start` or `yarn start`.        To create a production bundle, use `npm run build` or `yarn build`.      -->      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>    </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react'  import { Link } from 'react-router-dom';  export class NavBar extends Component {      render() {          return (              <div>                  <nav className="navbar navbar-expand-lg bg-body-tertiary">                      <div className="container-fluid">                          <a className="navbar-brand" href="/">NewsMonkey</a>                          <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                              <span className="navbar-toggler-icon"></span>                          </button>                          <div className="collapse navbar-collapse" id="navbarSupportedContent">                              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                                  <li className="nav-item">                                      <a className="nav-link active" aria-current="page" href="/">Home</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" href="/about">About</a>                                  </li>                                </ul>                          </div>                      </div>                  </nav>              </div>          )      }  }  export default NavBar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  export default class App extends Component {    render() {      return (        <div>          <NavBar />          <News />        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {    render() {      return (        <div>          this is a news component          <NewsItem />          <NewsItem />          <NewsItem />          <NewsItem />          <NewsItem />        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {  render() {  return (  <div>  This is a news item  </div>  )  }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **24. Fetching API Key from News API** | **Date: 25/12/24** |

Use the NewsAPI API and Json file too.

Take news API URL from here (customize it).

A screenshot of a computer

Description automatically generated

<https://newsapi.org/v2/top-headlines?apiKey=ecb6b24c8cd54f3fb15590e183275f0b&q=cricket> (Like this)

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\simpleOutput.json  {      "status": "ok",      "totalResults": 3,      "articles": [        {          "source": {            "id": "bbc-sport",            "name": "BBC Sport"          },          "author": null,          "title": "ICC Champions Trophy 2025: India face Pakistan in Dubai as fixtures released",          "description": "India will face Pakistan in Dubai on 23 February as the International Cricket Council releases the Champions Trophy 2025 fixtures.",          "url": "http://www.bbc.co.uk/sport/cricket/articles/cd75z2xxzpwo",          "urlToImage": "https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg",          "publishedAt": "2024-12-24T14:37:20.6320709Z",          "content": "February\r\n19 Pakistan v New Zealand, Karachi\r\n20 Bangladesh v India, Dubai\r\n21 Afghanistan v South Africa, Karachi\r\n22 Australia v England, Lahore, Pakistan\r\n23 Pakistan v India, Dubai\r\n24 Bangladesh… [+412 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "PCB hands Umar Akmal three-year ban from all cricket | ESPNcricinfo.com",          "description": "Penalty after the batsman pleaded guilty to not reporting corrupt approaches | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/29103103/pcb-hands-umar-akmal-three-year-ban-all-cricket",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1099495\_800x450.jpg",          "publishedAt": "2020-04-27T11:41:47Z",          "content": "Umar Akmal's troubled cricket career has hit its biggest roadblock yet, with the PCB handing him a ban from all representative cricket for three years after he pleaded guilty of failing to report det… [+1506 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "What we learned from watching the 1992 World Cup final in full again | ESPNcricinfo.com",          "description": "Wides, lbw calls, swing - plenty of things were different in white-ball cricket back then | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/28970907/learned-watching-1992-world-cup-final-full-again",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1219926\_1296x729.jpg",          "publishedAt": "2020-03-30T15:26:05Z",          "content": "Last week, we at ESPNcricinfo did something we have been thinking of doing for eight years now: pretend-live ball-by-ball commentary for a classic cricket match. We knew the result, yes, but we tried… [+6823 chars]"        }      ]    } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let {title, description} = this.props;      return (        <div>          <div className="card" style={{width: "18rem"}} >            <img src="..." className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a href="/newsdetail" className="btn btn-primary">Go somewhere</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {    render() {      return (        <div className="container my-3">          <h2>NewsMonkey - Top Headlines</h2>          <div className="row">            <div className="col-md-4">              <NewsItem title="myTitle" description="myDesc" />            </div>            <div className="col-md-4">              <NewsItem title="myTitle" description="myDesc" />            </div>            <div className="col-md-4">              <NewsItem title="myTitle" description="myDesc" />            </div>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |  |
| --- | --- |
| **25. Understanding state in class-based components** | **Date: 26/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card" style={{width: "18rem"}} >            <img src={imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a href="/newsdetail/newsUrl" className="btn btn-sm btn-primary">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {    articles = [        {          "source": {            "id": "bbc-sport",            "name": "BBC Sport"          },          "author": null,          "title": "ICC Champions Trophy 2025: India face Pakistan in Dubai as fixtures released",          "description": "India will face Pakistan in Dubai on 23 February as the International Cricket Council releases the Champions Trophy 2025 fixtures.",          "url": "http://www.bbc.co.uk/sport/cricket/articles/cd75z2xxzpwo",          "urlToImage": "https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg",          "publishedAt": "2024-12-24T14:37:20.6320709Z",          "content": "February\r\n19 Pakistan v New Zealand, Karachi\r\n20 Bangladesh v India, Dubai\r\n21 Afghanistan v South Africa, Karachi\r\n22 Australia v England, Lahore, Pakistan\r\n23 Pakistan v India, Dubai\r\n24 Bangladesh… [+412 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "PCB hands Umar Akmal three-year ban from all cricket | ESPNcricinfo.com",          "description": "Penalty after the batsman pleaded guilty to not reporting corrupt approaches | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/29103103/pcb-hands-umar-akmal-three-year-ban-all-cricket",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1099495\_800x450.jpg",          "publishedAt": "2020-04-27T11:41:47Z",          "content": "Umar Akmal's troubled cricket career has hit its biggest roadblock yet, with the PCB handing him a ban from all representative cricket for three years after he pleaded guilty of failing to report det… [+1506 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "What we learned from watching the 1992 World Cup final in full again | ESPNcricinfo.com",          "description": "Wides, lbw calls, swing - plenty of things were different in white-ball cricket back then | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/28970907/learned-watching-1992-world-cup-final-full-again",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1219926\_1296x729.jpg",          "publishedAt": "2020-03-30T15:26:05Z",          "content": "Last week, we at ESPNcricinfo did something we have been thinking of doing for eight years now: pretend-live ball-by-ball commentary for a classic cricket match. We knew the result, yes, but we tried… [+6823 chars]"        }      ]    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: this.articles,        loading: false      }    }    render() {      return (        <div className="container my-3">          <h2>NewsMonkey - Top Headlines</h2>          <div className="row">            <div className="col-md-4">              <NewsItem title="myTitle" description="mydesc" imageUrl="https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg" newsUrl="TODO" />            </div>            <div className="col-md-4">              <NewsItem title="myTitle" description="mydesc"  />            </div>            <div className="col-md-4">              <NewsItem title="myTitle" description="mydesc"  />            </div>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\simpleOutput.json  {      "status": "ok",      "totalResults": 3,      "articles": [        {          "source": {            "id": "bbc-sport",            "name": "BBC Sport"          },          "author": null,          "title": "ICC Champions Trophy 2025: India face Pakistan in Dubai as fixtures released",          "description": "India will face Pakistan in Dubai on 23 February as the International Cricket Council releases the Champions Trophy 2025 fixtures.",          "url": "http://www.bbc.co.uk/sport/cricket/articles/cd75z2xxzpwo",          "urlToImage": "https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg",          "publishedAt": "2024-12-24T14:37:20.6320709Z",          "content": "February\r\n19 Pakistan v New Zealand, Karachi\r\n20 Bangladesh v India, Dubai\r\n21 Afghanistan v South Africa, Karachi\r\n22 Australia v England, Lahore, Pakistan\r\n23 Pakistan v India, Dubai\r\n24 Bangladesh… [+412 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "PCB hands Umar Akmal three-year ban from all cricket | ESPNcricinfo.com",          "description": "Penalty after the batsman pleaded guilty to not reporting corrupt approaches | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/29103103/pcb-hands-umar-akmal-three-year-ban-all-cricket",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1099495\_800x450.jpg",          "publishedAt": "2020-04-27T11:41:47Z",          "content": "Umar Akmal's troubled cricket career has hit its biggest roadblock yet, with the PCB handing him a ban from all representative cricket for three years after he pleaded guilty of failing to report det… [+1506 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "What we learned from watching the 1992 World Cup final in full again | ESPNcricinfo.com",          "description": "Wides, lbw calls, swing - plenty of things were different in white-ball cricket back then | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/28970907/learned-watching-1992-world-cup-final-full-again",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1219926\_1296x729.jpg",          "publishedAt": "2020-03-30T15:26:05Z",          "content": "Last week, we at ESPNcricinfo did something we have been thinking of doing for eight years now: pretend-live ball-by-ball commentary for a classic cricket match. We knew the result, yes, but we tried… [+6823 chars]"        }      ]    } |

|  |  |
| --- | --- |
| **26. Looping through an array in JSX to display NewsItems from state** | **Date: 26/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card" style={{width: "18rem"}} >            <img src={imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}...</h5>                <p className="card-text">{description}...</p>                <a href={newsUrl} target="\_blank" className="btn btn-sm btn-primary">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {    articles = [      {        "source": {          "id": "bbc-sport",          "name": "BBC Sport"        },        "author": null,        "title": "ICC Champions Trophy 2025: India face Pakistan in Dubai as fixtures released",        "description": "India will face Pakistan in Dubai on 23 February as the International Cricket Council releases the Champions Trophy 2025 fixtures.",        "url": "http://www.bbc.co.uk/sport/cricket/articles/cd75z2xxzpwo",        "urlToImage": "https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg",        "publishedAt": "2024-12-24T14:37:20.6320709Z",        "content": "February\r\n19 Pakistan v New Zealand, Karachi\r\n20 Bangladesh v India, Dubai\r\n21 Afghanistan v South Africa, Karachi\r\n22 Australia v England, Lahore, Pakistan\r\n23 Pakistan v India, Dubai\r\n24 Bangladesh… [+412 chars]"      },      {        "source": {          "id": "espn-cric-info",          "name": "ESPN Cric Info"        },        "author": null,        "title": "PCB hands Umar Akmal three-year ban from all cricket | ESPNcricinfo.com",        "description": "Penalty after the batsman pleaded guilty to not reporting corrupt approaches | ESPNcricinfo.com",        "url": "http://www.espncricinfo.com/story/\_/id/29103103/pcb-hands-umar-akmal-three-year-ban-all-cricket",        "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1099495\_800x450.jpg",        "publishedAt": "2020-04-27T11:41:47Z",        "content": "Umar Akmal's troubled cricket career has hit its biggest roadblock yet, with the PCB handing him a ban from all representative cricket for three years after he pleaded guilty of failing to report det… [+1506 chars]"      },      {        "source": {          "id": "espn-cric-info",          "name": "ESPN Cric Info"        },        "author": null,        "title": "What we learned from watching the 1992 World Cup final in full again | ESPNcricinfo.com",        "description": "Wides, lbw calls, swing - plenty of things were different in white-ball cricket back then | ESPNcricinfo.com",        "url": "http://www.espncricinfo.com/story/\_/id/28970907/learned-watching-1992-world-cup-final-full-again",        "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1219926\_1296x729.jpg",        "publishedAt": "2020-03-30T15:26:05Z",        "content": "Last week, we at ESPNcricinfo did something we have been thinking of doing for eight years now: pretend-live ball-by-ball commentary for a classic cricket match. We knew the result, yes, but we tried… [+6823 chars]"      }    ]    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: this.articles,        loading: false      }    }    render() {      return (        <div className="container my-3">          <h2>NewsMonkey - Top Headlines</h2>          <div className="row">            {this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title.slice(0, 45)} description={element.description.slice(0, 88)} imageUrl={element.urlToImage} newsUrl={element.url} />              </div>            })}          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\simpleOutput.json  {      "status": "ok",      "totalResults": 3,      "articles": [        {          "source": {            "id": "bbc-sport",            "name": "BBC Sport"          },          "author": null,          "title": "ICC Champions Trophy 2025: India face Pakistan in Dubai as fixtures released",          "description": "India will face Pakistan in Dubai on 23 February as the International Cricket Council releases the Champions Trophy 2025 fixtures.",          "url": "http://www.bbc.co.uk/sport/cricket/articles/cd75z2xxzpwo",          "urlToImage": "https://ichef.bbci.co.uk/ace/branded\_sport/1200/cpsprodpb/41f7/live/fc4f0b20-c1fd-11ef-bf74-197c255fcbb1.jpg",          "publishedAt": "2024-12-24T14:37:20.6320709Z",          "content": "February\r\n19 Pakistan v New Zealand, Karachi\r\n20 Bangladesh v India, Dubai\r\n21 Afghanistan v South Africa, Karachi\r\n22 Australia v England, Lahore, Pakistan\r\n23 Pakistan v India, Dubai\r\n24 Bangladesh… [+412 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "PCB hands Umar Akmal three-year ban from all cricket | ESPNcricinfo.com",          "description": "Penalty after the batsman pleaded guilty to not reporting corrupt approaches | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/29103103/pcb-hands-umar-akmal-three-year-ban-all-cricket",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1099495\_800x450.jpg",          "publishedAt": "2020-04-27T11:41:47Z",          "content": "Umar Akmal's troubled cricket career has hit its biggest roadblock yet, with the PCB handing him a ban from all representative cricket for three years after he pleaded guilty of failing to report det… [+1506 chars]"        },        {          "source": {            "id": "espn-cric-info",            "name": "ESPN Cric Info"          },          "author": null,          "title": "What we learned from watching the 1992 World Cup final in full again | ESPNcricinfo.com",          "description": "Wides, lbw calls, swing - plenty of things were different in white-ball cricket back then | ESPNcricinfo.com",          "url": "http://www.espncricinfo.com/story/\_/id/28970907/learned-watching-1992-world-cup-final-full-again",          "urlToImage": "https://a4.espncdn.com/combiner/i?img=%2Fi%2Fcricket%2Fcricinfo%2F1219926\_1296x729.jpg",          "publishedAt": "2020-03-30T15:26:05Z",          "content": "Last week, we at ESPNcricinfo did something we have been thinking of doing for eight years now: pretend-live ball-by-ball commentary for a classic cricket match. We knew the result, yes, but we tried… [+6823 chars]"        }      ]    } |

|  |  |
| --- | --- |
| **27. Using Fetch API in React to populate NewsItems** | **Date: 26/12/24** |

A screenshot of a computer

Description automatically generated

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {      constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: [],        loading: false      }    }    async componentDidMount() {      console.log("cdm");      let url = "https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b";      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({articles: parsedData.articles})    }    render() {      console.log("render");      return (        <div className="container my-3">          <h1>NewsMonkey - Top Headlines</h1>          <div className="row">            {this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title?element.title:""} description={element.description?element.description:""} imageUrl={element.urlToImage} newsUrl={element.url} />              </div>            })}          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card" style={{width: "18rem"}} >            <img src={!imageUrl?"https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg":imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a href={newsUrl} target="\_blank" className="btn btn-sm btn-primary">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **28. Adding Previous & Next Buttons to populate NewsItems** | **Date: 28/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react';  import { Link } from 'react-router-dom';  export class NavBar extends Component {      render() {          return (              <div>                  <nav className="navbar navbar-expand-lg navbar-dark bg-dark">                      <div className="container-fluid">                          <a className="navbar-brand" href="/">NewsMonkey</a>                          <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                              <span className="navbar-toggler-icon"></span>                          </button>                          <div className="collapse navbar-collapse" id="navbarSupportedContent">                              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                                  <li className="nav-item">                                      <a className="nav-link active" aria-current="page" href="/">Home</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" href="/about">About</a>                                  </li>                                </ul>                          </div>                      </div>                  </nav>              </div>          )      }  }  export default NavBar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  export class News extends Component {    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: [],        loading: false,        page: 1      }    }    async componentDidMount() {      console.log("cdm");      let url = "https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=1&pageSize=20";      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({ articles: parsedData.articles, totalResults: parsedData.totalResults })    }    handlePrevClick = async () => {      console.log("Previous");      console.log("cdm");      let url = `https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page - 1}&pageSize=20`;      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({        page: this.state.page - 1,        articles: parsedData.articles      });    }    handleNextClick = async () => {      console.log("Next");      if (this.state.page + 1 > Math.ceil(this.state.totalResults / 20)) {      }      else {        console.log("cdm");        let url = `https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page + 1}&pageSize=20`;        let data = await fetch(url);        let parsedData = await data.json();        console.log(parsedData);        this.setState({          page: this.state.page + 1,          articles: parsedData.articles        });      }    }    render() {      console.log("render");      return (        <div className="container my-3">          <h1>NewsMonkey - Top Headlines</h1>          <div className="row">            {this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card" style={{width: "18rem"}} >            <img src={!imageUrl?"https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg":imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **29. Adding loading spinner & variable pageSize to NewsMonkey** | **Date: 28/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  export default class App extends Component {    render() {      return (        <div>          <NavBar />          <News pageSize={5} />        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  export class News extends Component {    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: [],        loading: false,        page: 1      }    }    async componentDidMount() {      console.log("cdm");      let url = `https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=1&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false })    }    handlePrevClick = async () => {      console.log("Previous");      let url = `https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page - 1}&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        page: this.state.page - 1,        articles: parsedData.articles,        loading: false      });    }    handleNextClick = async () => {      console.log("Next");      if (!(this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize))) {        console.log("cdm");        let url = `https://newsapi.org/v2/top-headlines?country=us&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page + 1}&pageSize=${this.props.pageSize}`;        this.setState({loading: true});        let data = await fetch(url);        let parsedData = await data.json();        this.setState({          page: this.state.page + 1,          articles: parsedData.articles,          loading: false        });      }    }    render() {      console.log("render");      return (        <div className="container my-3">          <h1 className='text-center'>NewsMonkey - Top Headlines</h1>          {this.state.loading && <Spinner/>}          <div className="row">            {!this.state.loading && this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card" style={{width: "18rem"}} >            <img src={!imageUrl?"https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg":imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\Spinner.js  import React, { Component } from 'react'  import loading from './loading.svg'  export class Spinner extends Component {    render() {      return (        <div className='text-center'>          <img src={loading} alt="loading" />        </div>      )    }  }  export default Spinner  // rce |

|  |  |
| --- | --- |
| **30. Adding Categories & propTypes to NewsMonkey React App** | **Date: 28/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  export default class App extends Component {    render() {      return (        <div>          <NavBar />          <News pageSize={5} country="us" category="science" />        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react';  import { Link } from 'react-router-dom';  export class NavBar extends Component {      render() {          return (              <div>                  <nav className="navbar navbar-expand-lg navbar-dark bg-dark">                      <div className="container-fluid">                          <a className="navbar-brand" href="/">NewsMonkey</a>                          <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                              <span className="navbar-toggler-icon"></span>                          </button>                          <div className="collapse navbar-collapse" id="navbarSupportedContent">                              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">Home</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" href="/about">About</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">business</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">entertainment</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">general</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">health</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">science</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">sports</a>                                  </li>                                  <li className="nav-item">                                      <a className="nav-link" aria-current="page" href="/">technology</a>                                  </li>                              </ul>                          </div>                      </div>                  </nav>              </div>          )      }  }  export default NavBar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  export class News extends Component {      static defaultProps ={      country: 'us',      pageSize: 8,      category: 'general',    }      static propTypes ={      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: [],        loading: false,        page: 1      }    }    async componentDidMount() {      console.log("cdm");      let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=1&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false })    }    handlePrevClick = async () => {      console.log("Previous");      let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page - 1}&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        page: this.state.page - 1,        articles: parsedData.articles,        loading: false      });    }    handleNextClick = async () => {      console.log("Next");      if (!(this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize))) {        console.log("cdm");        let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page + 1}&pageSize=${this.props.pageSize}`;        this.setState({loading: true});        let data = await fetch(url);        let parsedData = await data.json();        this.setState({          page: this.state.page + 1,          articles: parsedData.articles,          loading: false        });      }    }    render() {      console.log("render");      return (        <div className="container my-3">          <h1 className='text-center' style={{margin: '35px 0px'}}>NewsMonkey - Top Headlines</h1>          {this.state.loading && <Spinner/>}          <div className="row">            {!this.state.loading && this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {      render() {      let {title, description, imageUrl, newsUrl} = this.props;      return (        <div className="my-3">          <div className="card">            <img src={!imageUrl?"https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg":imageUrl} className="card-img-top" alt="..." />              <div className="card-body">                <h5 className="card-title">{title}</h5>                <p className="card-text">{description}</p>                <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>              </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\Spinner.js  import React, { Component } from 'react'  import loading from './loading.svg'  export class Spinner extends Component {    render() {      return (        <div className='text-center'>          <img src={loading} alt="loading" />        </div>      )    }  }  export default Spinner  // rce |

|  |  |
| --- | --- |
| **31. Fetching News category wise in NewsMonkey React App** | **Date: 28/12/24** |

|  |
| --- |
| Terminal:  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd newsapp  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp> npm install react-router-dom  added 6 packages, and audited 1376 packages in 7s  269 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  export default class App extends Component {    render() {      return (        <div>          <Router>            <NavBar />            <Routes>              <Route exact path="/general" element={<News key="general" pageSize={5} country="us" category="general" />} />              <Route exact path="/business" element={<News key="business" pageSize={5} country="us" category="business" />} />              <Route exact path="/entertainment" element={<News key="entertainment" pageSize={5} country="us" category="entertainment" />} />              <Route exact path="/health" element={<News key="health" pageSize={5} country="us" category="health" />} />              <Route exact path="/science" element={<News key="science" pageSize={5} country="us" category="science" />} />              <Route exact path="/sports" element={<News key="sports" pageSize={5} country="us" category="sports" />} />              <Route exact path="/technology" element={<News key="technology" pageSize={5} country="us" category="technology" />} />            </Routes>          </Router>        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react';  import { Link } from 'react-router-dom';  export class NavBar extends Component {      render() {          return (              <div>                  <nav className="navbar navbar-expand-lg navbar-dark bg-dark">                      <div className="container-fluid">                          <Link className="navbar-brand" to="/">NewsMonkey</Link>                          <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                              <span className="navbar-toggler-icon"></span>                          </button>                          <div className="collapse navbar-collapse" id="navbarSupportedContent">                              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                                  <li className="nav-item">                                      <Link className="nav-link" aria-current="page" to="/">Home</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/business">Business</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/entertainment">Entertainment</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/general">General</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/health">Health</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/science">Science</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/sports">Sports</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/technology">Technology</Link>                                  </li>                              </ul>                          </div>                      </div>                  </nav>              </div>          )      }  }  export default NavBar |

|  |  |
| --- | --- |
| **32. Adding time, author and news source to News React App** | **Date: 28/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  export class News extends Component {      static defaultProps ={      country: 'us',      pageSize: 8,      category: 'general',    }      static propTypes ={      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    constructor() {      super();      console.log("Hello I am a Constructor from News Component");      this.state = {        articles: [],        loading: false,        page: 1      }    }    async componentDidMount() {      console.log("cdm");      let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=1&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      console.log(parsedData);      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false })    }    handlePrevClick = async () => {      console.log("Previous");      let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page - 1}&pageSize=${this.props.pageSize}`;      this.setState({loading: true});      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        page: this.state.page - 1,        articles: parsedData.articles,        loading: false      });    }    handleNextClick = async () => {      console.log("Next");      if (!(this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize))) {        console.log("cdm");        let url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page + 1}&pageSize=${this.props.pageSize}`;        this.setState({loading: true});        let data = await fetch(url);        let parsedData = await data.json();        this.setState({          page: this.state.page + 1,          articles: parsedData.articles,          loading: false        });      }    }    render() {      console.log("render");      return (        <div className="container my-3">          <h1 className='text-center' style={{margin: '35px 0px'}}>NewsMonkey - Top Headlines</h1>          {this.state.loading && <Spinner/>}          <div className="row">            {!this.state.loading && this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} author={element.author} date={element.publishedAt} source={element.source.name} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">          <span class="position-absolute top-0 translate-middle badge rounded-pill bg-danger" style={{left: '90%', zIndex: '1'}}>{source}</span>            <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />            <div className="card-body">              <h5 className="card-title">{title}</h5>              <p className="card-text">{description}</p>              <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>              <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>            </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **33. Refactoring News component to use the same function** | **Date: 28/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  export default class App extends Component {    pageSize = 5;    render() {      return (        <div>          <Router>            <NavBar />            <Routes>              <Route exact path="/" element={<News key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/general" element={<News key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/business" element={<News key="business" pageSize={this.pageSize} country="us" category="business" />} />              <Route exact path="/entertainment" element={<News key="entertainment" pageSize={this.pageSize} country="us" category="entertainment" />} />              <Route exact path="/health" element={<News key="health" pageSize={this.pageSize} country="us" category="health" />} />              <Route exact path="/science" element={<News key="science" pageSize={this.pageSize} country="us" category="science" />} />              <Route exact path="/sports" element={<News key="sports" pageSize={this.pageSize} country="us" category="sports" />} />              <Route exact path="/technology" element={<News key="technology" pageSize={this.pageSize} country="us" category="technology" />} />            </Routes>          </Router>        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  export class News extends Component {    static defaultProps = {      country: 'us',      pageSize: 8,      category: 'general',    }    static propTypes = {      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    constructor() {      super();      this.state = {        articles: [],        loading: false,        page: 1      }    }    async updateNews() {      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      this.setState({ loading: true });      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false      })    }    async componentDidMount() {      this.updateNews();    }    handlePrevClick = async () => {      this.setState({ page: this.state.page - 1 });      this.updateNews();    }    handleNextClick = async () => {      this.setState({ page: this.state.page + 1 });      this.updateNews();    }    render() {      return (        <div className="container my-3">          <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top Headlines</h1>          {this.state.loading && <Spinner />}          <div className="row">            {!this.state.loading && this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} author={element.author} date={element.publishedAt} source={element.source.name} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">          <span className="position-absolute top-0 translate-middle badge rounded-pill bg-danger" style={{left: '90%', zIndex: '1'}}>{source}</span>            <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />            <div className="card-body">              <h5 className="card-title">{title}</h5>              <p className="card-text">{description}</p>              <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>              <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>            </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **34. React Component Lifecycle & Lifecycle methods** | **Date: 28/12/24** |



A close-up of a computer component

Description automatically generated

A screenshot of a diagram

Description automatically generated

https://projects.wojtekmaj.pl/react-lifecycle-methods-diagram/

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  export class News extends Component {    static defaultProps = {      country: 'us',      pageSize: 8,      category: 'general',    }    static propTypes = {      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    constructor(props) {      super(props);      this.state = {        articles: [],        loading: false,        page: 1      }      document.title = `${this.capitalizeFirstLetter(this.props.category)} - NewsMonkey`;    }    async updateNews() {      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      this.setState({ loading: true });      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false      })    }    async componentDidMount() {      this.updateNews();    }    handlePrevClick = async () => {      this.setState({ page: this.state.page - 1 });      this.updateNews();    }    handleNextClick = async () => {      this.setState({ page: this.state.page + 1 });      this.updateNews();    }    render() {      return (        <div className="container my-3">          <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top {this.capitalizeFirstLetter(this.props.category)} Headlines</h1>          {this.state.loading && <Spinner />}          <div className="row">            {!this.state.loading && this.state.articles.map((element) => {              return <div className="col-md-4" key={element.url}>                <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} author={element.author} date={element.publishedAt} source={element.source.name} />              </div>            })}          </div>          <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div>        </div>      )    }  }  export default News  // rce (class base component) |

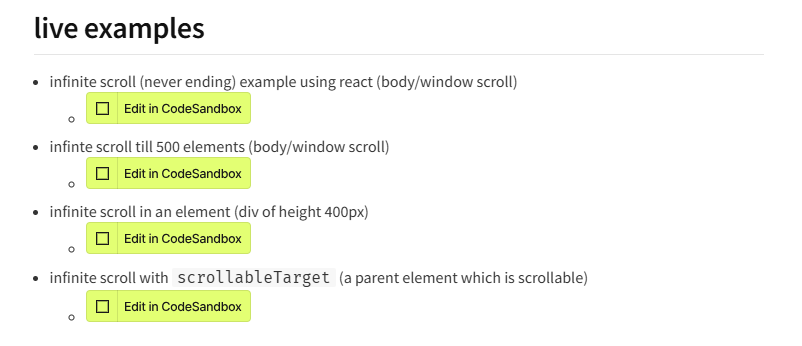
|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">          <span className="position-absolute top-0 translate-middle badge rounded-pill bg-danger" style={{left: '90%', zIndex: '1'}}>{source}</span>            <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />            <div className="card-body">              <h5 className="card-title">{title}</h5>              <p className="card-text">{description}</p>              <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>              <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>            </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **35. Adding Infinite Scroll to NewsMonkey** | **Date: 28/12/24** |

<https://www.npmjs.com/package/react-infinite-scroll-component>

A white rectangular object with black text

Description automatically generated



|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd newsapp  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp> npm i react-infinite-scroll-component  added 2 packages, and audited 1378 packages in 6s  269 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\Spinner.js  import React, { Component } from 'react'  import loading from './loading.svg'  export class Spinner extends Component {    render() {      return (        <div className='text-center'>          <img className="my-3" src={loading} alt="loading" />        </div>      )    }  }  export default Spinner  // rce |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  import InfiniteScroll from "react-infinite-scroll-component";  export class News extends Component {    static defaultProps = {      country: 'us',      pageSize: 8,      category: 'general',    }    static propTypes = {      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    constructor(props) {      super(props);      this.state = {        articles: [],        loading: true,        page: 1,        totalResults: 0      }      document.title = `${this.capitalizeFirstLetter(this.props.category)} - NewsMonkey`;    }    async updateNews() {      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      this.setState({ loading: true });      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false,      })    }    async componentDidMount() {      this.updateNews();    }    handlePrevClick = async () => {      this.setState({ page: this.state.page - 1 });      this.updateNews();    }    handleNextClick = async () => {      this.setState({ page: this.state.page + 1 });      this.updateNews();    }    fetchMoreData = async () => {      this.setState({ page: this.state.page + 1 })      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: this.state.articles.concat(parsedData.articles),        totalResults: parsedData.totalResults      })    };    render() {      return (        <>          <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top {this.capitalizeFirstLetter(this.props.category)} Headlines</h1>          {this.state.loading && <Spinner />}          <InfiniteScroll            dataLength={this.state.articles.length}            next={this.fetchMoreData}            hasMore={this.state.articles.length !== this.state.totalResults}            loader={<Spinner />}>            <div className="container">              <div className="row">                {this.state.articles.map((element) => {                  return <div className="col-md-4" key={element.url}>                    <NewsItem title={element.title ? element.title : ""} description={element.description ? element.description : ""} imageUrl={element.urlToImage} newsUrl={element.url} author={element.author} date={element.publishedAt} source={element.source.name} />                  </div>                })}              </div>            </div>          </InfiniteScroll>          {/\* <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div> \*/}        </>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">          <span className="position-absolute top-0 translate-middle badge rounded-pill bg-danger" style={{left: '90%', zIndex: '1'}}>{source}</span>            <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />            <div className="card-body">              <h5 className="card-title">{title}</h5>              <p className="card-text">{description}</p>              <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>              <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>            </div>          </div>        </div>      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **36. Adding a top loading bar to NewsMonkey** | **Date: 30/12/24** |

https://www.npmjs.com/package/react-top-loading-bar

A close up of a sign

Description automatically generated

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd newsapp  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp> npm i react-top-loading-bar  added 1 package, and audited 1379 packages in 12s  269 packages are looking for funding  run `npm fund` for details  8 vulnerabilities (2 moderate, 6 high)  To address all issues (including breaking changes), run:  npm audit fix --force  Run `npm audit` for details. |



|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import LoadingBar from "react-top-loading-bar";  export default class App extends Component {    pageSize = 5;    state = {      progress: 0    }    setProgress = (progress) => {      this.setState({progress: progress});    }    render() {      return (        <div>          <Router>            <NavBar />            <LoadingBar              height={3}              color="#f11946"              progress={this.state.progress}            />            <Routes>              <Route exact path="/" element={<News setProgress={this.setProgress} key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/general" element={<News setProgress={this.setProgress} key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/business" element={<News setProgress={this.setProgress} key="business" pageSize={this.pageSize} country="us" category="business" />} />              <Route exact path="/entertainment" element={<News setProgress={this.setProgress} key="entertainment" pageSize={this.pageSize} country="us" category="entertainment" />} />              <Route exact path="/health" element={<News setProgress={this.setProgress} key="health" pageSize={this.pageSize} country="us" category="health" />} />              <Route exact path="/science" element={<News setProgress={this.setProgress} key="science" pageSize={this.pageSize} country="us" category="science" />} />              <Route exact path="/sports" element={<News setProgress={this.setProgress} key="sports" pageSize={this.pageSize} country="us" category="sports" />} />              <Route exact path="/technology" element={<News setProgress={this.setProgress} key="technology" pageSize={this.pageSize} country="us" category="technology" />} />            </Routes>          </Router>        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react';  import { Link } from 'react-router-dom';  export class NavBar extends Component {      render() {          return (              <div>                  <nav className="navbar navbar-expand-lg navbar-dark bg-dark">                      <div className="container-fluid">                          <Link className="navbar-brand" to="/">NewsMonkey</Link>                          <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                              <span className="navbar-toggler-icon"></span>                          </button>                          <div className="collapse navbar-collapse" id="navbarSupportedContent">                              <ul className="navbar-nav me-auto mb-2 mb-lg-0">                                  <li className="nav-item">                                      <Link className="nav-link" aria-current="page" to="/">Home</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/business">Business</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/entertainment">Entertainment</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/general">General</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/health">Health</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/science">Science</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/sports">Sports</Link>                                  </li>                                  <li className="nav-item">                                      <Link className="nav-link" to="/technology">Technology</Link>                                  </li>                              </ul>                          </div>                      </div>                  </nav>              </div>          )      }  }  export default NavBar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  import InfiniteScroll from "react-infinite-scroll-component";  export class News extends Component {    static defaultProps = {      country: 'us',      pageSize: 8,      category: 'general',    }    static propTypes = {      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    constructor(props) {      super(props);      this.state = {        articles: [],        loading: true,        page: 1,        totalResults: 0      }      document.title = `${this.capitalizeFirstLetter(this.props.category)} - NewsMonkey`;    }    async updateNews() {      this.props.setProgress(10);      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      this.setState({ loading: true });      let data = await fetch(url);      this.props.setProgress(30);      let parsedData = await data.json();      this.props.setProgress(70);      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false,      })      this.props.setProgress(100);    }    async componentDidMount() {      this.updateNews();    }    handlePrevClick = async () => {      this.setState({ page: this.state.page - 1 });      this.updateNews();    }    handleNextClick = async () => {      this.setState({ page: this.state.page + 1 });      this.updateNews();    }    fetchMoreData = async () => {      this.setState({ page: this.state.page + 1 })      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=ecb6b24c8cd54f3fb15590e183275f0b&page=${this.state.page}&pageSize=${this.props.pageSize}`;      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: this.state.articles.concat(parsedData.articles),        totalResults: parsedData.totalResults      })    };    render() {      return (        <>          <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top {this.capitalizeFirstLetter(this.props.category)} Headlines</h1>          {this.state.loading && <Spinner />}          <InfiniteScroll            dataLength={this.state.articles.length}            next={this.fetchMoreData}            hasMore={this.state.articles.length !== this.state.totalResults}            loader={<Spinner />}>            <div className="container">              <div className="row">                {this.state.articles.map((element) => {                  return <div className="col-md-4" key={element.url}>                    <NewsItem                      title={element.title ? element.title : ""}                      description={element.description ? element.description : ""}                      imageUrl={element.urlToImage}                      newsUrl={element.url}                      author={element.author}                      date={element.publishedAt}                      source={element.source.name}                    />                  </div>                })}              </div>            </div>          </InfiniteScroll>          {/\* <div className="container d-flex justify-content-between">            <button disabled={this.state.page <= 1} type="button" className="btn btn-dark" onClick={this.handlePrevClick}> &larr; Previous</button>            <button disabled={this.state.page + 1 > Math.ceil(this.state.totalResults / this.props.pageSize)} type="button" className="btn btn-dark" onClick={this.handleNextClick}>Next &rarr; </button>          </div> \*/}        </>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">            <div style={{display:'flex', justifyContent:'flex-end', position:'absolute', right:'0'}}>            <span className="badge rounded-pill bg-danger">{source}</span>          </div>          <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />          <div className="card-body">            <h5 className="card-title">{title}</h5>            <p className="card-text">{description}</p>            <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>            <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>          </div>        </div>        </div >      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **37. Hiding API Key by Adding Custom Environment Variables** | **Date: 30/12/24** |

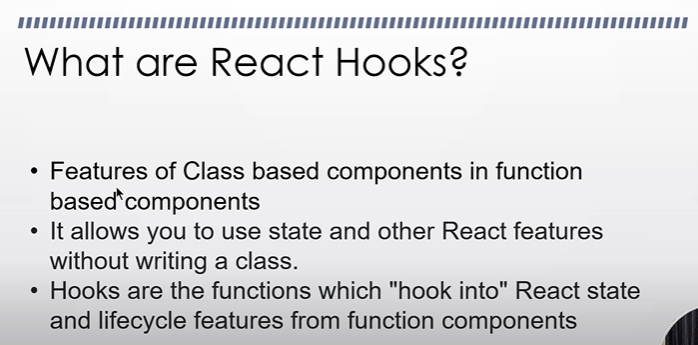
|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\.env.local  REACT\_APP\_NEWS\_API="ecb6b24c8cd54f3fb15590e183275f0b" |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { Component } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import LoadingBar from "react-top-loading-bar";  export default class App extends Component {    pageSize = 5;    apiKey = process.env.REACT\_APP\_NEWS\_API    state = {      progress: 0    }    setProgress = (progress) => {      this.setState({progress: progress});    }    render() {      return (        <div>          <Router>            <NavBar />            <LoadingBar              height={3}              color="#f11946"              progress={this.state.progress}            />            <Routes>              <Route exact path="/" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/general" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="general" pageSize={this.pageSize} country="us" category="general" />} />              <Route exact path="/business" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="business" pageSize={this.pageSize} country="us" category="business" />} />              <Route exact path="/entertainment" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="entertainment" pageSize={this.pageSize} country="us" category="entertainment" />} />              <Route exact path="/health" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="health" pageSize={this.pageSize} country="us" category="health" />} />              <Route exact path="/science" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="science" pageSize={this.pageSize} country="us" category="science" />} />              <Route exact path="/sports" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="sports" pageSize={this.pageSize} country="us" category="sports" />} />              <Route exact path="/technology" element={<News setProgress={this.setProgress} apiKey={this.apiKey} key="technology" pageSize={this.pageSize} country="us" category="technology" />} />            </Routes>          </Router>        </div>      )    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { Component } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  import InfiniteScroll from "react-infinite-scroll-component";  export class News extends Component {    static defaultProps = {      country: 'us',      pageSize: 8,      category: 'general',    }    static propTypes = {      country: PropTypes.string, //pts [shortcut]      pageSize: PropTypes.number, //ptn [shortcut]      category: PropTypes.string,    }    capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    constructor(props) {      super(props);      this.state = {        articles: [],        loading: true,        page: 1,        totalResults: 0      }      document.title = `${this.capitalizeFirstLetter(this.props.category)} - NewsMonkey`;    }    async updateNews() {      this.props.setProgress(10);      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=${this.props.apiKey}&page=${this.state.page}&pageSize=${this.props.pageSize}`;      this.setState({ loading: true });      let data = await fetch(url);      this.props.setProgress(30);      let parsedData = await data.json();      this.props.setProgress(70);      this.setState({        articles: parsedData.articles,        totalResults: parsedData.totalResults,        loading: false,      })      this.props.setProgress(100);    }    async componentDidMount() {      this.updateNews();    }    handlePrevClick = async () => {      this.setState({ page: this.state.page - 1 });      this.updateNews();    }    handleNextClick = async () => {      this.setState({ page: this.state.page + 1 });      this.updateNews();    }    fetchMoreData = async () => {      this.setState({ page: this.state.page + 1 })      const url = `https://newsapi.org/v2/top-headlines?country=${this.props.country}&category=${this.props.category}&apiKey=${this.props.apiKey}&page=${this.state.page}&pageSize=${this.props.pageSize}`;      let data = await fetch(url);      let parsedData = await data.json();      this.setState({        articles: this.state.articles.concat(parsedData.articles),        totalResults: parsedData.totalResults      })    };    render() {      return (        <>          <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top {this.capitalizeFirstLetter(this.props.category)} Headlines</h1>          {this.state.loading && <Spinner />}          <InfiniteScroll            dataLength={this.state.articles.length}            next={this.fetchMoreData}            hasMore={this.state.articles.length !== this.state.totalResults}            loader={<Spinner />}>            <div className="container">              <div className="row">                {this.state.articles.map((element) => {                  return <div className="col-md-4" key={element.url}>                    <NewsItem                      title={element.title ? element.title : ""}                      description={element.description ? element.description : ""}                      imageUrl={element.urlToImage}                      newsUrl={element.url}                      author={element.author}                      date={element.publishedAt}                      source={element.source.name}                    />                  </div>                })}              </div>            </div>          </InfiniteScroll>        </>      )    }  }  export default News  // rce (class base component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React, { Component } from 'react'  export class NewsItem extends Component {    render() {      let { title, description, imageUrl, newsUrl, author, date, source } = this.props;      return (        <div className="my-3">          <div className="card">            <div style={{display:'flex', justifyContent:'flex-end', position:'absolute', right:'0'}}>            <span className="badge rounded-pill bg-danger">{source}</span>          </div>          <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />          <div className="card-body">            <h5 className="card-title">{title}</h5>            <p className="card-text">{description}</p>            <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>            <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>          </div>        </div>        </div >      )    }  }  export default NewsItem  // rce (class based component) |

|  |  |
| --- | --- |
| **38. Introduction to React Hooks** | **Date: 30/12/24** |



s

|  |  |
| --- | --- |
| **39. Changing Class based NewsMonkey components to Functional based** | **Date: 30/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React, { Component } from 'react';  import { Link } from 'react-router-dom';  const NavBar = () => {      return (          <div>              <nav className="navbar navbar-expand-lg navbar-dark bg-dark">                  <div className="container-fluid">                      <Link className="navbar-brand" to="/">NewsMonkey</Link>                      <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                          <span className="navbar-toggler-icon"></span>                      </button>                      <div className="collapse navbar-collapse" id="navbarSupportedContent">                          <ul className="navbar-nav me-auto mb-2 mb-lg-0">                              <li className="nav-item">                                  <Link className="nav-link" aria-current="page" to="/">Home</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/business">Business</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/entertainment">Entertainment</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/general">General</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/health">Health</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/science">Science</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/sports">Sports</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/technology">Technology</Link>                              </li>                          </ul>                      </div>                  </div>              </nav>          </div>      )  }  export default NavBar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\Spinner.js  import React, { Component } from 'react'  import loading from './loading.svg'  const Spinner = () => {    return (      <div className='text-center'>        <img className="my-3" src={loading} alt="loading" />      </div>    )  }  export default Spinner |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React from 'react'  const NewsItem = (props) => {    let { title, description, imageUrl, newsUrl, author, date, source } = props;    return (      <div className="my-3">        <div className="card">          <div style={{ display: 'flex', justifyContent: 'flex-end', position: 'absolute', right: '0' }}>            <span className="badge rounded-pill bg-danger">{source}</span>          </div>          <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />          <div className="card-body">            <h5 className="card-title">{title}</h5>            <p className="card-text">{description}</p>            <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>            <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>          </div>        </div>      </div >    )  }  export default NewsItem  // rce (class based component) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { useEffect, useState } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  import InfiniteScroll from "react-infinite-scroll-component";  const News = (props) => {    const [articles, setArticles] = useState([])    const [loading, setLoading] = useState(true)    const [page, setPage] = useState(1)    const [totalResults, settotalResults] = useState(0)    // document.title = `${capitalizeFirstLetter(props.category)} - NewsMonkey`;    const capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    const updateNews = async () => {      props.setProgress(10);      const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=${props.apiKey}&page=${page}&pageSize=${props.pageSize}`;      setLoading(true);      let data = await fetch(url);      props.setProgress(30);      let parsedData = await data.json();      props.setProgress(70);      setArticles(parsedData.articles)      settotalResults(parsedData.totalResults)      setLoading(false)      props.setProgress(100);    }    useEffect(() => {      updateNews();    }, [])    const handlePrevClick = async () => {      setPage(page - 1);      updateNews();    }    const handleNextClick = async () => {      setPage(page + 1);      updateNews();    }    const fetchMoreData = async () => {      setPage(page + 1);      const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=${props.apiKey}&page=${page}&pageSize=${props.pageSize}`;      let data = await fetch(url);      let parsedData = await data.json();      setArticles(articles.concat(parsedData.articles));      settotalResults(parsedData.totalResults);    };    return (      <>        <h1 className='text-center' style={{ margin: '35px 0px' }}>NewsMonkey - Top {capitalizeFirstLetter(props.category)} Headlines</h1>        {loading && <Spinner />}        <InfiniteScroll          dataLength={articles.length}          next={fetchMoreData}          hasMore={articles.length !== totalResults}          loader={<Spinner />}>          <div className="container">            <div className="row">              {articles.map((element) => {                return <div className="col-md-4" key={element.url}>                  <NewsItem                    title={element.title ? element.title : ""}                    description={element.description ? element.description : ""}                    imageUrl={element.urlToImage}                    newsUrl={element.url}                    author={element.author}                    date={element.publishedAt}                    source={element.source.name}                  />                </div>              })}            </div>          </div>        </InfiniteScroll>      </>    )  }  News.defaultProps = {    country: 'us',    pageSize: 8,    category: 'general',  }  News.propTypes = {    country: PropTypes.string, //pts [shortcut]    pageSize: PropTypes.number, //ptn [shortcut]    category: PropTypes.string,  }  export default News  // rce (class base component) |

|  |  |
| --- | --- |
| **40. Sticky Navbar & NewsMonkey bug fixes** | **Date: 31/12/24** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NavBar.js  import React from 'react';  import { Link } from 'react-router-dom';  const NavBar = () => {      return (          <div>              <nav className="navbar fixed-top navbar-expand-lg navbar-dark bg-dark">                  <div className="container-fluid">                      <Link className="navbar-brand" to="/">NewsMonkey</Link>                      <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                          <span className="navbar-toggler-icon"></span>                      </button>                      <div className="collapse navbar-collapse" id="navbarSupportedContent">                          <ul className="navbar-nav me-auto mb-2 mb-lg-0">                              <li className="nav-item">                                  <Link className="nav-link" aria-current="page" to="/">Home</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/business">Business</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/entertainment">Entertainment</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/general">General</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/health">Health</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/science">Science</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/sports">Sports</Link>                              </li>                              <li className="nav-item">                                  <Link className="nav-link" to="/technology">Technology</Link>                              </li>                          </ul>                      </div>                  </div>              </nav>          </div>      )  }  export default NavBar |

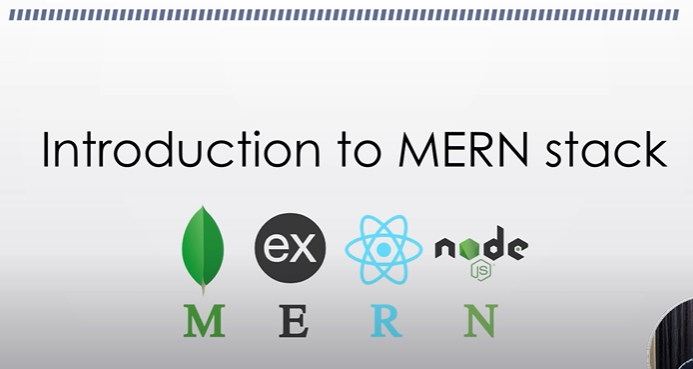
|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\Spinner.js  import React from 'react'  import loading from './loading.svg'  const Spinner = () => {    return (      <div className='text-center'>        <img className="my-3" src={loading} alt="loading" />      </div>    )  }  export default Spinner |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\NewsItem.js  import React from 'react'  const NewsItem = (props) => {    let { title, description, imageUrl, newsUrl, author, date, source } = props;    return (      <div className="my-3">        <div className="card">          <div style={{ display: 'flex', justifyContent: 'flex-end', position: 'absolute', right: '0' }}>            <span className="badge rounded-pill bg-danger">{source}</span>          </div>          <img src={!imageUrl ? "https://assets.bwbx.io/images/users/iqjWHBFdfxIU/iIzW7Rc7W6y8/v1/1200x772.jpg" : imageUrl} className="card-img-top" alt="..." />          <div className="card-body">            <h5 className="card-title">{title}</h5>            <p className="card-text">{description}</p>            <p className="card-text"><small className="text-body-secondary">By {!author ? "Unknown" : author} on {new Date(date).toGMTString()}</small></p>            <a rel="noreferrer" href={newsUrl} target="\_blank" className="btn btn-sm btn-dark">Read More</a>          </div>        </div>      </div >    )  }  export default NewsItem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\components\News.js  import React, { useEffect, useState } from 'react';  import NewsItem from './NewsItem';  import Spinner from './Spinner';  import PropTypes from 'prop-types';  import InfiniteScroll from "react-infinite-scroll-component";  const News = (props) => {    const [articles, setArticles] = useState([])    const [loading, setLoading] = useState(true)    const [page, setPage] = useState(1)    const [totalResults, settotalResults] = useState(0)    const capitalizeFirstLetter = (string) => {      return string.charAt(0).toUpperCase() + string.slice(1);    }    const updateNews = async () => {      props.setProgress(10);      const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=${props.apiKey}&page=${page}&pageSize=${props.pageSize}`;      setLoading(true);      let data = await fetch(url);      props.setProgress(30);      let parsedData = await data.json();      props.setProgress(70);      setArticles(parsedData.articles)      settotalResults(parsedData.totalResults)      setLoading(false)      props.setProgress(100);    }    useEffect(() => {      document.title = `${capitalizeFirstLetter(props.category)} - NewsMonkey`;      updateNews();      // eslint-disable-next-line    }, []);    const fetchMoreData = async () => {      const url = `https://newsapi.org/v2/top-headlines?country=${props.country}&category=${props.category}&apiKey=${props.apiKey}&page=${page+1}&pageSize=${props.pageSize}`;      setPage(page + 1);      let data = await fetch(url);      let parsedData = await data.json();      setArticles(articles.concat(parsedData.articles));      settotalResults(parsedData.totalResults);    };    return (      <>        <h1 className='text-center' style={{ margin: '35px 0px', marginTop: '90px' }}>NewsMonkey - Top {capitalizeFirstLetter(props.category)} Headlines</h1>        {loading && <Spinner />}        <InfiniteScroll          dataLength={articles.length}          next={fetchMoreData}          hasMore={articles.length !== totalResults}          loader={<Spinner />}>          <div className="container">            <div className="row">              {articles.map((element) => {                return <div className="col-md-4" key={element.url}>                  <NewsItem                    title={element.title ? element.title : ""}                    description={element.description ? element.description : ""}                    imageUrl={element.urlToImage}                    newsUrl={element.url}                    author={element.author}                    date={element.publishedAt}                    source={element.source.name}                  />                </div>              })}            </div>          </div>        </InfiniteScroll>      </>    )  }  News.defaultProps = {    country: 'us',    pageSize: 8,    category: 'general',  }  News.propTypes = {    country: PropTypes.string, //pts [shortcut]    pageSize: PropTypes.number, //ptn [shortcut]    category: PropTypes.string,  }  export default News |

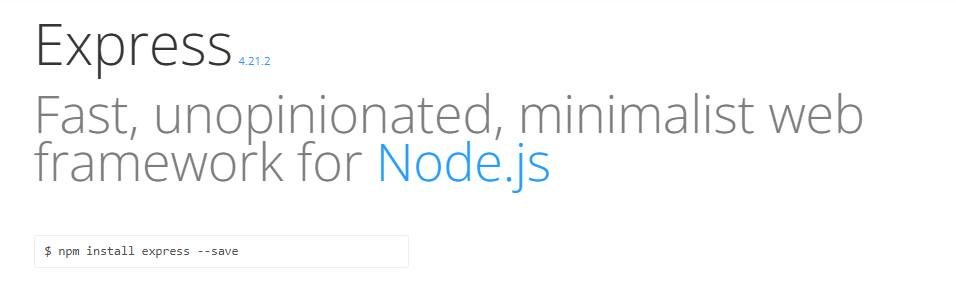
|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\newsapp\src\App.js  import './App.css';  import React, { useState } from 'react';  import NavBar from './components/NavBar';  import News from './components/News';  import { BrowserRouter as Router, Routes, Route } from "react-router-dom";  import LoadingBar from "react-top-loading-bar";  const App = () => {    const pageSize = 5;    const apiKey = process.env.REACT\_APP\_NEWS\_API      const [progress, setProgress] = useState(0);    return (      <div>        <Router>          <NavBar />          <LoadingBar            height={3}            color="#f11946"            progress={progress}          />          <Routes>            <Route exact path="/" element={<News setProgress={setProgress} apiKey={apiKey} key="general" pageSize={pageSize} country="us" category="general" />} />            <Route exact path="/general" element={<News setProgress={setProgress} apiKey={apiKey} key="general" pageSize={pageSize} country="us" category="general" />} />            <Route exact path="/business" element={<News setProgress={setProgress} apiKey={apiKey} key="business" pageSize={pageSize} country="us" category="business" />} />            <Route exact path="/entertainment" element={<News setProgress={setProgress} apiKey={apiKey} key="entertainment" pageSize={pageSize} country="us" category="entertainment" />} />            <Route exact path="/health" element={<News setProgress={setProgress} apiKey={apiKey} key="health" pageSize={pageSize} country="us" category="health" />} />            <Route exact path="/science" element={<News setProgress={setProgress} apiKey={apiKey} key="science" pageSize={pageSize} country="us" category="science" />} />            <Route exact path="/sports" element={<News setProgress={setProgress} apiKey={apiKey} key="sports" pageSize={pageSize} country="us" category="sports" />} />            <Route exact path="/technology" element={<News setProgress={setProgress} apiKey={apiKey} key="technology" pageSize={pageSize} country="us" category="technology" />} />          </Routes>        </Router>      </div>    )  }  export default App; |

|  |  |
| --- | --- |
| **41. Introduction to MERN stack** | **Date: 31/12/24** |



A white paper with black text

Description automatically generated



|  |  |
| --- | --- |
| **42. Project 3: iNotebook backend and React frontend setup** | **Date: 31/12/24** |

|  |
| --- |
| **Create backend folder in inotebook**  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js> cd inotebook\backend  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npm init  This utility will walk you through creating a package.json file.  It only covers the most common items, and tries to guess sensible defaults.  See `npm help init` for definitive documentation on these fields  and exactly what they do.  Use `npm install <pkg>` afterwards to install a package and  save it as a dependency in the package.json file.  Press ^C at any time to quit.  package name: (backend) inotebook-backend  version: (1.0.0)  description: iNotebook - Your notebook on the cloud  entry point: (index.js)  test command:  git repository:  keywords:  author:  license: (ISC)  About to write to C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\package.json:  {  "name": "inotebook-backend",  "version": "1.0.0",  "description": "iNotebook - Your notebook on the cloud",  "main": "index.js",  "scripts": {  "test": "echo \"Error: no test specified\" && exit 1"  },  "license": "ISC"  }  Is this OK? (yes) yes  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npm i express  added 68 packages, and audited 69 packages in 8s  14 packages are looking for funding  run `npm fund` for details  found 0 vulnerabilities  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npm i mongoose  added 20 packages, and audited 89 packages in 16s  15 packages are looking for funding  run `npm fund` for details  found 0 vulnerabilities |

|  |  |
| --- | --- |
| **43. iNotebook db & Express Server Setup** | **Date: 01/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\package.json  {    "name": "inotebook-backend",    "version": "1.0.0",    "description": "iNotebook - Your notebook on the cloud",    "main": "index.js",    "scripts": {      "test": "echo \"Error: no test specified\" && exit 1"    },    "author": "",    "license": "ISC",    "dependencies": {      "express": "^4.21.2",      "mongoose": "^8.9.3"    },    "devDependencies": {      "nodemon": "^3.1.9"    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\db.js  const mongoose = require('mongoose');  const mongoURI = "mongodb://localhost:27017/inotebook"; // Add a database name, e.g., "inotebook"  const connectToMongo = async () => {      try {          await mongoose.connect(mongoURI, {              useNewUrlParser: true,              useUnifiedTopology: true,          });          console.log("Connected to Mongo Successfully");      } catch (error) {          console.error("Error connecting to Mongo:", error);          process.exit(1); // Exit the process with an error code      }  };  module.exports = connectToMongo; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  connectToMongo(); |

|  |
| --- |
| npm i -D nodemon (-D is for don’t make the part of Dependencies)  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook> cd backend  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npm i -D nodemon  added 29 packages, and audited 118 packages in 5s  19 packages are looking for funding  run `npm fund` for details  found 0 vulnerabilities  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npx nodemon .\index.js  [nodemon] 3.1.9  [nodemon] to restart at any time, enter `rs`  [nodemon] watching path(s): \*.\*  [nodemon] watching extensions: js,mjs,cjs,json  [nodemon] starting `node .\index.js`  (node:15248) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version  (Use `node --trace-warnings ...` to show where the warning was created)  (node:15248) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and  will be removed in the next major version  Connected to Mongo Successfully |

A screenshot of a computer

Description automatically generated

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 3000  app.get('/', (req, res) => {    res.send('Hello World!')  })  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| **Use thunder client instead of Postman** |

|  |  |
| --- | --- |
| **44. Creating Routes & Schema for Storing Notes & Users** | **Date: 02/01/25** |

|  |
| --- |
| When creating model give first letter Capital like “Notes.js” |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\Notes.js  const { type } = require('@testing-library/user-event/dist/type');  const mongoose = require('mongoose');  const NotesSchema = new Schema({      title: {          type: String,          required: true      },      description: {          type: String,          required: true      },      tag: {          type: String,          default: "General"      },      date: {          type: Date,          default: Date.now      }    });    module.exports = mongoose.model('notes', NotesSchema); |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const { type } = require('@testing-library/user-event/dist/type');  const mongoose = require('mongoose');  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    module.exports = mongoose.model('User', UserSchema); |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const router = express.Router();  router.get('/', (req, res) => {      obj = {          a: 'thios',          number: 34      }      res.json(obj);  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\notes.js  const express = require('express');  const router = express.Router();  router.get('/', (req, res) => {        res.json([]);  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 3000  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |  |
| --- | --- |
| **45. Storing Data into the Database Using Mongoose Model** | **Date: 02/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 3000  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  // Create a User using: POST "/api/auth/". Doesn't require Auth  router.post('/', (req, res) => {      console.log(req.body);      const user = User(req.body);      user.save();      res.send(req.body);  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    module.exports = mongoose.model('User', UserSchema); |

|  |  |
| --- | --- |
| **46. Adding Data Validation Using express validator** | **Date: 03/01/25** |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook> cd backend  PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npm install express-validator  added 3 packages, and audited 121 packages in 10s  19 packages are looking for funding  run `npm fund` for details  found 0 vulnerabilities |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  // Create a User using: POST "/api/auth/". Doesn't require Auth  router.post('/', [      body('name').isLength({ min: 3 }),      body('email').isEmail(),      body('password').isLength({ min: 5 }),  ], (req, res) => {      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      res.send(req.body);  })  module.exports = router; |

|  |
| --- |
| PS C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend> npx nodemon .\index.js  [nodemon] 3.1.9  [nodemon] to restart at any time, enter `rs`  [nodemon] watching path(s): \*.\*  [nodemon] watching extensions: js,mjs,cjs,json  [nodemon] starting `node .\index.js`  (node:7872) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version  (Use `node --trace-warnings ...` to show where the warning was created)  (node:7872) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version  Example app listening on port 3000  Connected to Mongo Successfully    **Update version**  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  // Create a User using: POST "/api/auth/". Doesn't require Auth  router.post('/', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], (req, res) => {      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      User.create({          name: req.body.name,          password: req.body.password,          email: req.body.email,      }).then(user => res.json(user));  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    const User = mongoose.model('User', UserSchema);    User.createIndexes();    module.exports = User; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 3000  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  // Create a User using: POST "/api/auth/". Doesn't require Auth  router.post('/', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], (req, res) => {      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      User.create({          name: req.body.name,          password: req.body.password,          email: req.body.email,      }).then(user => res.json(user))      .catch(err => {console.log(err)          res.json({error: 'Please enter a unquie value for email', message: err.message})      })  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    const User = mongoose.model('User', UserSchema);    User.createIndexes();    module.exports = User; |

|  |  |
| --- | --- |
| **47. Creating ThunderClient Collections to Manage Requests** | **Date: 03/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 5000  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  // Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      // Check whether the user with this email exists already      try {          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          user = await User.create({              name: req.body.name,              password: req.body.password,              email: req.body.email,          })          res.json(user)      } catch (error) {          console.error(error.message);          res.status(500).send("Some Error occured");      }  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    const User = mongoose.model('User', UserSchema);    module.exports = User; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **48. Understanding Password Hashing, Salt & Pepper** | **Date: 03/01/25** |

|  |
| --- |
| Hashing Passwords with Bcrypt, Salt, and Pepper |

|  |  |
| --- | --- |
| **49. Hashing Passwords using bcryptjs in NodeJs** | **Date: 03/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\User.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const UserSchema = new Schema({      name: {          type: String,          required: true      },      email: {          type: String,          required: true,          unique: true      },      password: {          type: String,          required: true      },      date: {          type: Date,          default: Date.now      }    });    const User = mongoose.model('User', UserSchema);    module.exports = User; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 5000  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  const JWT\_SECRET = 'Harryisagoodb$oy'  // Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Some Error occured");      }  })  module.exports = router; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **50. Creating Login Endpoint & sending auth token** | **Date: 03/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  connectToMongo();  const app = express()  const port = 5000  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`Example app listening on port ${port}`)  }) |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  const JWT\_SECRET = 'Harryisagoodb$oy';  // Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // Authenticate the User using: POST "/api/auth/login". No login required  router.post('/login', [      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password cannot be blank').exists(),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      const {email, password} = req.body;      try {          let user = await User.findOne({email});          if (!user){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const passwordCompare = await bcrypt.compare(password, user.password);          if (!passwordCompare){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  module.exports = router; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **51. Creating a middleware to decode user from a JWT** | **Date: 04/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\middleware\fetchuser.js  var jwt = require('jsonwebtoken');  const JWT\_SECRET = 'Harryisagoodb$oy';  const fetchuser = (req, res, next) => {      // Get the user from the JWT token and add id to req object.      const token = req.header('auth-token');      if (!token) {          res.status(401).send({ error: "Please authenticate using a valid token." });      }      try {          const data = jwt.verify(token, JWT\_SECRET);          req.user = data.user;          next();      } catch (error) {          res.status(401).send({ error: "Please authenticate using a valid token." });      }  }  module.exports = fetchuser; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  var fetchuser = require('../middleware/fetchuser');  const JWT\_SECRET = 'Harryisagoodb$oy';  // ROUTE 1: Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Authenticate the User using: POST "/api/auth/login". No login required  router.post('/login', [      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password cannot be blank').exists(),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      const {email, password} = req.body;      try {          let user = await User.findOne({email});          if (!user){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const passwordCompare = await bcrypt.compare(password, user.password);          if (!passwordCompare){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 3: Get loggedin User Details using: POST "/api/auth/getuser". Login required  router.post('/getuser', fetchuser, async (req, res) => {  try {      userId = req.user.id;      const user = await User.findById(userId).select("-password");      res.send(user);  } catch (error) {      console.error(error.message);          res.status(500).send("Internal Server Error");  }  })  module.exports = router; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **52. Fetching all notes & Adding a Note** | **Date: 07/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\notes.js  const express = require('express');  const router = express.Router();  var fetchuser = require('../middleware/fetchuser');  const Note = require('../models/Note');  const { body, validationResult } = require('express-validator');  // ROUTE 1: Get All the Notes using: GET "/api/auth/getuser". Login required  router.get('/fetchallnotes', fetchuser, async (req, res) => {      try {          const notes = await Note.find({ user: req.user.id });          res.json(notes);      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Add a new Note using: POST "/api/auth/addnote". Login required  router.post('/addnote', fetchuser, [      body('title', 'Enter a valid title').isLength({ min: 3 }),      body('description', 'Description atleast must be 5 characters').isLength({ min: 5 }),], async (req, res) => {          try {              const { title, description, tag } = req.body;              // If there are errors, return bad request and the errors              const errors = validationResult(req);              if (!errors.isEmpty()) {                  return res.status(400).json({ errors: errors.array() });              }              const note = new Note({                  title, description, tag, user: req.user.id              })              const savedNote = await note.save();              res.json(savedNote);          } catch (error) {              console.error(error.message);              res.status(500).send("Internal Server Error");          }      })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\models\Note.js  const mongoose = require('mongoose');  const { Schema } = mongoose;  const NotesSchema = new Schema({      user: {          type: mongoose.Schema.Types.ObjectId,          ref: 'user'      },      title: {          type: String,          required: true      },      description: {          type: String,          required: true      },      tag: {          type: String,          default: "General"      },      date: {          type: Date,          default: Date.now      }    });    module.exports = mongoose.model('notes', NotesSchema); |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  var fetchuser = require('../middleware/fetchuser');  const JWT\_SECRET = 'Harryisagoodb$oy';  // ROUTE 1: Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Authenticate the User using: POST "/api/auth/login". No login required  router.post('/login', [      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password cannot be blank').exists(),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      const {email, password} = req.body;      try {          let user = await User.findOne({email});          if (!user){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const passwordCompare = await bcrypt.compare(password, user.password);          if (!passwordCompare){              return res.status(404).json({ errors: "Please to try to login with correct Credentials" });          }          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 3: Get loggedin User Details using: POST "/api/auth/getuser". Login required  router.post('/getuser', fetchuser, async (req, res) => {  try {      userId = req.user.id;      const user = await User.findById(userId).select("-password");      res.send(user);  } catch (error) {      console.error(error.message);          res.status(500).send("Internal Server Error");  }  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\middleware\fetchuser.js  var jwt = require('jsonwebtoken');  const JWT\_SECRET = 'Harryisagoodb$oy';  const fetchuser = (req, res, next) => {      // Get the user from the JWT token and add id to req object.      const token = req.header('auth-token');      if (!token) {          res.status(401).send({ error: "Please authenticate using a valid token." });      }      try {          const data = jwt.verify(token, JWT\_SECRET);          req.user = data.user;          next();      } catch (error) {          res.status(401).send({ error: "Please authenticate using a valid token." });      }  }  module.exports = fetchuser; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **53. Updating an existing Note** | **Date: 08/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\notes.js  const express = require('express');  const router = express.Router();  var fetchuser = require('../middleware/fetchuser');  const Note = require('../models/Note');  const { body, validationResult } = require('express-validator');  // ROUTE 1: Get All the Notes using: GET "/api/notes/getuser". Login required  router.get('/fetchallnotes', fetchuser, async (req, res) => {      try {          const notes = await Note.find({ user: req.user.id });          res.json(notes);      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Add a new Note using: POST "/api/notes/addnote". Login required  router.post('/addnote', fetchuser, [      body('title', 'Enter a valid title').isLength({ min: 3 }),      body('description', 'Description atleast must be 5 characters').isLength({ min: 5 }),], async (req, res) => {          try {              const { title, description, tag } = req.body;              // If there are errors, return bad request and the errors              const errors = validationResult(req);              if (!errors.isEmpty()) {                  return res.status(400).json({ errors: errors.array() });              }              const note = new Note({                  title, description, tag, user: req.user.id              })              const savedNote = await note.save();              res.json(savedNote);          } catch (error) {              console.error(error.message);              res.status(500).send("Internal Server Error");          }      })  // ROUTE 3: Update an existing Note using: POST "/api/notes/updatenote". Login required  router.put('/updatenote/:id', fetchuser, async (req, res) =>      {          const {title, description, tag} = req.body;          // create a newNote object          const newNote = {};          if (title){newNote.title = title};          if (description){newNote.description = description};          if (tag){newNote.tag = tag};          // Find the note to be updated and update it.          let note = await Note.findById(req.params.id);          if(!note){return res.status(404).send("Not Found")}          if(note.user.toString() !== req.user.id){              return res.status(404).send("Not Allowed");          }          note = await Note.findByIdAndUpdate(req.params.id, {$set: newNote}, {new:true});          res.json({note});      })    module.exports = router; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **54. Endpoint for deleting a Note** | **Date: 08/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\notes.js  const express = require('express');  const router = express.Router();  var fetchuser = require('../middleware/fetchuser');  const Note = require('../models/Note');  const { body, validationResult } = require('express-validator');  // ROUTE 1: Get All the Notes using: GET "/api/notes/getuser". Login required  router.get('/fetchallnotes', fetchuser, async (req, res) => {      try {          const notes = await Note.find({ user: req.user.id });          res.json(notes);      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Add a new Note using: POST "/api/notes/addnote". Login required  router.post('/addnote', fetchuser, [      body('title', 'Enter a valid title').isLength({ min: 3 }),      body('description', 'Description atleast must be 5 characters').isLength({ min: 5 }),], async (req, res) => {          try {              const { title, description, tag } = req.body;              // If there are errors, return bad request and the errors              const errors = validationResult(req);              if (!errors.isEmpty()) {                  return res.status(400).json({ errors: errors.array() });              }              const note = new Note({                  title, description, tag, user: req.user.id              })              const savedNote = await note.save();              res.json(savedNote);          } catch (error) {              console.error(error.message);              res.status(500).send("Internal Server Error");          }      })  // ROUTE 3: Update an existing Note using: PUT "/api/notes/updatenote". Login required  router.put('/updatenote/:id', fetchuser, async (req, res) => {      try {          const { title, description, tag } = req.body;          // create a newNote object          const newNote = {};          if (title) { newNote.title = title };          if (description) { newNote.description = description };          if (tag) { newNote.tag = tag };          // Find the note to be updated and update it.          let note = await Note.findById(req.params.id);          if (!note) { return res.status(404).send("Not Found") }          if (note.user.toString() !== req.user.id) {              return res.status(404).send("Not Allowed");          }          note = await Note.findByIdAndUpdate(req.params.id, { $set: newNote }, { new: true });          res.json({ note });      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 4: Delete an existing Note using: DELETE "/api/notes/deletenote". Login required  router.delete('/deletenote/:id', fetchuser, async (req, res) => {      try {          // Find the note to be delete and delete it.          let note = await Note.findById(req.params.id);          if (!note) { return res.status(404).send("Not Found") }          // Allow deletion only if user owns this Note          if (note.user.toString() !== req.user.id) {              return res.status(404).send("Not Allowed");          }          note = await Note.findByIdAndDelete(req.params.id);          res.json({ "success": "Note has been deleted", note: note });      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  module.exports = router; |

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **55. iNotebook React Project Setup** | **Date: 08/01/25** |

|  |
| --- |
| **npm i react-router-dom concurrently**  concurrently help you to use 2 server at a time. |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\package.json  {    "name": "inotebook",    "version": "0.1.0",    "private": true,    "dependencies": {      "@testing-library/jest-dom": "^5.17.0",      "@testing-library/react": "^13.4.0",      "@testing-library/user-event": "^13.5.0",      "concurrently": "^9.1.2",      "react": "^19.0.0",      "react-dom": "^19.0.0",      "react-router-dom": "^7.1.1",      "react-scripts": "5.0.1",      "web-vitals": "^2.1.4"    },    "scripts": {      "start": "react-scripts start",      "build": "react-scripts build",      "test": "react-scripts test",      "eject": "react-scripts eject",  **"both": "concurrently \"npm run start\" \"nodemon backend/index.js\""**    },    "eslintConfig": {      "extends": [        "react-app",        "react-app/jest"      ]    },    "browserslist": {      "production": [        ">0.2%",        "not dead",        "not op\_mini all"      ],      "development": [        "last 1 chrome version",        "last 1 firefox version",        "last 1 safari version"      ]    },    "devDependencies": {      "ajv": "^7.2.4"    }  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  function App() {    return (      <div className="App">        <h1>This is iNotebook</h1>      </div>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\index.js  import React from 'react';  import ReactDOM from 'react-dom/client';  import './index.css';  import App from './App';  const root = ReactDOM.createRoot(document.getElementById('root'));  root.render(    <React.StrictMode>      <App />    </React.StrictMode>  ); |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\public\index.html  <!DOCTYPE html>  <html lang="en">    <head>      <meta charset="utf-8" />      <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />      <meta name="viewport" content="width=device-width, initial-scale=1" />      <meta name="theme-color" content="#000000" />      <meta        name="description"        content="Web site created using create-react-app"      />      <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />        <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />        <title>React App</title>    </head>    <body>      <noscript>You need to enable JavaScript to run this app.</noscript>      <div id="root"></div>    </body>  </html> |

|  |  |
| --- | --- |
| **56. Creating Navbar and Routes** | **Date: 14/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\public\index.html  <!DOCTYPE html>  <html lang="en">    <head>      <meta charset="utf-8" />      <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />      <meta name="viewport" content="width=device-width, initial-scale=1" />      <meta name="theme-color" content="#000000" />      <meta        name="description"        content="Web site created using create-react-app"      />      <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />        <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />        <title>iNotebook - Your notes secured in the cloud</title>      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    </head>    <body>      <noscript>You need to enable JavaScript to run this app.</noscript>      <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>    </body>  </html> |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  function App() {    return (      <>        <Router>          <Navbar />          <Routes>            <Route exact path="/" element={<Home />} />            <Route exact path="/about" element={<About />} />          </Routes>        </Router>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import React from 'react'  export const Home = () => {    return (      <div>          <h1>This is iNotebook</h1>      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React from 'react'  const About = () => {    return (      <div>          This is About      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React from 'react'  import { Link } from "react-router-dom";  const Navbar = () => {      return (          <nav className="navbar navbar-expand-lg bg-body-tertiary">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">Navbar</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className="nav-link active" aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className="nav-link" to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\public\index.html  <!DOCTYPE html>  <html lang="en">    <head>      <meta charset="utf-8" />      <link rel="icon" href="%PUBLIC\_URL%/favicon.ico" />      <meta name="viewport" content="width=device-width, initial-scale=1" />      <meta name="theme-color" content="#000000" />      <meta        name="description"        content="Web site created using create-react-app"      />      <link rel="apple-touch-icon" href="%PUBLIC\_URL%/logo192.png" />        <link rel="manifest" href="%PUBLIC\_URL%/manifest.json" />        <title>iNotebook - Your notes secured in the cloud</title>      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-QWTKZyjpPEjISv5WaRU9OFeRpok6YctnYmDr5pNlyT2bRjXh0JMhjY6hW+ALEwIH" crossorigin="anonymous">    </head>    <body>      <noscript>You need to enable JavaScript to run this app.</noscript>      <div id="root"></div>      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-YvpcrYf0tY3lHB60NNkmXc5s9fDVZLESaAA55NDzOxhy9GkcIdslK1eN7N6jIeHz" crossorigin="anonymous"></script>    </body>  </html> |

|  |  |
| --- | --- |
| **57. Introduction to React Context API** | **Date: 15/01/25** |

|  |  |
| --- | --- |
| **58. useContext hook: Using React Context API** | **Date: 16/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <Routes>              <Route exact path="/" element={<Home />} />              <Route exact path="/about" element={<About />} />            </Routes>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React, { useContext, useEffect } from 'react'  import noteContext from '../context/notes/noteContext'  const About = () => {    const a = useContext(noteContext)    useEffect(() => {      a.update();      // eslint-disable-next-line    }, [])      return (      <div>          This is About {a.state.name} and he is in class {a.state.class}      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\noteContext.js  import { createContext } from "react";  const noteContext = createContext();  export default noteContext; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const s1 = {          "name": "Harry",          "class": "5b"      }      const [state, setState] = useState(s1);      const update = () => {          setTimeout(() => {              setState({                  "name": "Larry",                  "class": "10b"              })          }, 1000);      }      return (          <NoteContext.Provider value={{state:state, update:update}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **59. useLocation Hook in React** | **Date: 20/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React, { useContext, useEffect } from 'react'  const About = () => {      return (      <div>          This is About Page      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\noteContext.js  import { createContext } from "react";  const noteContext = createContext();  export default noteContext; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {        return (          <NoteContext.Provider value={{}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React, { useEffect } from 'react'  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      useEffect(() => {        console.log(location.pathname);      }, [location]);      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import React from 'react'  export const Home = () => {    return (      <div>        <div className="container my-3">          <h2>Add a Note</h2>          <form className='my-3'>            <div class="mb-3">              <label for="exampleInputEmail1" class="form-label">Email address</label>              <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp" />              <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>            </div>            <div class="mb-3">              <label for="exampleInputPassword1" class="form-label">Password</label>              <input type="password" class="form-control" id="exampleInputPassword1" />            </div>            <div class="mb-3 form-check">              <input type="checkbox" class="form-check-input" id="exampleCheck1" />              <label class="form-check-label" for="exampleCheck1">Check me out</label>            </div>            <button type="submit" class="btn btn-primary">Submit</button>          </form>        </div>        <div className="container my-3">          <h2>You Note</h2>        </div>      </div>    )  } |

|  |  |
| --- | --- |
| **60. iNotebook: Fetching Notes from Context** | **Date: 20/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  export const Home = () => {    const context = useContext(noteContext);    const {notes, setNotes} = context;    return (      <div>        <div className="container my-3">          <h2>Add a Note</h2>          <form className='my-3'>            <div class="mb-3">              <label for="exampleInputEmail1" class="form-label">Email address</label>              <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp" />              <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>            </div>            <div class="mb-3">              <label for="exampleInputPassword1" class="form-label">Password</label>              <input type="password" class="form-control" id="exampleInputPassword1" />            </div>            <div class="mb-3 form-check">              <input type="checkbox" class="form-check-input" id="exampleCheck1" />              <label class="form-check-label" for="exampleCheck1">Check me out</label>            </div>            <button type="submit" class="btn btn-primary">Submit</button>          </form>        </div>        <div className="container my-3">          <h2>You Note</h2>          {notes.map((note)=>{            return note.title;          })}        </div>      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\noteContext.js  import { createContext } from "react";  const noteContext = createContext();  export default noteContext; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React, { useEffect } from 'react'  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      useEffect(() => {        console.log(location.pathname);      }, [location]);      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const notesInitial = [          {            "\_id": "677e8e67631f92e1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          }        ]        const [notes, setNotes] = useState(notesInitial)      return (          <NoteContext.Provider value={{notes, setNotes}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **61. iNotebook: Adding NoteItem in a Separate Note component** | **Date: 21/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = () => {    return (      <div>        <div className="container my-3">          <h2>Add a Note</h2>          <form className='my-3'>            <div class="mb-3">              <label for="exampleInputEmail1" class="form-label">Email address</label>              <input type="email" class="form-control" id="exampleInputEmail1" aria-describedby="emailHelp" />              <div id="emailHelp" class="form-text">We'll never share your email with anyone else.</div>            </div>            <div class="mb-3">              <label for="exampleInputPassword1" class="form-label">Password</label>              <input type="password" class="form-control" id="exampleInputPassword1" />            </div>            <div class="mb-3 form-check">              <input type="checkbox" class="form-check-input" id="exampleCheck1" />              <label class="form-check-label" for="exampleCheck1">Check me out</label>            </div>            <button type="submit" class="btn btn-primary">Submit</button>          </form>        </div>        <Notes />      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  const Notes = () => {      const context = useContext(noteContext);      const { notes, setNotes } = context;      return (          <div className="row my-3">              <h2>You Note</h2>              {notes.map((note) => {                  return <Noteitem note={note} />              })}          </div>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React from 'react'  const Noteitem = (props) => {      const { note } = props;      return (          <div className='col-md-3'>              <div class="card my-3">                  <div class="card-body">                      <h5 class="card-title">{note.title}</h5>                      <p class="card-text">{note.description} Lorem ipsum, dolor sit amet consectetur adipisicing elit. Illum non iste nemo accusamus eaque eos ex. Porro harum aut laboriosam voluptatum, eum adipisci cupiditate reprehenderit, sed corrupti atque praesentium non libero vitae nostrum consequatur.</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\noteContext.js  import { createContext } from "react";  const noteContext = createContext();  export default noteContext; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React, { useEffect } from 'react'  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      useEffect(() => {        console.log(location.pathname);      }, [location]);      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const notesInitial = [          {            "\_id": "677e8e67631f92e1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92e1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          }          ]        const [notes, setNotes] = useState(notesInitial)      return (          <NoteContext.Provider value={{notes, setNotes}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **62. Adding font awesome icons to iNotebook** | **Date: 21/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  import { Alert } from './components/Alert';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <Alert message="This is amazing React course" />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = () => {    return (      <div>        <div className="container my-3">          <h2>Add a Note</h2>          <form className='my-3'>            <div className="mb-3">              <label htmlFor="exampleInputEmail1" className="form-label">Email address</label>              <input type="email" className="form-control" id="exampleInputEmail1" aria-describedby="emailHelp" />              <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>            </div>            <div className="mb-3">              <label htmlFor="exampleInputPassword1" className="form-label">Password</label>              <input type="password" className="form-control" id="exampleInputPassword1" />            </div>            <div className="mb-3 form-check">              <input type="checkbox" className="form-check-input" id="exampleCheck1" />              <label className="form-check-label" htmlFor="exampleCheck1">Check me out</label>            </div>            <button type="submit" className="btn btn-primary">Submit</button>          </form>        </div>        <Notes />      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Alert.js  import React from 'react'  export const Alert = (props) => {      return (          <div>              <div class="alert alert-primary" role="alert">                  {props.message}              </div>          </div>      )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  const Notes = () => {      const context = useContext(noteContext);      const { notes, setNotes } = context;      return (          <div className="row my-3">              <h2>You Note</h2>              {notes.map((note) => {                  return <Noteitem key={note.\_id} note={note} />              })}          </div>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React from 'react'  const Noteitem = (props) => {      const { note } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2"></i>                          <i className="fa-solid fa-pen-to-square mx-2"></i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\noteContext.js  import { createContext } from "react";  const noteContext = createContext();  export default noteContext; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React, { useEffect } from 'react'  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      useEffect(() => {        console.log(location.pathname);      }, [location]);      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const notesInitial = [          {            "\_id": "677e8e675631f92e1de1e2d7e",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e676361f92e1de1e2d80",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e41de1e2d83",            "user": "6778de460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c98434e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92ek1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f9l2e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e961663s1f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834ce2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          }          ]        const [notes, setNotes] = useState(notesInitial)      return (          <NoteContext.Provider value={{notes, setNotes}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **63. Adding AddNote component to iNotebook** | **Date: 22/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  import { Alert } from './components/Alert';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <Alert message="This is amazing React course" />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React, { useContext, useEffect } from 'react'  const About = () => {      return (      <div>          This is About Page      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = () => {    return (      <div>        <Notes />      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = () => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: "default"})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" aria-describedby="emailHelp" onChange={onChange} />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" onChange={onChange}/>                  </div>                  <div className="mb-3 form-check">                      <input type="checkbox" className="form-check-input" id="exampleCheck1" />                      <label className="form-check-label" htmlFor="exampleCheck1">Check me out</label>                  </div>                  <button type="submit" className="btn btn-primary" onClick={handleClick}>Submit</button>              </form>          </div>      )  }  export default AddNote |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes, addNote } = context;      return (          <>              <AddNote />              <div className="row my-3">                  <h2>You Note</h2>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React from 'react'  const Noteitem = (props) => {      const { note } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2"></i>                          <i className="fa-solid fa-pen-to-square mx-2"></i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const notesInitial = [          {            "\_id": "677e8e675631f92e1de1e2d7e",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e676361f92e1de1e2d80",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e41de1e2d83",            "user": "6778de460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c98434e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92ek1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f9l2e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e961663s1f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834ce2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          }          ]        const [notes, setNotes] = useState(notesInitial)        // Add a Note        const addNote = (title, description, tag) => {          // TODO: API Call          console.log("Adding a new note")          const note = {            "\_id": "677ea5c9834ce2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": title,            "description": description,            "tag": tag,            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          };          setNotes(notes.concat(note))        }        // Delete a Note        const deleteNote = (id) => {          }        // Edit a Note        const editNote = (id) => {          }      return (          <NoteContext.Provider value={{notes, addNote, deleteNote, editNote}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **64. Adding "delete note" functionality to iNotebook** | **Date: 24/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = () => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: "default"})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" aria-describedby="emailHelp" onChange={onChange} />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" onChange={onChange}/>                  </div>                  <div className="mb-3 form-check">                      <input type="checkbox" className="form-check-input" id="exampleCheck1" />                      <label className="form-check-label" htmlFor="exampleCheck1">Check me out</label>                  </div>                  <button type="submit" className="btn btn-primary" onClick={handleClick}>Add Note</button>              </form>          </div>      )  }  export default AddNote |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes } = context;      return (          <>              <AddNote />              <div className="row my-3">                  <h2>You Note</h2>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  const Noteitem = (props) => {      const context = useContext(noteContext);      const { deleteNote } = context;      const { note } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2" onClick={()=>{deleteNote(note.\_id)}}></i>                          <i className="fa-solid fa-pen-to-square mx-2"></i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {      const notesInitial = [          {            "\_id": "677e8e675631f92e1de1e2d7e",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e676361f92e1de1e2d80",            "user": "6778e460590f7cddddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e9616631f92e41de1e2d83",            "user": "6778de460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c98434e2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f92ek1de1e2d7e",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.200Z",            "\_\_v": 0          },          {            "\_id": "677e8e67631f9l2e1de1e2d80",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Title",            "description": "Please wake-up early",            "tag": "personal",            "date": "2025-01-08T14:40:39.719Z",            "\_\_v": 0          },          {            "\_id": "677e961663s1f92e1de1e2d83",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T15:13:26.837Z",            "\_\_v": 0          },          {            "\_id": "677ea5c9834ce2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": "My Notes",            "description": "Please enjoy your life",            "tag": "YouTube",            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          }          ]        const [notes, setNotes] = useState(notesInitial)        // Add a Note        const addNote = (title, description, tag) => {          // TODO: API Call          console.log("Adding a new note")          const note = {            "\_id": "677ea5c9834ce2ebbfee4fbaf",            "user": "6778e460590f7cdddc6ddc5c",            "title": title,            "description": description,            "tag": tag,            "date": "2025-01-08T16:20:25.619Z",            "\_\_v": 0          };          setNotes(notes.concat(note))        }        // Delete a Note        const deleteNote = (id) => {          // TODO: API Call          console.log("Delete the note with id " + id);          const newNotes = notes.filter((note)=>{return note.\_id !== id})          setNotes(newNotes)        }        // Edit a Note        const editNote = (id, title, description, tag) => {          }      return (          <NoteContext.Provider value={{notes, addNote, deleteNote, editNote}}>              {props.children}          </NoteContext.Provider>      );  }  export default NoteState; |

|  |  |
| --- | --- |
| **65. Adding "fetch notes" functionality to iNotebook** | **Date: 24/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  import { Alert } from './components/Alert';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <Alert message="This is amazing React course" />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React from 'react'  const About = () => {      return (      <div>          This is About Page      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React from 'react';  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = () => {    return (      <div>        <Notes />      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = () => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: "default"})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" aria-describedby="emailHelp" onChange={onChange} />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" onChange={onChange}/>                  </div>                  <div className="mb-3 form-check">                      <input type="checkbox" className="form-check-input" id="exampleCheck1" />                      <label className="form-check-label" htmlFor="exampleCheck1">Check me out</label>                  </div>                  <button type="submit" className="btn btn-primary" onClick={handleClick}>Add Note</button>              </form>          </div>      )  }  export default AddNote |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes, getNotes } = context;      useEffect(() => {          getNotes()      }, [])        return (          <>              <AddNote />              <div className="row my-3">                  <h2>You Note</h2>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  const Noteitem = (props) => {      const context = useContext(noteContext);      const { deleteNote } = context;      const { note } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2" onClick={()=>{deleteNote(note.\_id)}}></i>                          <i className="fa-solid fa-pen-to-square mx-2"></i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          'auth-token': 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o',        }      });      const json = await response.json();      console.log(json);      setNotes(json);    }    // Add a Note (\*)    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      // const response = await fetch(`${host}/api/notes/addnote`, {      //   method: 'POST',      //   headers: {      //     'Content-Type': 'application/json',      //     'auth-token': 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o',      //   },      //   body: JSON.stringify(title, description, tag)      // });      console.log("Adding a new note")      const note = {        "\_id": "677ea5c9834ce2ebbfee4fbaf",        "user": "6778e460590f7cdddc6ddc5c",        "title": title,        "description": description,        "tag": tag,        "date": "2025-01-08T16:20:25.619Z",        "\_\_v": 0      };      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = (id) => {      // TODO: API Call      console.log("Delete the note with id " + id);      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          'auth-token': 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o',        },        body: JSON.stringify({ title, description, tag })      });      const json = response.json();      // Logic to edit in client      for (let index = 0; index < notes.length; index++) {        const element = notes[index];        if (element.\_id === id) {          element.title = title;          element.description = description;          element.tag = tag;        }      }    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    );  }  export default NoteState; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\index.js  const connectToMongo = require('./db');  const express = require('express')  var cors = require('cors')  connectToMongo();  const app = express()  const port = 5000  app.use(cors())  app.use(express.json())  // Available Routes  app.use('/api/auth', require('./routes/auth'))  app.use('/api/notes', require('./routes/notes'))  app.listen(port, () => {    console.log(`iNotebook backend listening at http://localhost:${port}`)  }) |

|  |  |
| --- | --- |
| **66. Adding a Modal for Editing Notes** | **Date: 24/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  import { Alert } from './components/Alert';  function App() {    return (      <>        <NoteState>          <Router>            <Navbar />            <Alert message="This is amazing React course" />            <div className="container">              <Routes>                <Route exact path="/" element={<Home />} />                <Route exact path="/about" element={<About />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React from 'react'  const About = () => {      return (      <div>          This is About Page      </div>    )  }  export default About |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React from 'react';  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/"? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname=="/about"? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <input className="form-control me-2" type="search" placeholder="Search" aria-label="Search" />                          <button className="btn btn-outline-success" type="submit">Search</button>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = () => {    return (      <div>        <Notes />      </div>    )  } |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = () => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: "default"})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" onChange={onChange} />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" onChange={onChange}/>                  </div>                  <div className="mb-3">                      <label htmlFor="tag" className="form-label">Tag</label>                      <input type="text" className="form-control" id="tag" name="tag" onChange={onChange}/>                  </div>                  <button type="submit" className="btn btn-primary" onClick={handleClick}>Add Note</button>              </form>          </div>      )  }  export default AddNote |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect, useRef, useState } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes, getNotes } = context;      useEffect(() => {          getNotes()          // eslint-disable-next-line      }, [])      const ref = useRef(null)      const [note, setNote] = useState({ etitle: "", edescription: "", etag: "" })      const updateNote = (currentNote) => {          ref.current.click();          setNote({etitle: currentNote.title, edescription: currentNote.description, etag: currentNote.tag});      }      const handleClick = (e) => {          console.log("Updating the note...", note);          e.preventDefault();      }      const onChange = (e) => {          setNote({ ...note, [e.target.name]: e.target.value })      }      return (          <>              <AddNote />              <button ref={ref} type="button" className="btn btn-primary d-none" data-bs-toggle="modal" data-bs-target="#exampleModal">                  Launch demo modal              </button>              <div className="modal fade" id="exampleModal" tabIndex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">                  <div className="modal-dialog">                      <div className="modal-content">                          <div className="modal-header">                              <h1 className="modal-title fs-5" id="exampleModalLabel">Edit Note</h1>                              <button type="button" className="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>                          </div>                          <div className="modal-body">                              <form className='my-3'>                                  <div className="mb-3">                                      <label htmlFor="etitle" className="form-label">Title</label>                                      <input type="text" className="form-control" id="etitle" name="etitle" value={note.etitle} onChange={onChange} />                                  </div>                                  <div className="mb-3">                                      <label htmlFor="edescription" className="form-label">Description</label>                                      <input type="text" className="form-control" id="edescription" name="edescription" value={note.edescription} onChange={onChange} />                                  </div>                                  <div className="mb-3">                                      <label htmlFor="etag" className="form-label">Tag</label>                                      <input type="text" className="form-control" id="etag" name="etag" value={note.etag} onChange={onChange} />                                  </div>                              </form>                          </div>                          <div className="modal-footer">                              <button type="button" className="btn btn-secondary" data-bs-dismiss="modal">Close</button>                              <button onClick={handleClick} type="button" className="btn btn-primary">Update Note</button>                          </div>                      </div>                  </div>              </div>              <div className="row my-3">                  <h2>You Note</h2>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} updateNote={updateNote} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  const Noteitem = (props) => {      const context = useContext(noteContext);      const { deleteNote } = context;      const { note, updateNote } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2" onClick={()=>{deleteNote(note.\_id)}}></i>                          <i className="fa-solid fa-pen-to-square mx-2" onClick={()=>{updateNote(note)}}></i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = await response.json()      console.log(json)      setNotes(json)    }    // Add a Note    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      const response = await fetch(`${host}/api/notes/addnote`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });        console.log("Adding a new note")      const note = {        "\_id": "61322f119553781a8ca8d0e08",        "user": "6131dc5e3e4037cd4734a0664",        "title": title,        "description": description,        "tag": tag,        "date": "2021-09-03T14:20:09.668Z",        "\_\_v": 0      };      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = async (id) => {      // API Call      const response = await fetch(`${host}/api/notes/deletenote/${id}`, {        method: 'DELETE',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = response.json();      console.log(json)      console.log("Deleting the note with id" + id);      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });      const json = response.json();      // Logic to edit in client      for (let index = 0; index < notes.length; index++) {        const element = notes[index];        if (element.\_id === id) {          element.title = title;          element.description = description;          element.tag = tag;        }      }    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    )  }  export default NoteState; |

|  |  |
| --- | --- |
| **67. Updating Notes on edit in the UI using React** | **Date: 27/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = await response.json()      console.log(json)      setNotes(json)    }    // Add a Note    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      const response = await fetch(`${host}/api/notes/addnote`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });        const json = await response.json();      console.log(json);      console.log("Adding a new note")      const note = {        "\_id": "61322f119553781a8ca8d0e08",        "user": "6131dc5e3e4037cd4734a0664",        "title": title,        "description": description,        "tag": tag,        "date": "2021-09-03T14:20:09.668Z",        "\_\_v": 0      };      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = async (id) => {      // API Call      const response = await fetch(`${host}/api/notes/deletenote/${id}`, {        method: 'DELETE',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = response.json();      console.log(json)      console.log("Deleting the note with id" + id);      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'PUT',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });      const json = response.json();      console.log(json);      let newNotes = JSON.parse(JSON.stringify(notes))      // Logic to edit in client      for (let index = 0; index < newNotes.length; index++) {        const element = newNotes[index];        if (element.\_id === id) {          newNotes[index].title = title;          newNotes[index].description = description;          newNotes[index].tag = tag;          break;        }      }      console.log(id, newNotes);      setNotes(newNotes);    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    )  }  export default NoteState; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect, useRef, useState } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes, getNotes, editNote } = context;      useEffect(() => {          getNotes()          // eslint-disable-next-line      }, [])      const ref = useRef(null)      const refClose = useRef(null)      const [note, setNote] = useState({id: "", etitle: "", edescription: "", etag: "" })      const updateNote = (currentNote) => {          ref.current.click();          setNote({id: currentNote.\_id, etitle: currentNote.title, edescription: currentNote.description, etag: currentNote.tag});      }      const handleClick = (e) => {          editNote(note.id, note.etitle, note.edescription, note.etag);          refClose.current.click();      }      const onChange = (e) => {          setNote({ ...note, [e.target.name]: e.target.value })      }      return (          <>              <AddNote />              <button ref={ref} type="button" className="btn btn-primary d-none" data-bs-toggle="modal" data-bs-target="#exampleModal">                  Launch demo modal              </button>              <div className="modal fade" id="exampleModal" tabIndex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">                  <div className="modal-dialog">                      <div className="modal-content">                          <div className="modal-header">                              <h1 className="modal-title fs-5" id="exampleModalLabel">Edit Note</h1>                              <button type="button" className="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>                          </div>                          <div className="modal-body">                              <form className='my-3'>                                  <div className="mb-3">                                      <label htmlFor="etitle" className="form-label">Title</label>                                      <input type="text" className="form-control" id="etitle" name="etitle" value={note.etitle} onChange={onChange} />                                  </div>                                  <div className="mb-3">                                      <label htmlFor="edescription" className="form-label">Description</label>                                      <input type="text" className="form-control" id="edescription" name="edescription" value={note.edescription} onChange={onChange} />                                  </div>                                  <div className="mb-3">                                      <label htmlFor="etag" className="form-label">Tag</label>                                      <input type="text" className="form-control" id="etag" name="etag" value={note.etag} onChange={onChange} />                                  </div>                              </form>                          </div>                          <div className="modal-footer">                              <button ref={refClose} type="button" className="btn btn-secondary" data-bs-dismiss="modal">Close</button>                              <button onClick={handleClick} type="button" className="btn btn-primary">Update Note</button>                          </div>                      </div>                  </div>              </div>              <div className="row my-3">                  <h2>You Note</h2>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} updateNote={updateNote} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |  |
| --- | --- |
| **68. Adding Frontend Validation + Few Fixes in iNotebook** | **Date: 27/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = await response.json();      setNotes(json)    }    // Add a Note    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      const response = await fetch(`${host}/api/notes/addnote`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });        const note = await response.json();      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = async (id) => {      // API Call      const response = await fetch(`${host}/api/notes/deletenote/${id}`, {        method: 'DELETE',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = response.json();      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'PUT',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });      const json = response.json();      let newNotes = JSON.parse(JSON.stringify(notes))      // Logic to edit in client      for (let index = 0; index < newNotes.length; index++) {        const element = newNotes[index];        if (element.\_id === id) {          newNotes[index].title = title;          newNotes[index].description = description;          newNotes[index].tag = tag;          break;        }      }      setNotes(newNotes);    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    )  }  export default NoteState; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect, useRef, useState } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = () => {      const context = useContext(noteContext);      const { notes, getNotes, editNote } = context;      useEffect(() => {          getNotes()          // eslint-disable-next-line      }, [])      const ref = useRef(null)      const refClose = useRef(null)      const [note, setNote] = useState({ id: "", etitle: "", edescription: "", etag: "" })      const updateNote = (currentNote) => {          ref.current.click();          setNote({ id: currentNote.\_id, etitle: currentNote.title, edescription: currentNote.description, etag: currentNote.tag });      }      const handleClick = (e) => {          editNote(note.id, note.etitle, note.edescription, note.etag);          refClose.current.click();      }      const onChange = (e) => {          setNote({ ...note, [e.target.name]: e.target.value })      }      return (          <>              <AddNote />              <button ref={ref} type="button" className="btn btn-primary d-none" data-bs-toggle="modal" data-bs-target="#exampleModal">                  Launch demo modal              </button>              <div className="modal fade" id="exampleModal" tabIndex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">                  <div className="modal-dialog">                      <div className="modal-content">                          <div className="modal-header">                              <h1 className="modal-title fs-5" id="exampleModalLabel">Edit Note</h1>                              <button type="button" className="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>                          </div>                          <div className="modal-body">                              <form className='my-3'>                                  <div className="mb-3">                                      <label htmlFor="etitle" className="form-label">Title</label>                                      <input type="text" className="form-control" id="etitle" name="etitle" value={note.etitle} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="edescription" className="form-label">Description</label>                                      <input type="text" className="form-control" id="edescription" name="edescription" value={note.edescription} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="etag" className="form-label">Tag</label>                                      <input type="text" className="form-control" id="etag" name="etag" value={note.etag} onChange={onChange} minLength={5} required />                                  </div>                              </form>                          </div>                          <div className="modal-footer">                              <button ref={refClose} type="button" className="btn btn-secondary" data-bs-dismiss="modal">Close</button>                              <button disabled={note.etitle.length<5 || note.edescription.length<5 } onClick={handleClick} type="button" className="btn btn-primary">Update Note</button>                          </div>                      </div>                  </div>              </div>              <div className="row my-3">                  <h2>You Note</h2>                  <div className="container mx-2">                      {notes.length === 0 && 'No Notes to Display'}                  </div>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} updateNote={updateNote} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = () => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: ""})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);          setNote({title: "", description: "", tag: ""});      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" value={note.title} onChange={onChange} minLength={5} required />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" value={note.description} onChange={onChange} minLength={5} required />                  </div>                  <div className="mb-3">                      <label htmlFor="tag" className="form-label">Tag</label>                      <input type="text" className="form-control" id="tag" name="tag" value={note.tag} onChange={onChange} minLength={5} required/>                  </div>                  <button disabled={note.title.length<5 || note.description.length<5 } type="submit" className="btn btn-primary" onClick={handleClick}>Add Note</button>              </form>          </div>      )  }  export default AddNote |

|  |  |
| --- | --- |
| **69. Adding Login Component to iNotebook** | **Date: 27/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Login.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Login = () => {      const [credentials, setCredentials] = useState({ email: "", password: "" })      let navigate = useNavigate();        const handleSubmit = async (e) => {          e.preventDefault();          const response = await fetch("http://localhost:5000/api/auth/login", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({email: credentials.email, password: credentials.password})          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");          } else {              alert("Invalid credentials");          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email address</label>                      <input type="email" className="form-control" value={credentials.email} onChange={onChange} id="email" name="email" aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" value={credentials.password} onChange={onChange} id="password" name="password" />                  </div>                  <button type="submit" className="btn btn-primary">Login</button>              </form>          </div>      )  }  export default Login |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React from 'react';  import { Link, useLocation } from "react-router-dom";  const Navbar = () => {      let location = useLocation();      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname == "/" ? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname == "/about" ? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      <form className="d-flex" role="search">                          <Link className="btn btn-primary mx-1" to="/login" role="button">Login</Link>                          <Link className="btn btn-primary mx-1" to="/signup" role="button">Sign-Up</Link>                      </form>                  </div>              </div>          </nav>      )  }  export default Navbar |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  var fetchuser = require('../middleware/fetchuser');  const JWT\_SECRET = 'Harryisagoodb$oy';  // ROUTE 1: Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          res.json({authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Authenticate the User using: POST "/api/auth/login". No login required  router.post('/login', [      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password cannot be blank').exists(),  ], async (req, res) => {      let success = false;      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          success = false;          return res.status(400).json({ success, errors: errors.array() });      }      const {email, password} = req.body;      try {          let user = await User.findOne({email});          if (!user){              success = false;              return res.status(404).json({ success, errors: "Please to try to login with correct Credentials" });          }          const passwordCompare = await bcrypt.compare(password, user.password);          if (!passwordCompare){              success = false;              return res.status(404).json({ success, errors: "Please to try to login with correct Credentials" });          }          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          success = true;          res.json({success, authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 3: Get loggedin User Details using: POST "/api/auth/getuser". Login required  router.post('/getuser', fetchuser, async (req, res) => {  try {      userId = req.user.id;      const user = await User.findById(userId).select("-password");      res.send(user);  } catch (error) {      console.error(error.message);          res.status(500).send("Internal Server Error");  }  })  module.exports = router; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = await response.json();      setNotes(json)    }    // Add a Note    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      const response = await fetch(`${host}/api/notes/addnote`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });        const note = await response.json();      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = async (id) => {      // API Call      const response = await fetch(`${host}/api/notes/deletenote/${id}`, {        method: 'DELETE',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        }      });      const json = response.json();      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'PUT',        headers: {          'Content-Type': 'application/json',          "auth-token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VyIjp7ImlkIjoiNjc3OGU0NjA1OTBmN2NkZGRjNmRkYzVjIn0sImlhdCI6MTczNjE4MDAzNX0.AVCIDk33tmKayuCCzFNam3NufaSpTFdPY47z6miMf9o"        },        body: JSON.stringify({title, description, tag})      });      const json = response.json();      let newNotes = JSON.parse(JSON.stringify(notes))      // Logic to edit in client      for (let index = 0; index < newNotes.length; index++) {        const element = newNotes[index];        if (element.\_id === id) {          newNotes[index].title = title;          newNotes[index].description = description;          newNotes[index].tag = tag;          break;        }      }      setNotes(newNotes);    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    )  }  export default NoteState; |

|  |  |
| --- | --- |
| **70. Adding SignUp Component to iNotebook** | **Date: 27/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Signup.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Signup = () => {      const [credentials, setCredentials] = useState({ name: "", email: "", password: "", cpassword: "" })      let navigate = useNavigate();      const handleSubmit = async (e) => {          e.preventDefault();          const { name, email, password } = credentials;          const response = await fetch("http://localhost:5000/api/auth/createuser", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({ name, email, password })          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");          } else {              alert("Invalid credentials");          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div className='container'>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="name" className="form-label">Name</label>                      <input type="text" className="form-control" id="name" onChange={onChange} name='name' aria-describedby="name" />                  </div>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email</label>                      <input type="email" className="form-control" id="email" onChange={onChange} name='email' aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" id="password" onChange={onChange} name='password' minLength={5} required/>                  </div>                  <div className="mb-3">                      <label htmlFor="cpassword" className="form-label">Confirm Password</label>                      <input type="password" className="form-control" id="cpassword" onChange={onChange} name='cpassword' minLength={5} required/>                  </div>                  <button type="submit" className="btn btn-primary">Sign-Up</button>              </form>          </div>      )  }  export default Signup |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Login.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Login = () => {      const [credentials, setCredentials] = useState({ email: "", password: "" })      let navigate = useNavigate();        const handleSubmit = async (e) => {          e.preventDefault();          const response = await fetch("http://localhost:5000/api/auth/login", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({email: credentials.email, password: credentials.password})          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");          } else {              alert("Invalid credentials");          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email address</label>                      <input type="email" className="form-control" value={credentials.email} onChange={onChange} id="email" name="email" aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" value={credentials.password} onChange={onChange} id="password" name="password" />                  </div>                  <button type="submit" className="btn btn-primary">Login</button>              </form>          </div>      )  }  export default Login |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\backend\routes\auth.js  const express = require('express');  const User = require('../models/User');  const router = express.Router();  const { body, validationResult } = require('express-validator');  const bcrypt = require('bcryptjs');  var jwt = require('jsonwebtoken');  var fetchuser = require('../middleware/fetchuser');  const JWT\_SECRET = 'Harryisagoodb$oy';  // ROUTE 1: Create a User using: POST "/api/auth/createuser". No login required  router.post('/createuser', [      body('name', 'Enter a valid name').isLength({ min: 3 }),      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password atleast must be 5 characters').isLength({ min: 5 }),  ], async (req, res) => {      let success = false;      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          return res.status(400).json({ success, errors: errors.array() });      }      try {          // Check whether the user with this email exists already          let user = await User.findOne({ email: req.body.email });          if (user) {              return res.status(400).json({ success, error: "Sorry a user with this email already exists" })          }          const salt = await bcrypt.genSalt(10);          const secPass = await bcrypt.hash(req.body.password, salt);          // Create a new user          user = await User.create({              name: req.body.name,              password: secPass,              email: req.body.email,          });          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          // res.json(user)          success = true;          res.json({ success, authtoken })      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 2: Authenticate the User using: POST "/api/auth/login". No login required  router.post('/login', [      body('email', 'Enter a valid email').isEmail(),      body('password', 'Password cannot be blank').exists(),  ], async (req, res) => {      let success = false;      // If there are errors, return bad request and the errors      const errors = validationResult(req);      if (!errors.isEmpty()) {          success = false;          return res.status(400).json({ success, errors: errors.array() });      }      const {email, password} = req.body;      try {          let user = await User.findOne({email});          if (!user){              success = false;              return res.status(404).json({ success, errors: "Please to try to login with correct Credentials" });          }          const passwordCompare = await bcrypt.compare(password, user.password);          if (!passwordCompare){              success = false;              return res.status(404).json({ success, errors: "Please to try to login with correct Credentials" });          }          const data = {              user: {                  id: user.id              }          }          const authtoken = jwt.sign(data, JWT\_SECRET);          success = true;          res.json({success, authtoken})      } catch (error) {          console.error(error.message);          res.status(500).send("Internal Server Error");      }  })  // ROUTE 3: Get loggedin User Details using: POST "/api/auth/getuser". Login required  router.post('/getuser', fetchuser, async (req, res) => {  try {      userId = req.user.id;      const user = await User.findById(userId).select("-password");      res.send(user);  } catch (error) {      console.error(error.message);          res.status(500).send("Internal Server Error");  }  })  module.exports = router; |

|  |  |
| --- | --- |
| **71. Adding Alerts to Login, Signup & Notes Component of iNotebook** | **Date: 27/01/25** |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Signup.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Signup = (props) => {      const [credentials, setCredentials] = useState({ name: "", email: "", password: "", cpassword: "" })      let navigate = useNavigate();      const handleSubmit = async (e) => {          e.preventDefault();          const { name, email, password } = credentials;          const response = await fetch("http://localhost:5000/api/auth/createuser", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({ name, email, password })          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");              props.showAlert("Account Created Successfully", "success");          } else {              props.showAlert("Invalid Details", "danger")          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div className='container'>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="name" className="form-label">Name</label>                      <input type="text" className="form-control" id="name" onChange={onChange} name='name' aria-describedby="name" />                  </div>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email</label>                      <input type="email" className="form-control" id="email" onChange={onChange} name='email' aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" id="password" onChange={onChange} name='password' minLength={5} required/>                  </div>                  <div className="mb-3">                      <label htmlFor="cpassword" className="form-label">Confirm Password</label>                      <input type="password" className="form-control" id="cpassword" onChange={onChange} name='cpassword' minLength={5} required/>                  </div>                  <button type="submit" className="btn btn-primary">Sign-Up</button>              </form>          </div>      )  }  export default Signup |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Login.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Login = (props) => {      const [credentials, setCredentials] = useState({ email: "", password: "" })      let navigate = useNavigate();        const handleSubmit = async (e) => {          e.preventDefault();          const response = await fetch("http://localhost:5000/api/auth/login", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({email: credentials.email, password: credentials.password})          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");              props.showAlert("Logged In Successfully", "success");          } else {              props.showAlert("Invalid Credentials", "danger");          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email address</label>                      <input type="email" className="form-control" value={credentials.email} onChange={onChange} id="email" name="email" aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" value={credentials.password} onChange={onChange} id="password" name="password" />                  </div>                  <button type="submit" className="btn btn-primary">Login</button>              </form>          </div>      )  }  export default Login |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\App.js  import './App.css';  import {    BrowserRouter as Router,    Routes,    Route  } from "react-router-dom";  import Navbar from './components/Navbar';  import { Home } from './components/Home';  import About from './components/About';  import NoteState from './context/notes/NoteState';  import Alert from './components/Alert';  import Signup from './components/Signup';  import Login from './components/Login';  import { useState } from 'react';  function App() {    const [alert, setAlert] = useState(null);    const showAlert = (message, type) => {      setAlert({        msg: message,        type: type      })      setTimeout(() => {        setAlert(null);      }, 1500);    }    return (      <>        <NoteState>          <Router>            <Navbar />            <Alert alert={alert} />            <div className="container">              <Routes>                <Route exact path="/" element={<Home showAlert={showAlert} />} />                <Route exact path="/about" element={<About />} />                <Route exact path="/login" element={<Login showAlert={showAlert} />} />                <Route exact path="/signup" element={<Signup showAlert={showAlert} />} />              </Routes>            </div>          </Router>        </NoteState>      </>    );  }  export default App; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Alert.js  import React from 'react'  function Alert(props) {      const capitalize = (word)=>{          if (word === 'danger') {              word = 'error';          }          const lower = word.toLowerCase();          return lower.charAt(0).toUpperCase() + lower.slice(1);      }      return (          <div style={{height: '50px'}}>          {props.alert && <div className={`alert alert-${props.alert.type} alert-dismissible fade show`} role="alert">             <strong>{capitalize(props.alert.type)}</strong>: {props.alert.msg}          </div>}          </div>      )  }  export default Alert |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Noteitem.js  import React, { useContext } from 'react';  import noteContext from '../context/notes/noteContext';  const Noteitem = (props) => {      const context = useContext(noteContext);      const { deleteNote } = context;      const { note, updateNote } = props;      return (          <div className='col-md-3'>              <div className="card my-3">                  <div className="card-body">                      <div className="d-flex align-items-center">                          <h5 className="card-title">{note.title}</h5>                          <i className="fa-solid fa-trash mx-2"                              onClick={() => {                                  deleteNote(note.\_id);                                  props.showAlert("Delete Successfully", "success");                              }}>                          </i>                          <i className="fa-solid fa-pen-to-square mx-2"                              onClick={() => {                                  updateNote(note);                              }}>                          </i>                      </div>                      <p className="card-text">{note.description}</p>                  </div>              </div>          </div>      )  }  export default Noteitem |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect, useRef, useState } from 'react';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = (props) => {      const context = useContext(noteContext);      const { notes, getNotes, editNote } = context;      useEffect(() => {          getNotes()          // eslint-disable-next-line      }, [])      const ref = useRef(null)      const refClose = useRef(null)      const [note, setNote] = useState({ id: "", etitle: "", edescription: "", etag: "" })      const updateNote = (currentNote) => {          ref.current.click();          setNote({ id: currentNote.\_id, etitle: currentNote.title, edescription: currentNote.description, etag: currentNote.tag });      }      const handleClick = (e) => {          editNote(note.id, note.etitle, note.edescription, note.etag);          refClose.current.click();          props.showAlert("Update Successfully", "success");      }      const onChange = (e) => {          setNote({ ...note, [e.target.name]: e.target.value })      }      return (          <>              <AddNote showAlert={props.showAlert} />              <button ref={ref} type="button" className="btn btn-primary d-none" data-bs-toggle="modal" data-bs-target="#exampleModal">                  Launch demo modal              </button>              <div className="modal fade" id="exampleModal" tabIndex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">                  <div className="modal-dialog">                      <div className="modal-content">                          <div className="modal-header">                              <h1 className="modal-title fs-5" id="exampleModalLabel">Edit Note</h1>                              <button type="button" className="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>                          </div>                          <div className="modal-body">                              <form className='my-3'>                                  <div className="mb-3">                                      <label htmlFor="etitle" className="form-label">Title</label>                                      <input type="text" className="form-control" id="etitle" name="etitle" value={note.etitle} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="edescription" className="form-label">Description</label>                                      <input type="text" className="form-control" id="edescription" name="edescription" value={note.edescription} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="etag" className="form-label">Tag</label>                                      <input type="text" className="form-control" id="etag" name="etag" value={note.etag} onChange={onChange} minLength={5} required />                                  </div>                              </form>                          </div>                          <div className="modal-footer">                              <button ref={refClose} type="button" className="btn btn-secondary" data-bs-dismiss="modal">Close</button>                              <button disabled={note.etitle.length<5 || note.edescription.length<5 } onClick={handleClick} type="button" className="btn btn-primary">Update Note</button>                          </div>                      </div>                  </div>              </div>              <div className="row my-3">                  <h2>You Note</h2>                  <div className="container mx-2">                      {notes.length === 0 && 'No Notes to Display'}                  </div>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} updateNote={updateNote} showAlert={props.showAlert} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\AddNote.js  import React, {useContext, useState} from 'react';  import noteContext from '../context/notes/noteContext';  const AddNote = (props) => {      const context = useContext(noteContext);      const { addNote } = context;      const [note, setNote] = useState({title: "", description: "", tag: ""})        const handleClick = (e) => {          e.preventDefault();          addNote(note.title, note.description, note.tag);          setNote({title: "", description: "", tag: ""});          props.showAlert("Added Successfully", "success");      }      const onChange = (e) => {          setNote({...note, [e.target.name]: e.target.value})      }      return (          <div className="container my-3">              <h2>Add a Note</h2>              <form className='my-3'>                  <div className="mb-3">                      <label htmlFor="title" className="form-label">Title</label>                      <input type="text" className="form-control" id="title" name="title" value={note.title} onChange={onChange} minLength={5} required />                  </div>                  <div className="mb-3">                      <label htmlFor="description" className="form-label">Description</label>                      <input type="text" className="form-control" id="description" name="description" value={note.description} onChange={onChange} minLength={5} required />                  </div>                  <div className="mb-3">                      <label htmlFor="tag" className="form-label">Tag</label>                      <input type="text" className="form-control" id="tag" name="tag" value={note.tag} onChange={onChange} minLength={5} required/>                  </div>                  <button disabled={note.title.length<5 || note.description.length<5 } type="submit" className="btn btn-primary" onClick={handleClick}>Add Note</button>              </form>          </div>      )  }  export default AddNote |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Home.js  import Notes from './Notes';  export const Home = (props) => {    const { showAlert } = props    return (      <div>        <Notes showAlert={showAlert} />      </div>    )  } |

|  |  |
| --- | --- |
| **72. Fetching User Specific Notes in iNotebook** | **Date: 27/01/25** |

|  |
| --- |
| npm run both |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\context\notes\NoteState.js  import NoteContext from "./noteContext";  import { useState } from "react";  const NoteState = (props) => {    const host = "http://localhost:5000"    const notesInitial = []    const [notes, setNotes] = useState(notesInitial)    // Get all Notes    const getNotes = async () => {      // API Call      const response = await fetch(`${host}/api/notes/fetchallnotes`, {        method: 'GET',        headers: {          'Content-Type': 'application/json',          "auth-token": localStorage.getItem('token')        }      });      const json = await response.json();      setNotes(json)    }    // Add a Note    const addNote = async (title, description, tag) => {      // TODO: API Call      // API Call      const response = await fetch(`${host}/api/notes/addnote`, {        method: 'POST',        headers: {          'Content-Type': 'application/json',          "auth-token": localStorage.getItem('token')        },        body: JSON.stringify({title, description, tag})      });        const note = await response.json();      setNotes(notes.concat(note))    }    // Delete a Note    const deleteNote = async (id) => {      // API Call      const response = await fetch(`${host}/api/notes/deletenote/${id}`, {        method: 'DELETE',        headers: {          'Content-Type': 'application/json',          "auth-token": localStorage.getItem('token')        }      });      const json = response.json();      console.log(json);      const newNotes = notes.filter((note) => { return note.\_id !== id })      setNotes(newNotes)    }    // Edit a Note    const editNote = async (id, title, description, tag) => {      // API Call      const response = await fetch(`${host}/api/notes/updatenote/${id}`, {        method: 'PUT',        headers: {          'Content-Type': 'application/json',          "auth-token": localStorage.getItem('token')        },        body: JSON.stringify({title, description, tag})      });      const json = response.json();      console.log(json);      let newNotes = JSON.parse(JSON.stringify(notes))      // Logic to edit in client      for (let index = 0; index < newNotes.length; index++) {        const element = newNotes[index];        if (element.\_id === id) {          newNotes[index].title = title;          newNotes[index].description = description;          newNotes[index].tag = tag;          break;        }      }      setNotes(newNotes);    }    return (      <NoteContext.Provider value={{ notes, addNote, deleteNote, editNote, getNotes }}>        {props.children}      </NoteContext.Provider>    )  }  export default NoteState; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Login.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Login = (props) => {      const [credentials, setCredentials] = useState({ email: "", password: "" })      let navigate = useNavigate();      const handleSubmit = async (e) => {          e.preventDefault();          const response = await fetch("http://localhost:5000/api/auth/login", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify({ email: credentials.email, password: credentials.password })          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              props.showAlert("Logged In Successfully", "success");              navigate("/");          } else {              props.showAlert("Invalid Credentials", "danger");          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div className='mt-3'>              <h2>Logi to Continue to iNotebook</h2>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email address</label>                      <input type="email" className="form-control" value={credentials.email} onChange={onChange} id="email" name="email" aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" value={credentials.password} onChange={onChange} id="password" name="password" />                  </div>                  <button type="submit" className="btn btn-primary">Login</button>              </form>          </div>      )  }  export default Login |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Signup.js  import React, { useState } from 'react'  import { useNavigate } from 'react-router-dom';  const Signup = (props) => {      const [credentials, setCredentials] = useState({ name: "", email: "", password: "", cpassword: "" })      let navigate = useNavigate();      const handleSubmit = async (e) => {          e.preventDefault();          const { name, email, password } = credentials;          const response = await fetch("http://localhost:5000/api/auth/createuser", {              method: 'POST',              headers: {                  'Content-Type': 'application/json',              },              body: JSON.stringify ({ name, email, password })          });          const json = await response.json();          console.log(json);          if (json.success) {              // Save the auth token and redirect              localStorage.setItem('token', json.authtoken);              navigate("/");              props.showAlert("Account Created Successfully", "success");          } else {              props.showAlert("Invalid Details", "danger")          }      }      const onChange = (e) => {          setCredentials({ ...credentials, [e.target.name]: e.target.value })      }      return (          <div className='container mt-2'>              <h2>Create a Account to use iNotebook</h2>              <form onSubmit={handleSubmit}>                  <div className="mb-3">                      <label htmlFor="name" className="form-label">Name</label>                      <input type="text" className="form-control" id="name" onChange={onChange} name='name' aria-describedby="name" />                  </div>                  <div className="mb-3">                      <label htmlFor="email" className="form-label">Email</label>                      <input type="email" className="form-control" id="email" onChange={onChange} name='email' aria-describedby="email" />                      <div id="emailHelp" className="form-text">We'll never share your email with anyone else.</div>                  </div>                  <div className="mb-3">                      <label htmlFor="password" className="form-label">Password</label>                      <input type="password" className="form-control" id="password" onChange={onChange} name='password' minLength={5} required/>                  </div>                  <div className="mb-3">                      <label htmlFor="cpassword" className="form-label">Confirm Password</label>                      <input type="password" className="form-control" id="cpassword" onChange={onChange} name='cpassword' minLength={5} required/>                  </div>                  <button type="submit" className="btn btn-primary">Sign-Up</button>              </form>          </div>      )  }  export default Signup |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Notes.js  import React, { useContext, useEffect, useRef, useState } from 'react';  import { useNavigate } from 'react-router-dom';  import noteContext from '../context/notes/noteContext';  import Noteitem from './Noteitem';  import AddNote from './AddNote';  const Notes = (props) => {      const context = useContext(noteContext);      let navigate = useNavigate();      const { notes, getNotes, editNote } = context;      useEffect(() => {          if (localStorage.getItem('token')) {              getNotes()          } else {              navigate("/login");          }          // eslint-disable-next-line      }, [])      const ref = useRef(null)      const refClose = useRef(null)      const [note, setNote] = useState({ id: "", etitle: "", edescription: "", etag: "" })      const updateNote = (currentNote) => {          ref.current.click();          setNote({ id: currentNote.\_id, etitle: currentNote.title, edescription: currentNote.description, etag: currentNote.tag });      }      const handleClick = (e) => {          editNote(note.id, note.etitle, note.edescription, note.etag);          refClose.current.click();          props.showAlert("Update Successfully", "success");      }      const onChange = (e) => {          setNote({ ...note, [e.target.name]: e.target.value })      }      return (          <>              <AddNote showAlert={props.showAlert} />              <button ref={ref} type="button" className="btn btn-primary d-none" data-bs-toggle="modal" data-bs-target="#exampleModal">                  Launch demo modal              </button>              <div className="modal fade" id="exampleModal" tabIndex="-1" aria-labelledby="exampleModalLabel" aria-hidden="true">                  <div className="modal-dialog">                      <div className="modal-content">                          <div className="modal-header">                              <h1 className="modal-title fs-5" id="exampleModalLabel">Edit Note</h1>                              <button type="button" className="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>                          </div>                          <div className="modal-body">                              <form className='my-3'>                                  <div className="mb-3">                                      <label htmlFor="etitle" className="form-label">Title</label>                                      <input type="text" className="form-control" id="etitle" name="etitle" value={note.etitle} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="edescription" className="form-label">Description</label>                                      <input type="text" className="form-control" id="edescription" name="edescription" value={note.edescription} onChange={onChange} minLength={5} required/>                                  </div>                                  <div className="mb-3">                                      <label htmlFor="etag" className="form-label">Tag</label>                                      <input type="text" className="form-control" id="etag" name="etag" value={note.etag} onChange={onChange} minLength={5} required />                                  </div>                              </form>                          </div>                          <div className="modal-footer">                              <button ref={refClose} type="button" className="btn btn-secondary" data-bs-dismiss="modal">Close</button>                              <button disabled={note.etitle.length<5 || note.edescription.length<5 } onClick={handleClick} type="button" className="btn btn-primary">Update Note</button>                          </div>                      </div>                  </div>              </div>              <div className="row my-3">                  <h2>You Note</h2>                  <div className="container mx-2">                      {notes.length === 0 && 'No Notes to Display'}                  </div>                  {notes.map((note) => {                      return <Noteitem key={note.\_id} updateNote={updateNote} showAlert={props.showAlert} note={note} />                  })}              </div>          </>      )  }  export default Notes |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\About.js  import React from 'react';  import { useNavigate } from 'react-router-dom';  import { useEffect } from 'react';  const About = () => {      let navigate = useNavigate();      useEffect(() => {          if (!localStorage.getItem('token')) {              navigate("/login");          }          // eslint-disable-next-line      }, []);      return (          <div className="container my-4">              <h2 className="text-center mb-4">About iNotebook</h2>              <div className="row">                  <div className="col-md-8 mx-auto">                      <div className="card">                          <div className="card-body">                              <h5 className="card-title">Your Digital Notebook Solution</h5>                              <p className="card-text">                                  iNotebook is a secure cloud-based note-taking application that allows you to:                              </p>                              <ul className="list-group list-group-flush mb-3">                                  <li className="list-group-item">✓ Create and manage personal notes</li>                                  <li className="list-group-item">✓ Access your notes from anywhere</li>                                  <li className="list-group-item">✓ Secure your notes with user authentication</li>                                  <li className="list-group-item">✓ Edit and delete notes easily</li>                                  <li className="list-group-item">✓ Add tags to organize your notes</li>                              </ul>                              <h6 className="card-subtitle mb-2 text-muted">How to Use:</h6>                              <p className="card-text">                                  Simply sign up for an account, log in, and start creating your notes.                                  Each note can have a title, description, and tag for better organization.                                  Your notes are private and can only be accessed with your credentials.                              </p>                              <div className="alert alert-info" role="alert">                                  <strong>Security Note:</strong> All your notes are encrypted and stored securely on our servers.                              </div>                          </div>                      </div>                  </div>              </div>              <div className="row mt-4">                  <div className="col-md-8 mx-auto">                      <div className="card">                          <div className="card-body">                              <h5 className="card-title">Technical Details</h5>                              <p className="card-text">                                  Built with modern web technologies:                              </p>                              <ul className="list-group list-group-flush">                                  <li className="list-group-item">• Frontend: React.js</li>                                  <li className="list-group-item">• Backend: Node.js & Express.js</li>                                  <li className="list-group-item">• Database: MongoDB</li>                                  <li className="list-group-item">• Authentication: JWT (JSON Web Tokens)</li>                              </ul>                          </div>                      </div>                  </div>              </div>          </div>      );  }  export default About; |

|  |
| --- |
| File path: C:\wamp64\www\MERN\_STACK\Code\_With\_Harry\React\_Js\inotebook\src\components\Navbar.js  import React from 'react';  import { Link, useLocation, useNavigate } from "react-router-dom";  const Navbar = () => {      let navigate = useNavigate();      const handleLogout = () => {          localStorage.removeItem('token');          navigate('/login');      }      let location = useLocation();      return (          <nav className="navbar navbar-expand-lg navbar-dark bg-dark">              <div className="container-fluid">                  <Link className="navbar-brand" to="/">iNotebook</Link>                  <button className="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">                      <span className="navbar-toggler-icon"></span>                  </button>                  <div className="collapse navbar-collapse" id="navbarSupportedContent">                      <ul className="navbar-nav me-auto mb-2 mb-lg-0">                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname == "/" ? "active" : ""}`} aria-current="page" to="/">Home</Link>                          </li>                          <li className="nav-item">                              <Link className={`nav-link ${location.pathname == "/about" ? "active" : ""}`} to="/about">About</Link>                          </li>                      </ul>                      {!localStorage.getItem('token') ?                          <form form className="d-flex" role="search">                              <Link className="btn btn-primary mx-1" to="/login" role="button">Login</Link>                              <Link className="btn btn-primary mx-1" to="/signup" role="button">Sign-Up</Link>                          </form> :                          <button className="btn btn-primary mx-1" onClick={handleLogout}>Logout</button>}                  </div>              </div>          </nav >      )  }  export default Navbar |

|  |  |
| --- | --- |
| **73. Introduction to Redux** | **Date: 30/01/25** |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |

|  |
| --- |
| File path: |