

Creating a Personal Web Portfolio

Introduction:

Hello, thank you for reaching out to us! If you are a college student in STEM with some experience in programming and are looking to expand your knowledge in front-end development, you may be a good fit for this position. The first step of your application requires you to create a one-page personal web portfolio in React. This may seem like a daunting task, however, we have created instructions that will guide you towards the deliverable. With our instructions, you will be able to create a simple web portfolio without any issues! After completing the deliverable, our hiring team will review your web portfolio. By creating a web portfolio, you are setting yourself apart from your peers, showing that you will go the extra mile, and will be able to learn new programming skills along the way. If you do not receive an offer from us, you have created a web portfolio for your own personal use and have gained knowledge in front-end development. Good luck!

Warnings/ Caution:

For this project, React version 16.12 or above is highly recommended. If you are importing images into your project, make sure they are either in .jpg, .jpeg, .png or .gif file format. At any point, you may receive error messages within your code that say your code is not working as intended. If this occurs, revise the previous steps and backtrack the issue you are facing. If you cannot figure out the issue, search the error message in [StackOverflow](#). This website will be able to assist you in debugging your code.

Technical Background:

- General Programming Background
 - Having some knowledge and experience with any programming language will be helpful. If you have experience with JavaScript, you may find the instructions to be easier.
 - If you would like to brush up on general knowledge of programming, you can find a beginner's guide to JavaScript [here](#).
- Familiarity with GitHub
 - Understanding how to download a project repository, and “pushing” your changes to the repository will be beneficial

Materials:

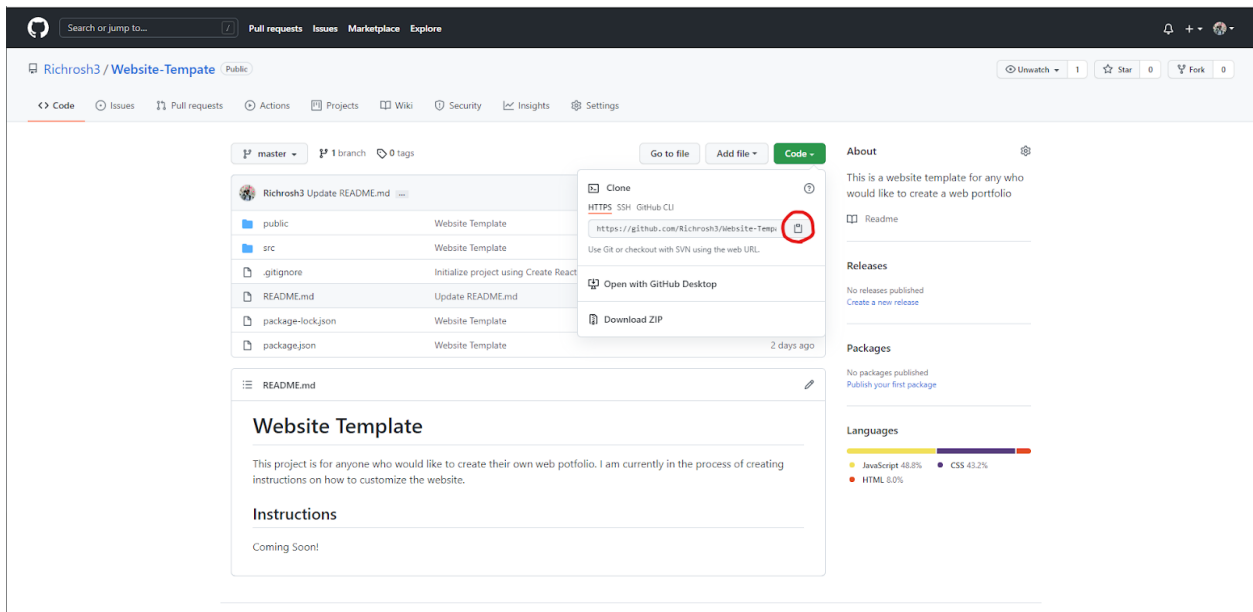
- A GitHub Account
 - This is where you will download the web portfolio template and launch your website
- A Code Editor
 - Visual Studio Code is recommended IDE; however, feel free to use any code editor that you are comfortable with. If you would like to use Visual Studio Code, you can download it [here](#).
- Your Main Browser
 - Your website will be displayed on your browser.
- React JS
 - React is a JavaScript library that allows you to enhance your UI. React is simple, fast, and scalable; which is exactly what we need when creating a web portfolio.

Having React JS installed on your laptop beforehand is crucial to this project. You can find instructions on how to install React JS [here](#). If you prefer a video tutorial, you can find one [here](#).

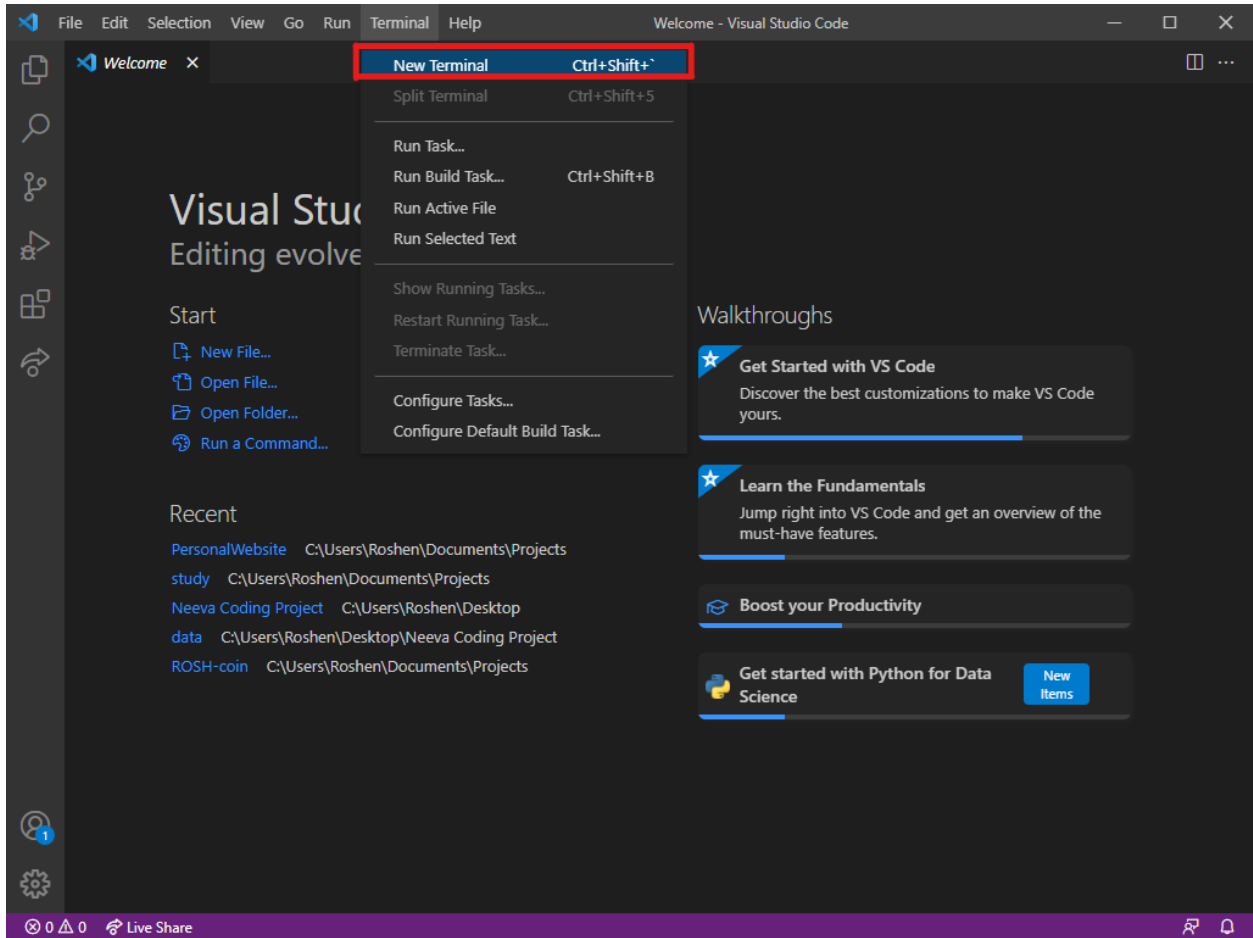
Setup:

Part 1: Cloning the Web Portfolio Template

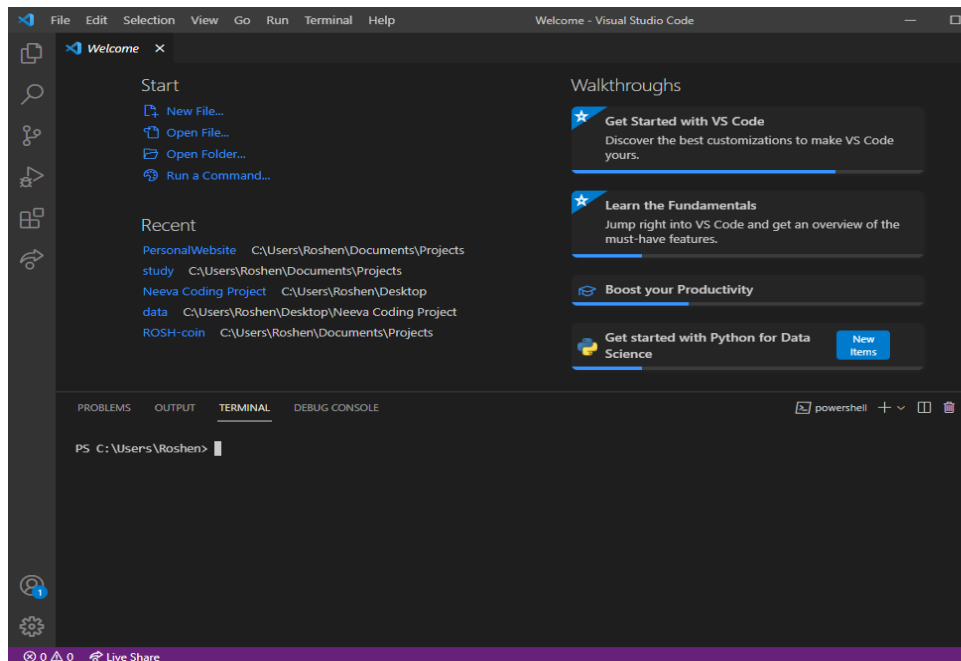
1. Log onto GitHub and direct yourself to the repository [here](#). Press the **Clipboard** icon that is circled in red. This will allow you to copy the Web Portfolio template.



2. Open VSCode and press the key combination “**Control + Shift + `**”. This will open up the VSCode Terminal. You can also press “**New Terminal**” from the top bar to open the terminal.



After opening the terminal, VSCode should look like this:

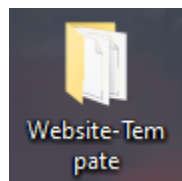


3. Clone our repository in a folder of our liking. I will clone my Web Template to a folder on my Desktop; however, you can choose wherever you would like your project folder to be located. Since we copied the repository address from GitHub earlier, all we have to do is run the clone command and paste the address.
 - a. Type “**git clone** ” in the terminal
 - b. Paste the repository address by pressing “**Control + V**”
 - c. Press “**Enter**”

Your terminal should have returned something like this:

```
PS C:\Users\Roshen\Desktop> git clone https://github.com/Richrosh3/Website-Tempate.git
Cloning into 'Website-Tempate'...
remote: Enumerating objects: 59, done.
remote: Counting objects: 100% (59/59), done.
remote: Compressing objects: 100% (53/53), done.
remote: Total 59 (delta 5), reused 59 (delta 5), pack-reused 0
Receiving objects: 100% (59/59), 1.91 MiB | 13.77 MiB/s, done.
Resolving deltas: 100% (5/5), done.
PS C:\Users\Roshen\Desktop> 
```

You should now see a folder called “Website-Template” in the location you choose.

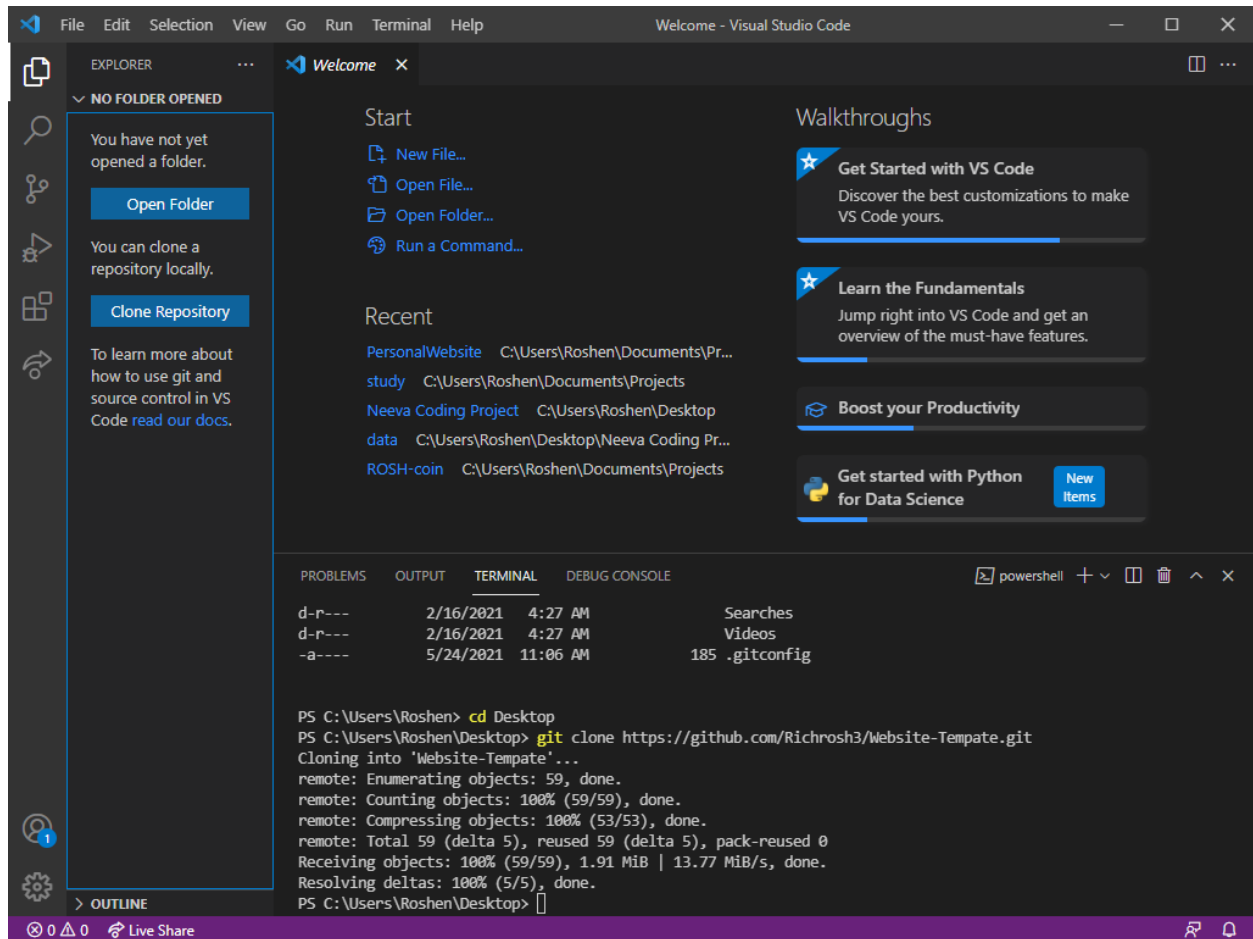


Congratulations, you just successfully cloned the repository!

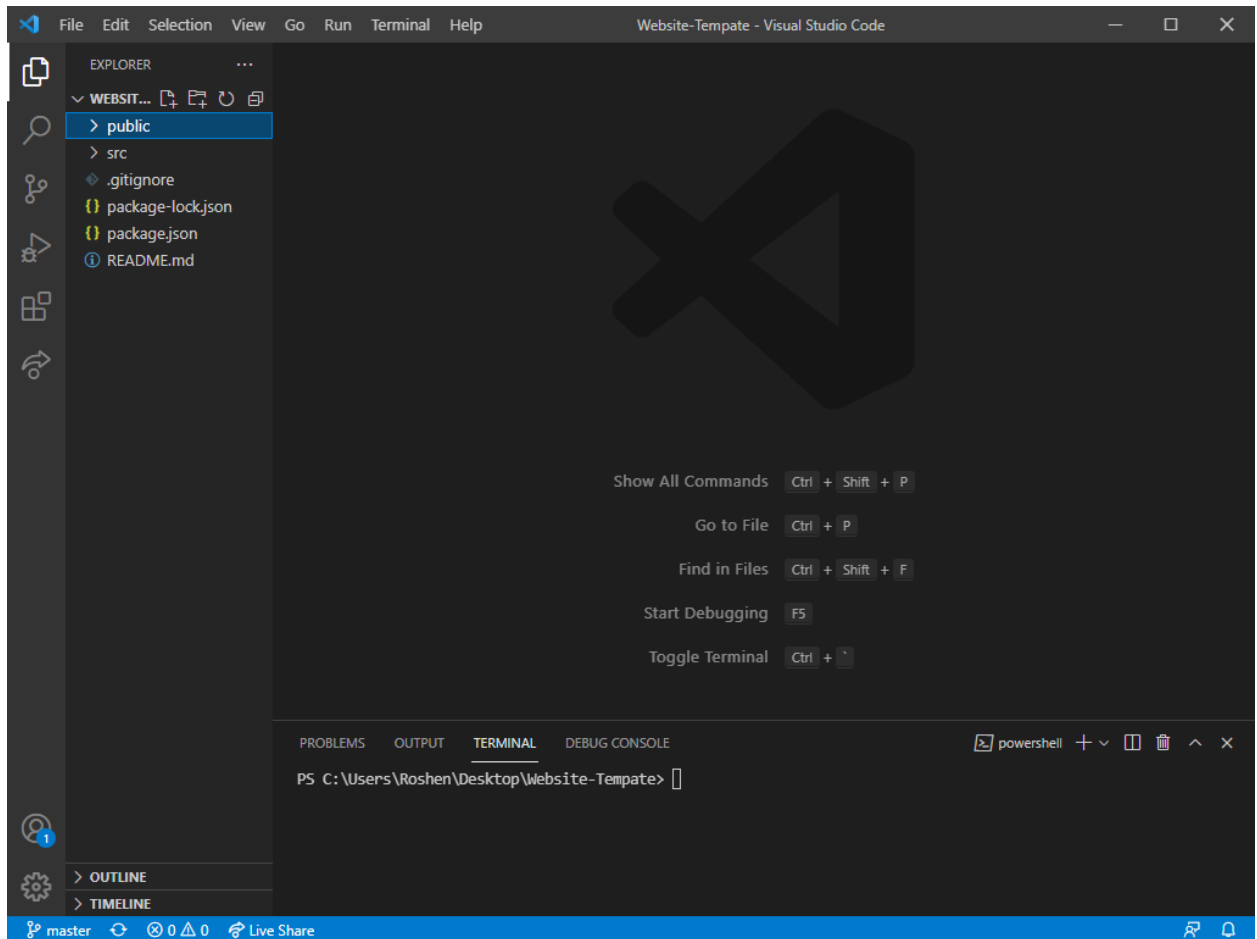
Part 2: Modifying the template



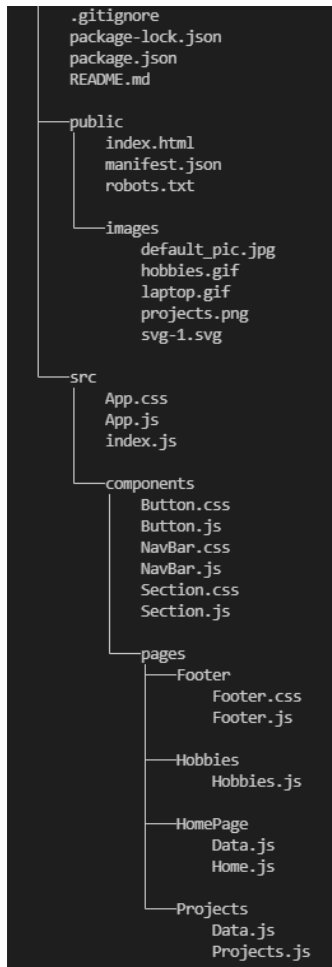
1. Click the icon located near the top left in VSCode. It will bring up a new side panel that should say “No Folder Opened”.



2. Press the **Open Folder** button and open the “Website-Template” folder that we just cloned from GitHub. Once you select the “Website-Template” folder, your VSCode should now look like this:



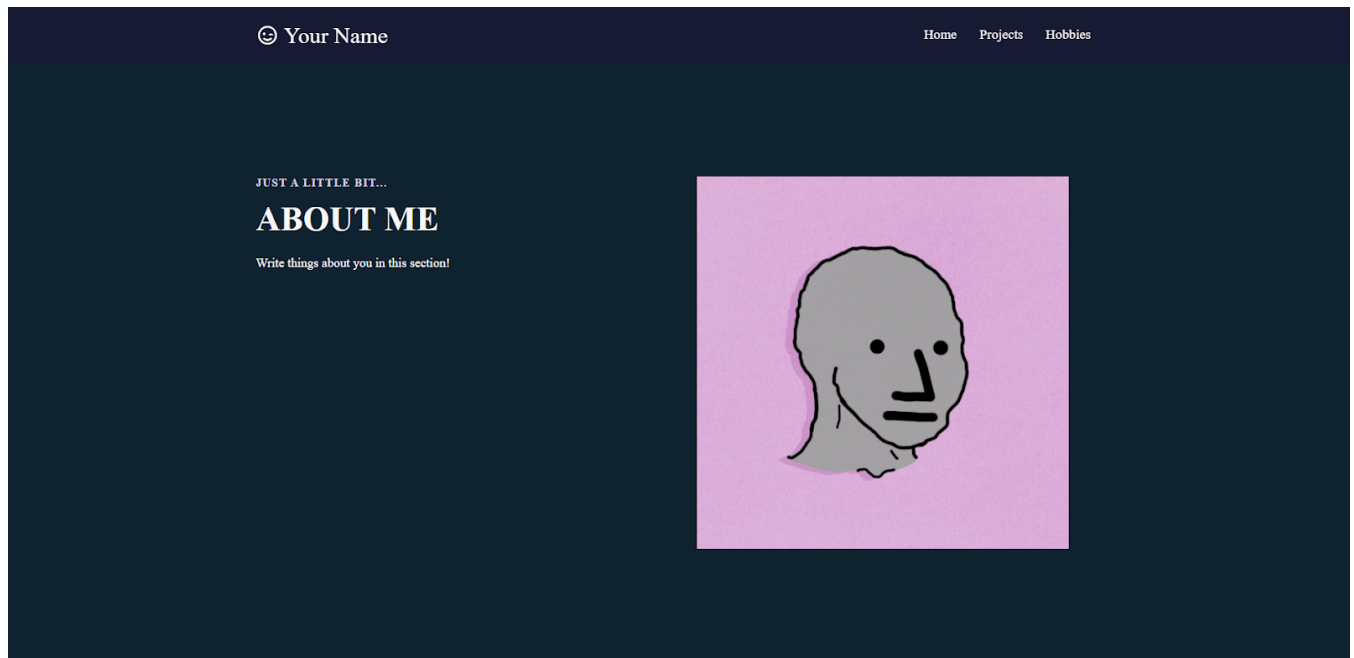
3. View the contents of the folder by pressing the arrows in the “Explorer” side panel. This side panel shows you the pathing of each file as well. Here is a diagram that displays the pathing of each file.



The main files we will be editing are:

- Components → NavBar.js
 - This is where you will add your name
- HomePage → Data.js
 - Allows you to customize each section of your Web Portfolio
- HomePage → Home.js
 - Allows you to control the number of sections you would like in your Web Portfolio
- Footer → Footer.js
 - This is where you will change your contact info and social media

4. Delete the “**Package-lock.json**” file.
5. Run “npm install” in your terminal
 - a. This could take a couple of minutes. Be patient!
 - b. This will create a new “package-lock.json” file. **DO NOT DELETE THIS NEW FILE**
6. Type “**npm start**” in the terminal.
 - a. If your browser is open, a new tab titled “React App” should be seen at the top.
 - b. This process should take around a minute the first time you run the command. If you don’t see a new tab called “React App” after waiting a while, restart your browser and try again.
7. Press the “React App” tab in your browser. Your web page should look like this:



Steps:

Part 3: Adding Your Name

1. Open the “NavBar.js” file by double-clicking it. Your VSCode should look like this:

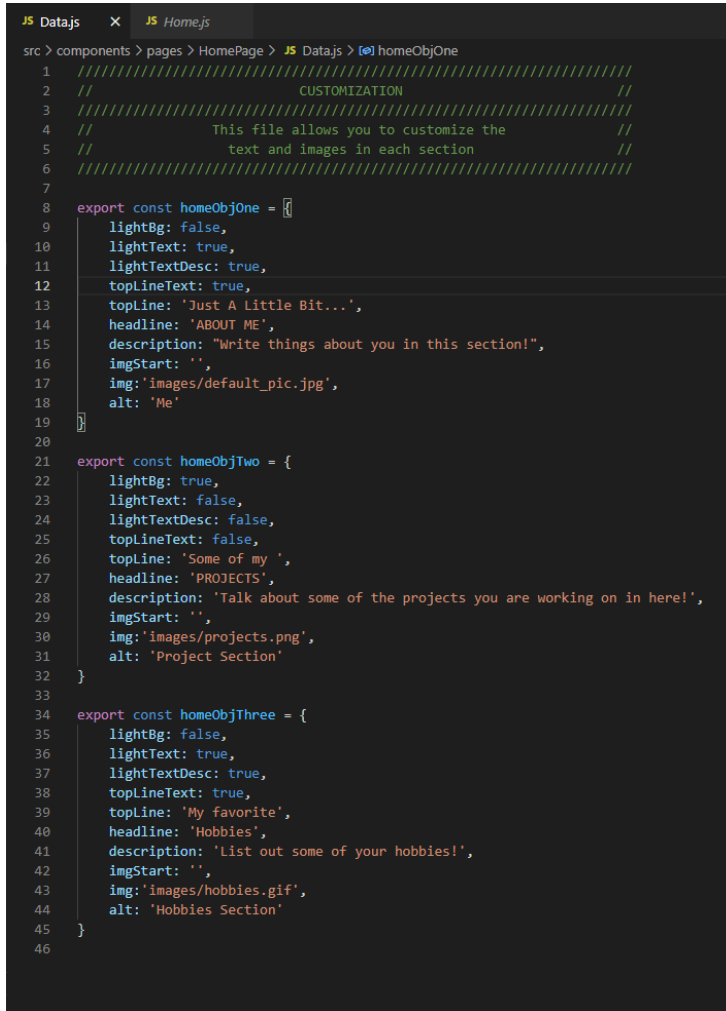
```
JS NavBar.js M X
src > components > JS NavBar.js > ...
1  //////////////////////////////////////////////////
2  //                                NAVIGATION BAR                                //
3  //////////////////////////////////////////////////
4  //  This file is to customize text in the top Navigation Bar  //
5  //////////////////////////////////////////////////
6
7
8  import React, { useState, useEffect } from 'react'
9  import { Link } from 'react-router-dom'
10 import { BiWinkSmile } from 'react-icons/bi'
11 import { FaBars, FaTimes } from 'react-icons/fa'
12 import './NavBar.css'
13 import { IconContext } from 'react-icons/lib'
14
15 function NavBar() {
16   const [click, setClick] = useState(false)
17   const [button, setButton] = useState(true)
18
19   const handleClick = () => setClick(!click)
20   const closeMobileMenu = () => setClick(false)
21
22   const showButton = () => {
23     if (window.innerWidth <= 960) {
24       setButton(false)
25     }
26     else {
27       setButton(true)
28     }
29   }
30
31   useEffect(() => {
32     showButton();
33   }, []);
34
35   window.addEventListener('resize', showButton)
36
37   return (
38     <>
39       <IconContext.Provider value={{ color: '#fff' }}>
40         <div className="navbar">
41           <div className="navbar-container container">
42             <Link to="/" className="navbar-logo" onClick={closeMobileMenu}>
43               <BiWinkSmile className='navbar-icon' />
44               Your Name
45             </Link>
46             <div className="menu-icon" onClick={handleClick}>
47               {click ? <FaTimes /> : <FaBars />}
48             </div>
49             <ul className={click ? 'nav-menu active' : 'nav-menu'}>
```

2. Change “Your Name” (on line 44) to the name you want others to see. For this example, I will change the name to Kanye West. Save this file and look at your “React App” tab in your browser. Your name should now be seen. The top Navigation Bar should look similar to this:



Part 4: Customizing your Web Portfolio Sections

1. Go to the HomePage folder in VSCode. Double-click “Data.js” to open up the file. It should look like this:



```
JS Data.js X JS Home.js
src > components > pages > HomePage > JS Data.js > homeObjOne
1 ///////////////////////////////////////////////////
2 //                      CUSTOMIZATION                      //
3 ///////////////////////////////////////////////////
4 //          This file allows you to customize the          //
5 //          text and images in each section                  //
6 ///////////////////////////////////////////////////
7
8 export const homeObjOne = {
9   lightBg: false,
10  lightText: true,
11  lightTextDesc: true,
12  topLineText: true,
13  topLine: 'Just A Little Bit...',
14  headline: 'ABOUT ME',
15  description: 'Write things about you in this section!',
16  imgStart: '',
17  img: 'images/default_pic.jpg',
18  alt: 'Me'
19 }
20
21 export const homeObjTwo = {
22   lightBg: true,
23   lightText: false,
24   lightTextDesc: false,
25   topLineText: false,
26   topLine: 'Some of my ',
27   headline: 'PROJECTS',
28   description: 'Talk about some of the projects you are working on in here!',
29   imgStart: '',
30   img: 'images/projects.png',
31   alt: 'Project Section'
32 }
33
34 export const homeObjThree = {
35   lightBg: false,
36   lightText: true,
37   lightTextDesc: true,
38   topLineText: true,
39   topLine: 'My favorite',
40   headline: 'Hobbies',
41   description: 'List out some of your hobbies!',
42   imgStart: '',
43   img: 'images/hobbies.gif',
44   alt: 'Hobbies Section'
45 }
46
```

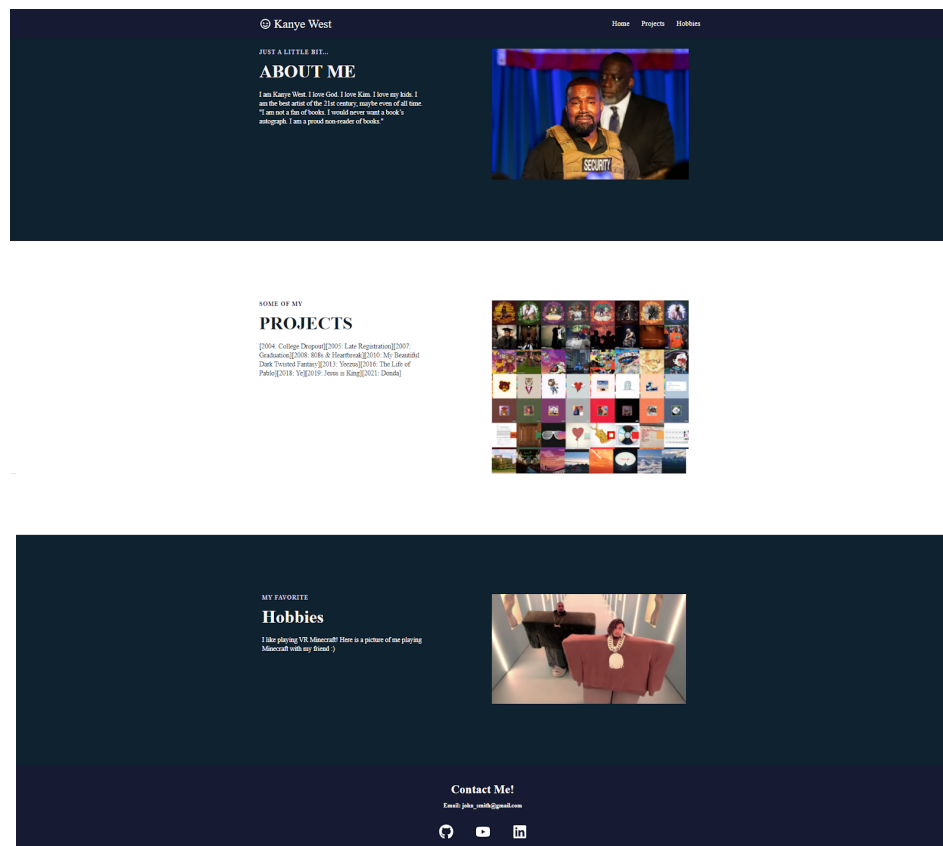
- HomeObjOne is the “About Me” Section
 - This is where you will write a quick summary about yourself
- HomeObjTwo is the “Projects” Section
 - List out your personal projects
- HomeObjThree is the “Hobbies” Section
 - Describe some of the hobbies you enjoy

2. Notice the “description” variable in HomeObjOne, HomeObjTwo, and HomeObjThree.

Delete the text inside the quotes and replace them with your own description for each section.

3. Notice the “img” variable in HomeObjOne, HomeObjTwo, and HomeObjThree. If you would like to change the picture of any section, go to the “images” and save your pictures with unique names there. Change the “img” variable of each section to the name of the picture you want to be displayed.

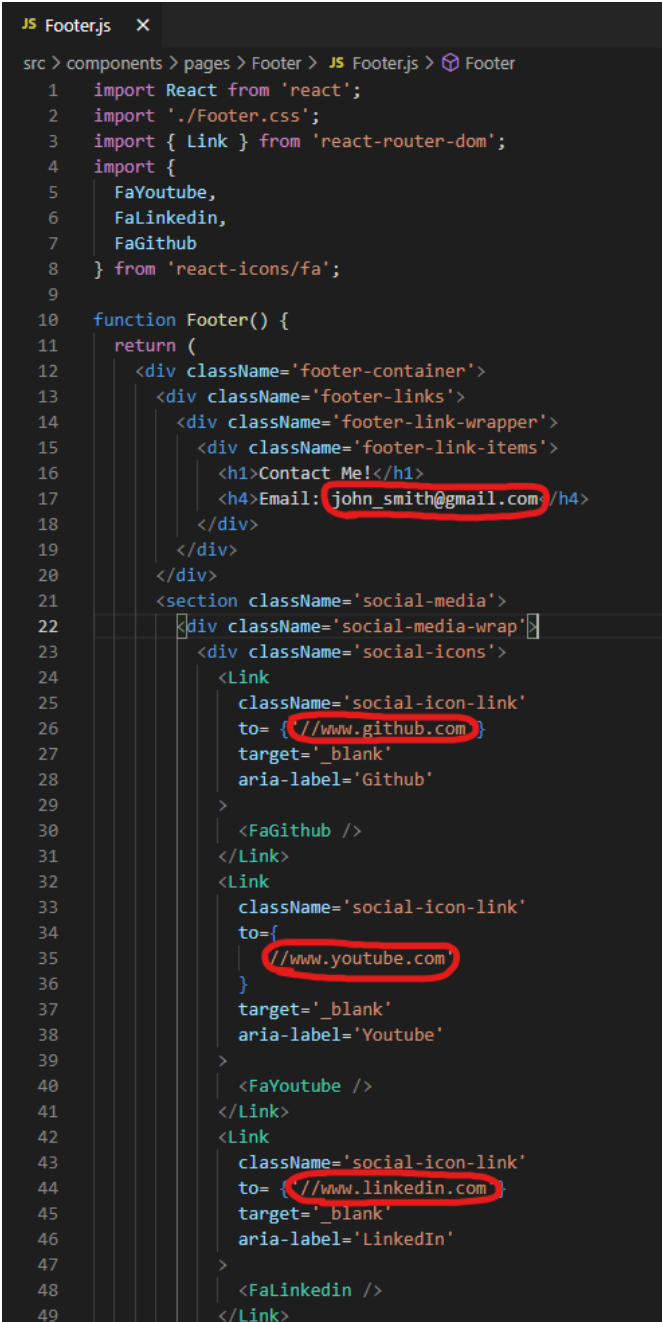
When you are done altering descriptions and images in the “Data.js” file, your web portfolio should look similar to this:



You're almost done!

Part 5: Contact Information

1. Find the “Footer.js” file and double-click it to open it. It should look like this:

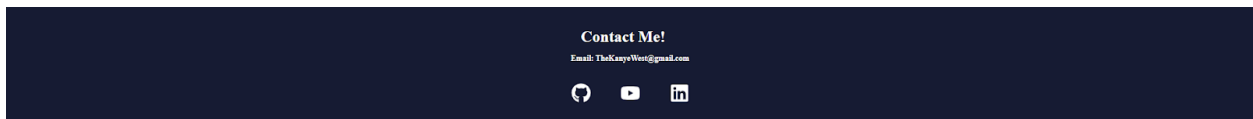


```
JS Footer.js X
src > components > pages > Footer > JS Footer.js > Footer
1  import React from 'react';
2  import './Footer.css';
3  import { Link } from 'react-router-dom';
4  import {
5    FaYoutube,
6    FaLinkedin,
7    FaGithub
8  } from 'react-icons/fa';
9
10 function Footer() {
11   return (
12     <div className='footer-container'>
13       <div className='footer-links'>
14         <div className='footer-link-wrapper'>
15           <div className='footer-link-items'>
16             <h1>Contact Me!</h1>
17             <h4>Email: john_smith@gmail.com</h4>
18           </div>
19         </div>
20       </div>
21       <section className='social-media'>
22         <div className='social-media-wrap'>
23           <div className='social-icons'>
24             <Link
25               className='social-icon-link'
26               to={ 'https://www.github.com' }
27               target='_blank'
28               aria-label='Github'
29             >
30               <FaGithub />
31             </Link>
32             <Link
33               className='social-icon-link'
34               to={
35                 'https://www.youtube.com'
36               }
37               target='_blank'
38               aria-label='Youtube'
39             >
40               <FaYoutube />
41             </Link>
42             <Link
43               className='social-icon-link'
44               to={ 'https://www.linkedin.com' }
45               target='_blank'
46               aria-label='LinkedIn'
47             >
48               <FaLinkedin />
49             </Link>
50           </div>
51         </div>
52       </section>
53     </div>
54   );
55 }
```

2. Display your email by going to line 17 and changing the email address to your own
3. Add a link to your Github account by going to line 26 and adding your GitHub address

AFTER THE “//”

4. Add a link to your Youtube account by going to line 35 and adding your Youtube address
AFTER THE “//”
5. Add a link to your Github account by going line 44 and adding your Youtube address
AFTER THE “//”
6. Go to your web portfolio on your browser and see if the changes have been made.
 - a. Check to see if each of your social media links worked by pressing the social media icons from your browser



Congratulations! You just finished your first web portfolio!

Now personalize your web portfolio however you would like.

Our hiring team will get back to you as soon as possible!