**React Tutorial - Single Page Application w/ Nav Routing**

**Code**: <https://github.com/benjaminrittenhouse/CIS4282-tutorials/tree/main/react-nav-project>

**Note:** At the time of creating this, Heroku is our method of deployment. You can link GitHub and Heroku (mentioned later in this tutorial), so I recommend maintaining your code via GitHub for version control and hosting.

1. Clone the above project and open it in your Visual Studio Code IDE
2. **First, let us run this project locally…**
   1. Open a terminal or a command prompt, and move to the folder we got from the link above. On Mac, for example, to get to my Desktop -> CIS4282-tutorials, I would do:
      1. “cd desktop” … enter, “cd CIS4282-tutorials”



* + 1. Once you are in this folder, again cd into “react-spa”. This is the name of the folder that contains our single page application, our parent folder. We must cd once more into “client”, which is where our react application lives.
    2. Finally, we can run the app **locally** by typing, in our terminal / cmd prompt:

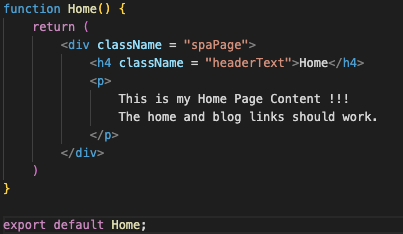
**npm start**

* + 1. Give it a moment, and this should open a tab in your browser with the app running locally. If not, open a browser and move to: [**http://localhost:3000/**](http://localhost:3000/)

1. **Understanding + Editing the Code**
   1. This is a **react app** with no server side. In other words, we only have a frontend to work with.
   2. In your IDE, visit the **src** folder within **/client** that we moved to earlier.
      1. The file that is running the React app is named **App.js.** Open this file.
      2. A React.js file contains a function that will be **exported.** These can also be **components** which will make more sense in a moment. For the time being, notice we are exporting a function named **SPA,** created on line 7 and exported at the bottom of the file.
   3. Within every function that we are **exporting** in react, there must be a single parent div that contains the rest of the component within it. In our case, it is the div at the very top that has **class=”App”**
   4. The rest of the code contains HTML that you should be familiar with. However the part that is new is the **Routing** and use of **Components.**
      1. React allows importing of components that you have created as well as React modules that you import from elsewhere. In this case, we imported **Routes** and **Link** from **react-router-dom,** an accessible react library for routing
      2. We use **Link** to create our navbar. Notice the two Link items we have that use **to=””** to tell React where to move when clicked.
         1. The value placed for the **to** attribute of **Link** is created in the code just below the Links, **Routes**
      3. We create Reacts version of a routing table by encapsulating individual Route components within **Routes**
         1. Route lets us determine the **path,** and the **component** to be rendered when that path is moved to. In this case, the syntax within Route is

**element = { <ComponentNameHere /> }**

* 1. You can see we have a simple routing with two components that we created, **Home** and **Blog.** Move to **src/components** and open these files
     1. Notice the consistency of creating a function and exporting it as we mentioned before.



* + 1. These components in combination with Routing allow for us to have a single page application that changes what is rendered below the nav depending on what route we have moved to!

1. **Publishing and Running**
   1. To deploy your app, linking Github and Heroku (the hosting service), follow these instructions:
      1. <https://devcenter.heroku.com/articles/git#:~:text=To%20deploy%20your%20app%20to,heroku%20main%20Initializing%20repository%2C%20done>.