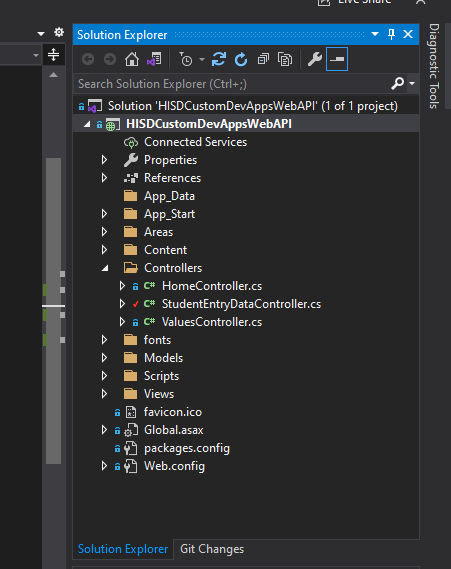
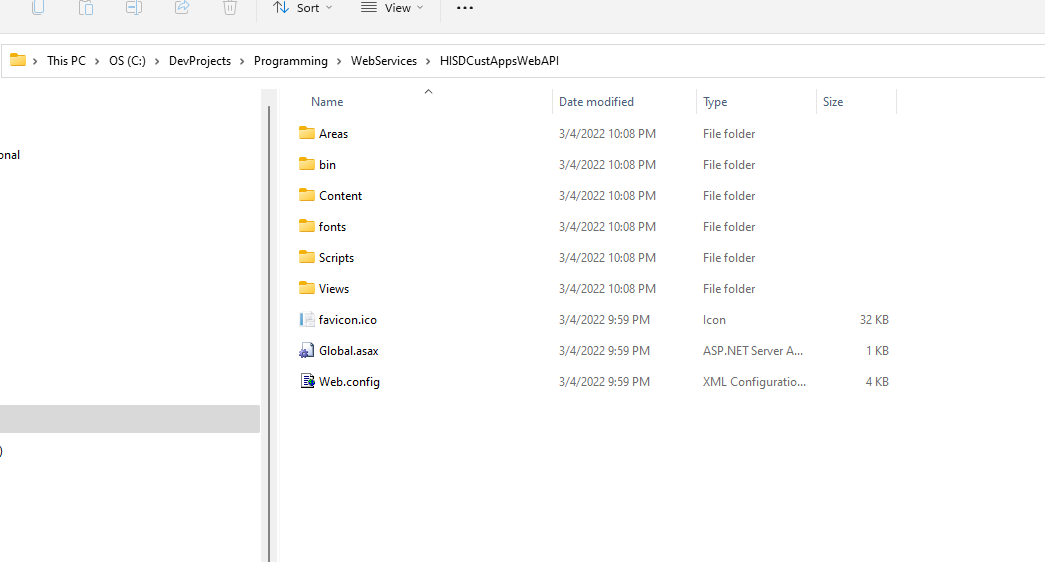
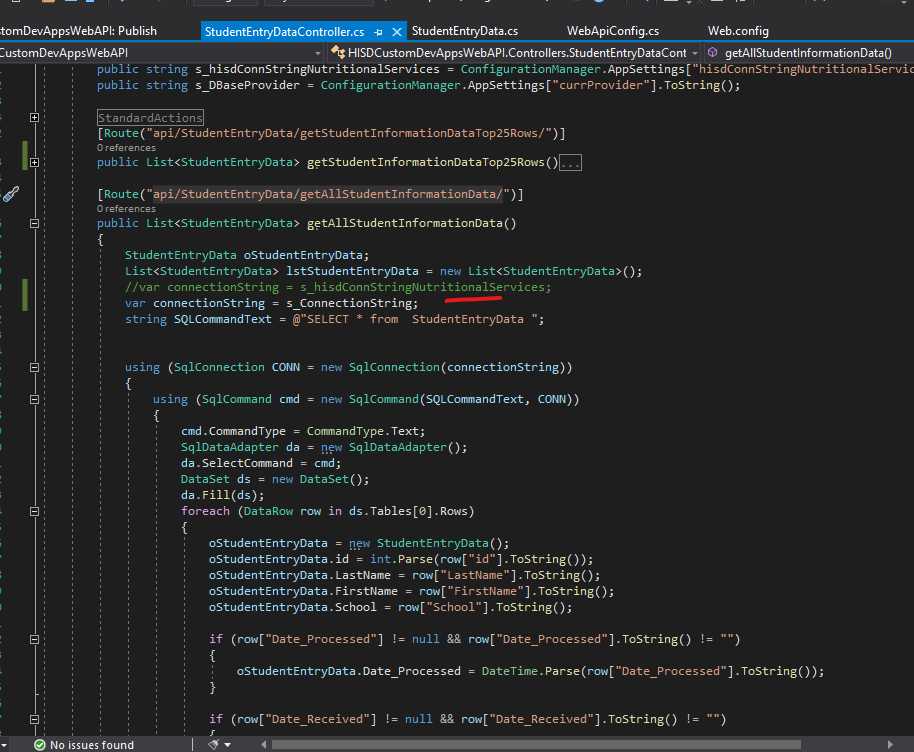
**WEBAPI Project**  
 **Location:**C:\DevProjects\Programming\VisualStudio\HISDCustomDevAppsWebAPI\  
Local Web Install – IIS  
C:\DevProjects\Programming\WebServices\HISDCustAppsWebAPI  
  
  
**I CHANGED THE CONNECTION STRINGS  
**

**Current Project Locations  
Name:**

HISDCustomDevAppsWebAPI

FOR GITHUB: - I pushed the project to GitHUB  
git clone:  
https://github.com/lionel5116/SBHISDCustomDevAppsWebAPI.git

**For my personal SandBox React Project – WEBAPI**

**Graphical user interface, text, application, email

Description automatically generated**

**Graphical user interface, text, application, email

Description automatically generated**

**https://github.com/lionel5116/SBHISDCustomDevAppsWebAPI.git**

C:\DevProjects\Programming\SANDBOX\_REACT\_WEBAPI\_PROJECT\_FILES\HISDCustomDevAppsWebAPI

echo "# SBHISDCustomDevAppsWebAPI" >> README.md

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin https://github.com/lionel5116/SBHISDCustomDevAppsWebAPI.git

git push -u origin main  
  
git remote add origin https://github.com/lionel5116/SBHISDCustomDevAppsWebAPI.git

git branch -M main

git push -u origin main

Text

Description automatically generated  
  
The Deployed Location  
Graphical user interface, table

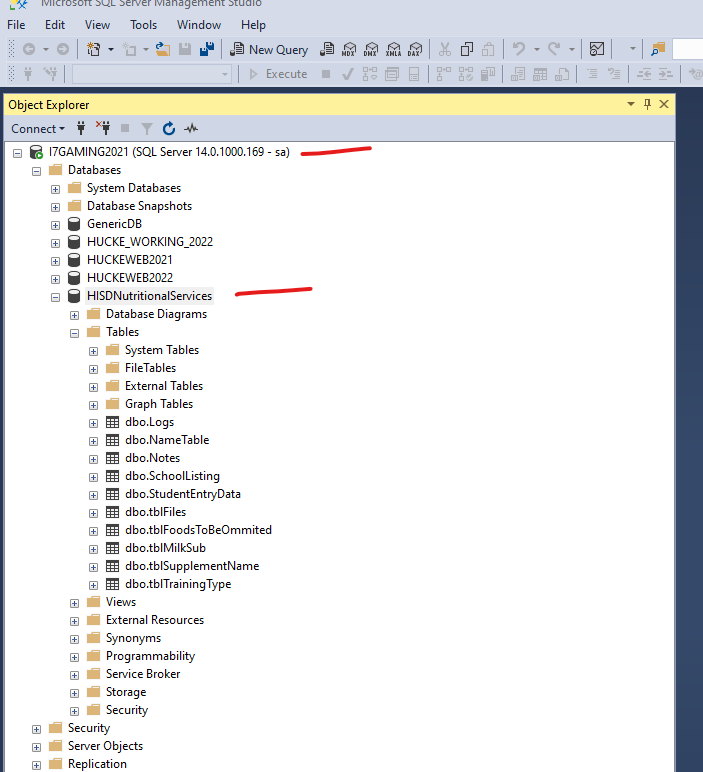
Description automatically generated with medium confidence  
Graphical user interface, text, application

Description automatically generated  
  
Graphical user interface, application

Description automatically generated  
  
Name is IIS

HISDCustAppsWebAPI

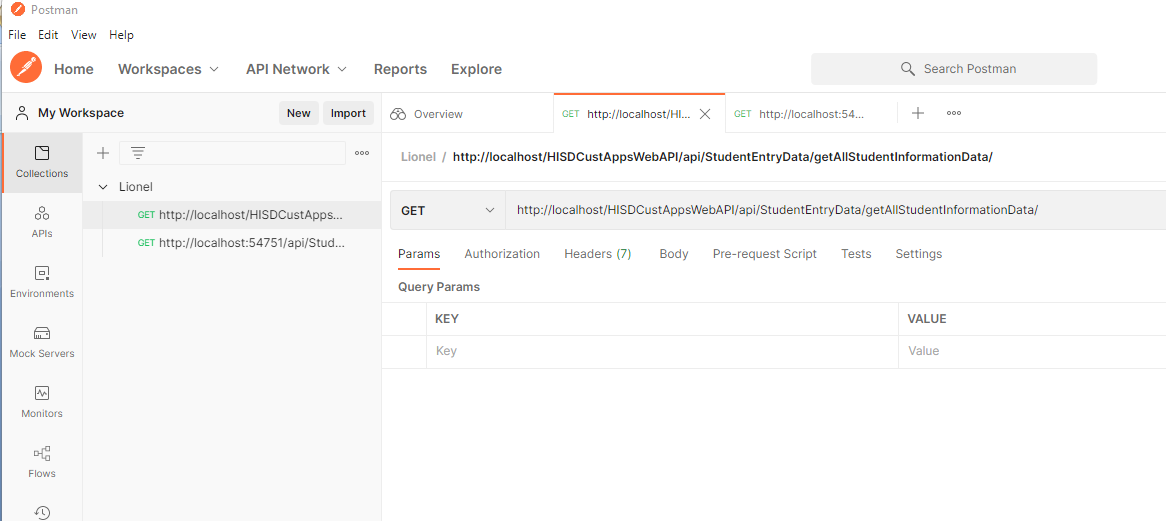
**FOR THE DATABASE**

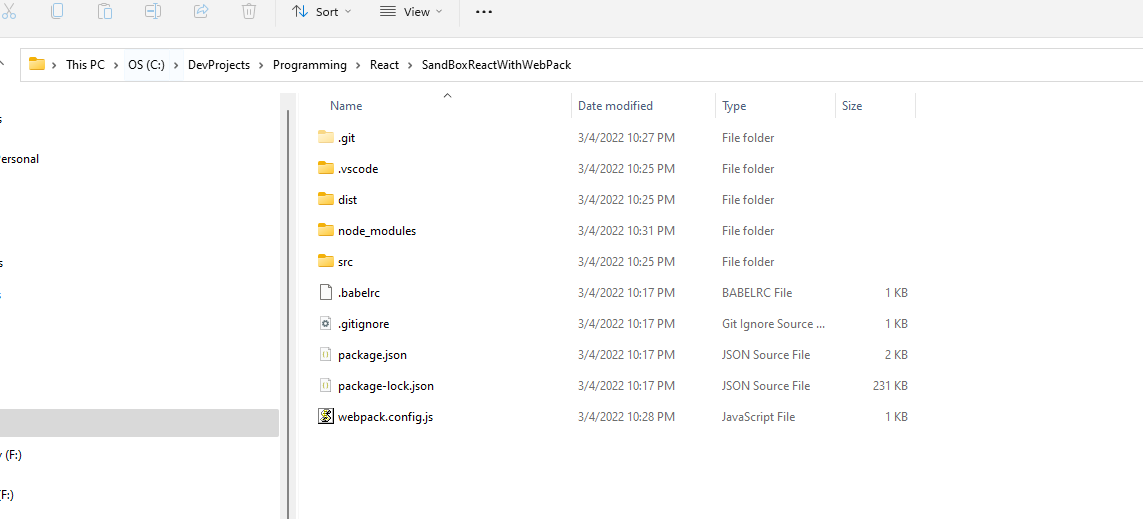
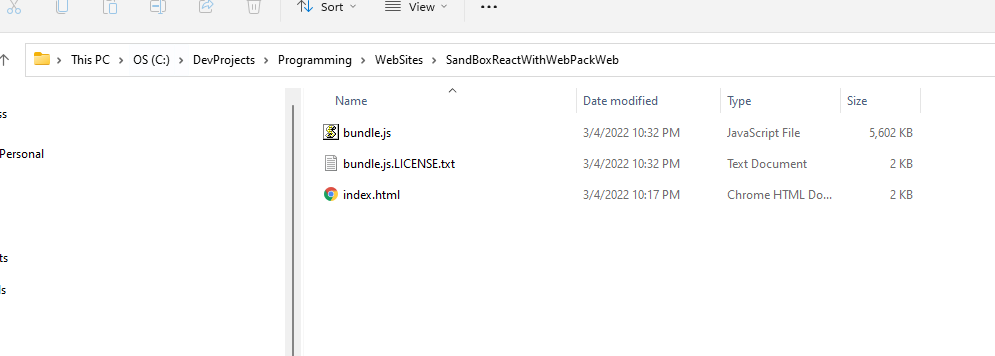
****

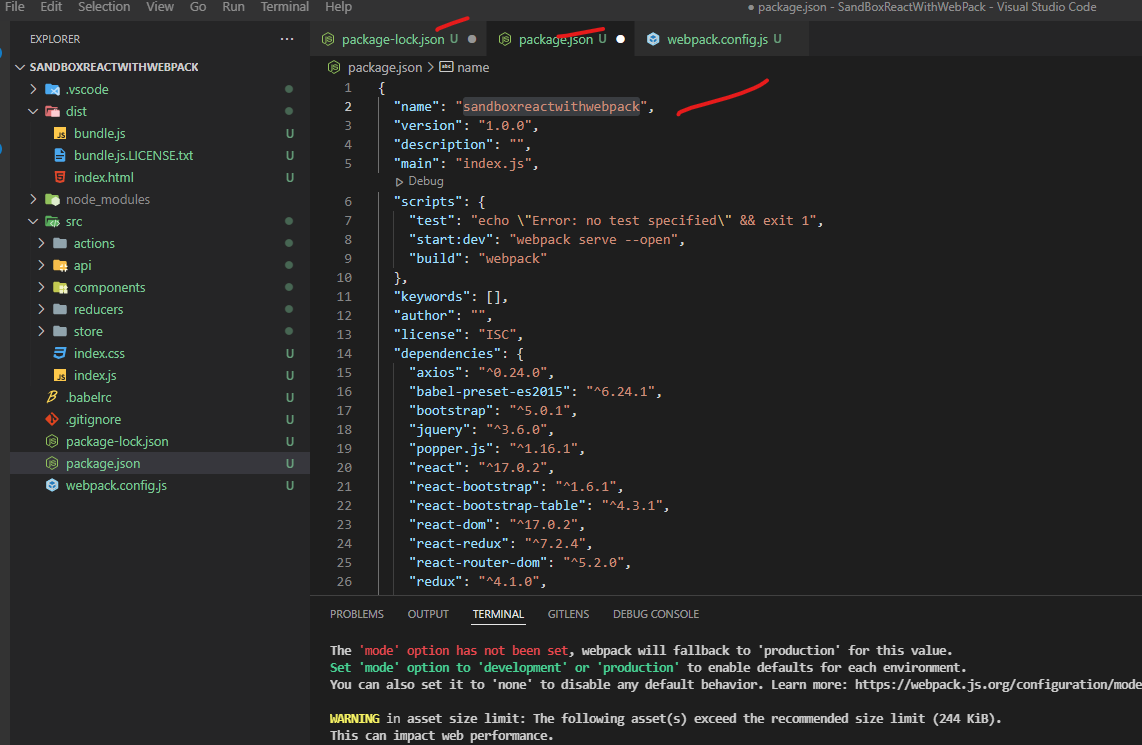
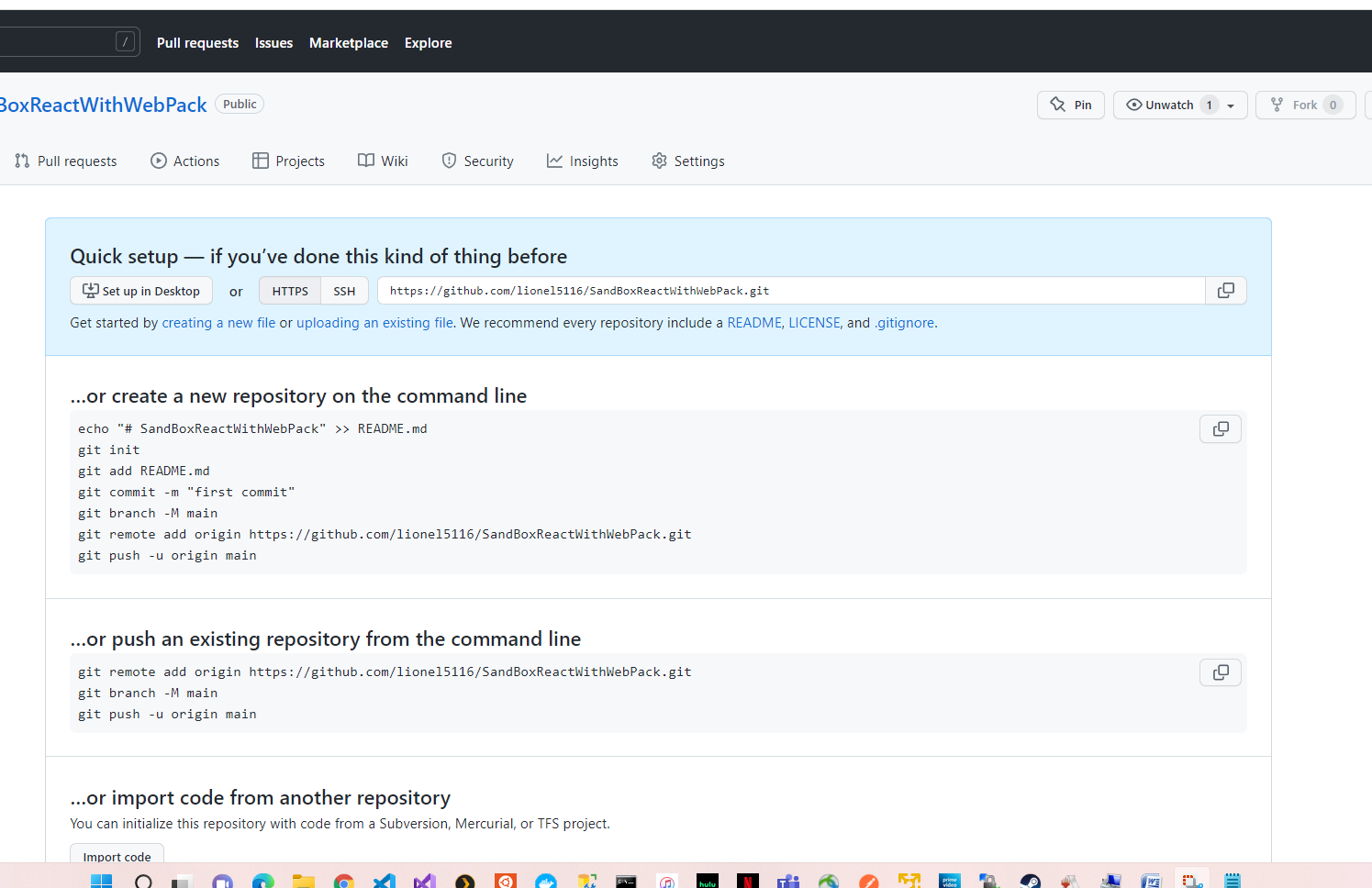
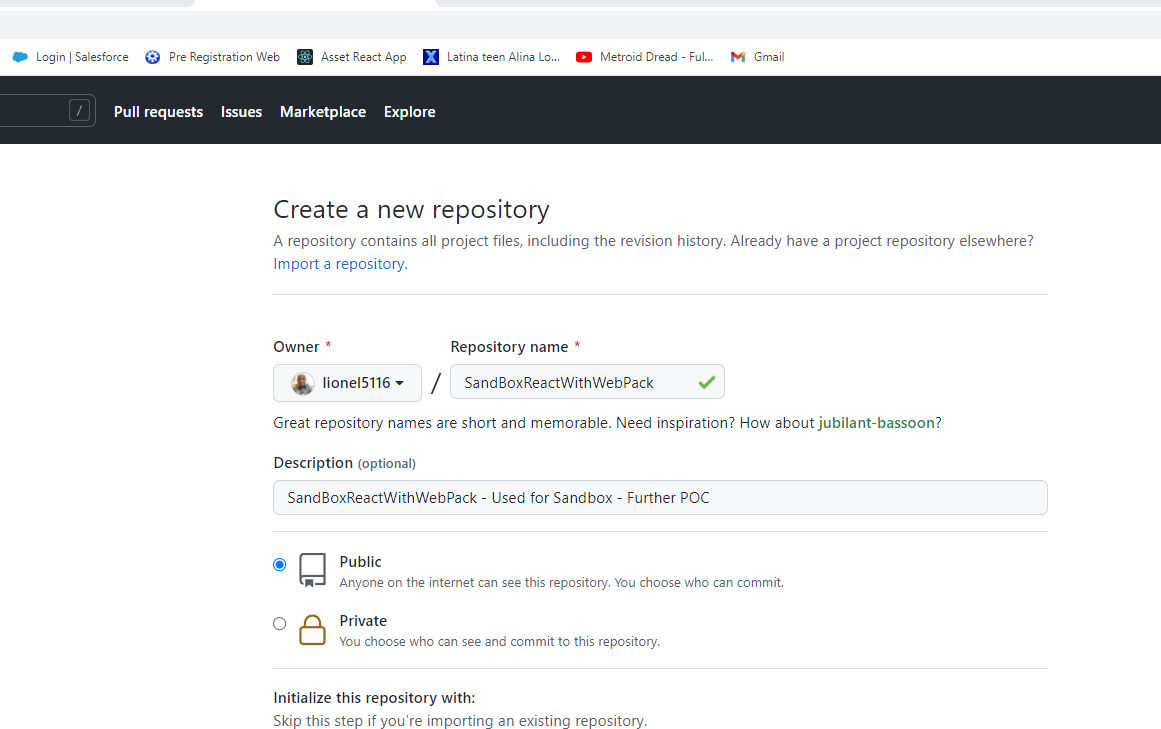
I am using a local copy of the HISDNutritionalServices DB

**POSTMAN ENDPOINTS**

http://localhost/HISDCustAppsWebAPI/api/StudentEntryData/getAllStudentInformationData/  
http://localhost:54751/api/StudentEntryData/getAllStudentInformationData/



React Project  
C:\DevProjects\Programming\React\SandBoxReactWithWebPack  
  
Site Install  
C:\DevProjects\Programming\WebSites\SandBoxReactWithWebPackWeb  
  
Renamed project in package.json file

  
Then deleted node\_modules and did a   
**npm install**  
To build project  
**npm run build**  
To run project:  
**npm run start:dev**  
  
**I CREATED A GIT REPO**  
**First Initialize**git init

git remote add origin https://github.com/lionel5116/SandBoxReactWithWebPack.git

**For subsequent pushes**

git add -A

git add .

git status

git commit -m "change notes"

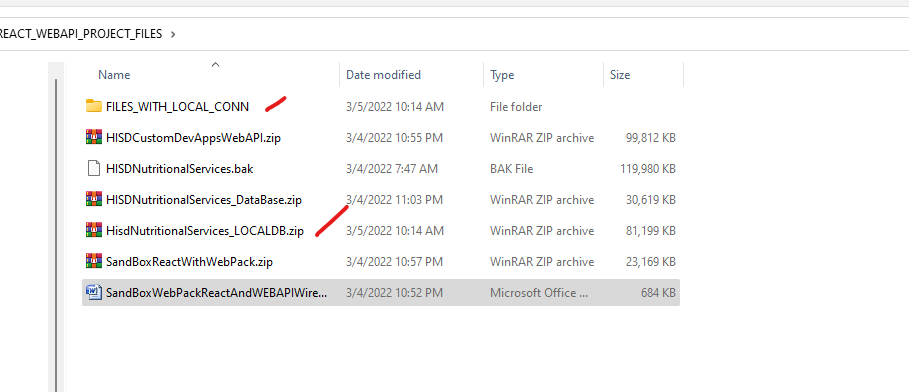
git push origin master

If you get this error:

error: failed to push some refs

to 'https://github.com/lionel5116/HisdNutritionalServices.git'

Type this command:

git push origin main  
  
**Nutritional Services – Web Project and Local DB Files**The FILES with LOCAL\_CONN are the web.config file and function for pulling shool data  
  
**React NPM’s for file-upload**  
<https://www.npmjs.com/package/use-file-upload>  
**React-dropzone-npm**

**Below is a good video on how to use react hooks with nodejs API**

<https://www.youtube.com/watch?v=PEGUFi9Sx-U>  
**Another good video on file upload**  
<https://www.youtube.com/watch?v=sp9r6hSWH_o>

**Good Video Series on React**

**React Hooks:**

Allow you to use React features without having to write a class, Ex; State of a component

<https://www.youtube.com/watch?v=cF2lQ_gZeA8&list=PLC3y8-rFHvwgg3vaYJgHGnModB54rxOk3&index=44>  
  
**React Effects**  
Mimics lifecycle hooks for a functional component. i:e componentDidMount, it essentially adds "effects" to a functional component

<https://www.youtube.com/watch?v=BH4xvzHa7H8&list=PLC3y8-rFHvwgg3vaYJgHGnModB54rxOk3&index=52>

**Great Article on React Hooks with REDUX**

<https://thoughtbot.com/blog/using-redux-with-react-hooks>  
**How to use Redux with Hooks**

React Redux now includes its own useSelector and useDispatch Hooks that can be used instead of **connect**.

**useSelector** is analogous to connect’s **mapStateToProps**. You pass it a function that takes the Redux store state and returns the pieces of state you’re interested in.

**useDispatch** replaces connect’s **mapDispatchToProps** but is lighter weight. All it does is return your store’s dispatch method so you can manually dispatch actions. I like this change, as binding action creators can be a little confusing to newcomers to React Redux.

Alright, so now let’s convert a React component that formerly used connect into one that uses Hooks.

Text

Description automatically generated

**Here a working example – This is in the sandbox project**

<https://www.freecodecamp.org/news/learn-redux-by-making-a-counter-application/>

A screenshot of a computer

Description automatically generated with medium confidence  
  
**And it works:**

Graphical user interface, text, application

Description automatically generated

**­­A recap of REDUX  
Diagram

Description automatically generated  
Actions:  
Text

Description automatically generated  
  
Text

Description automatically generated**

**React Bootstrap5 Sample Page Demo**

<https://www.youtube.com/watch?v=l2131Rok8XU>  
We implemented a react bootstrap web page  
**NOTE**: WE ALSO DID LIVE MARKUP CODING, WE STARTED THE WEBSITE AND CODED AND SAVED THE FILE WHILE IT DID LIVE RELOADING!!!  
Docs:  
<https://react-bootstrap.github.io/getting-started/introduction>  
Page: “SampleReactBootStrapWebPage.js”  
Key Notes on the Page  
Text

Description automatically generated  
  
The **’px-4 my-5’** is padding an margins in bootstrap react  
  
**Responsive Grids:**  
<https://react-bootstrap.netlify.app/layout/grid/#grid-system>  
  
**Images**  
Text

Description automatically generated **ClassName Keyword**Text

Description automatically generated

**Notice how when you using regular markup (not components):**class (for regular elements)  
className (for components)  
  
https://react-bootstrap.github.io/components/cards/  
A screenshot of a computer

Description automatically generated  
He also just copied and pasted the sample card code from the website (sweet), just like regular bootrap example coding  
  
After coding:

A screenshot of a computer

Description automatically generated with medium confidence  
SWEET!!!!!  
He also has some information on free web page hosting of your react application, kinda like Heroku  
<https://www.youtube.com/watch?v=VLGLlIhn3Ng>  
  
For spacing in bootstrap react

https://getbootstrap.com/docs/5.1/utilities/spacing/  
Text

Description automatically generated  
**You first specify what spacing you want either:**  
m (margin)  
p( padding)  
Then you specify the sides  
t (top  
b(bottom)  
…..  
Then a # between 0 and 5 (also auto) (You cannot specify any other numbers)

So for this example below, he used “mt-4” (margin-top) with a 4 spacer (\* 1.5)

Text

Description automatically generated

And for Text  
https://getbootstrap.com/docs/5.1/utilities/text/  
Its in the docs

Graphical user interface, text, application, email

Description automatically generated  
  
For example:

Text

Description automatically generated

**Adding a Custom Style css attribute**

If you want to add your own custom style for a component, you have to use in the format below:  
{ { <style: ‘ ‘}}  
Text

Description automatically generated  
  
  
For Components used by bootstrap react

https://react-bootstrap.github.io/components/alerts/  
  
  
<https://reactjs.org/docs/hooks-state.html>  
What is a hook in functional programming  
A Hook is a special function that lets you “hook into” React features  
**When would I use a Hook?**

If you write a function component and realize you need to add some state to it,

previously you had to convert it to a class. Now you can use a Hook inside the existing function component  
  
**Example:**

const [count, setCount] = useState(0);

What do we pass to useState as an argument? The only argument to the useState() Hook is the initial state.

Above we pass in 0 as the initial value

What does useState return?

It returns a pair of values: the current state and a **function** that updates it.

This is why we write const [count, **setCount**] = useState()

This is similar to this.state.count and this.**setState** in a class

Graphical user interface, text

Description automatically generated

Graphical user interface

Description automatically generated

Real world example I used in my file upload component example:  
Text

Description automatically generated

How to kill server when seeing “EADDRINUSE: address already in use”

<https://levelup.gitconnected.com/how-to-kill-server-when-seeing-eaddrinuse-address-already-in-use-16c4c4d7fe5d>

C:\Users\admin>netstat -ano|findstr "PID :3000"

Proto Local Address Foreign Address State PID

TCP 0.0.0.0:3000 0.0.0.0:0 LISTENING 18264

To kill this process (the /f is force):

taskkill /pid 18264 /f  
  
How React let’s you do an import as an “an”

A screenshot of a computer

Description automatically generated with medium confidence  
  
Notice above how you can reference a reducer as an alias without spelling out the actual function name