

React and React Functional Components - Practice Exercises

Rebuilding apple.com's Homepage with React

Question: Build apple.com's homepage using React (functional component). We have already provided you with a replica of a responsive apple.com's website built using HTML and CSS/Bootstrap (from 2020). All you have to do is use the replica to build apple.com's website using React.

Hint on how to organize your folders and files

- In your Evangadi folder, create the folder “**Apple React Functional Component**”
- Create a folder called “**Resources**” in your “**Apple React Functional Component**” folder
- You will find a replica of a responsive apple.com's homepage built using HTML and CSS/Bootstrap [here](#). Download, this folder, unzip it and save it under your “**Resources**” folder
- Open your “**Apple React Functional Component**” in your VSC and open a new VSC terminal while on this folder
- Set up your React development environment by creating a React App named “**react-apple**”. In other words, run the following command on your terminal:
 - `npx create-react-app react-apple`
 - Wait until your it finishes creating your React App
- Now, change your directory to go to the React App that was just created. You can do this by typing the following command:
 - `cd react-apple`
- Run/ start your React app by typing the following command:
 - `npm start`
 - A new browser window will open up with your newly created React App. If not, open your browser and type localhost:3000 in the address bar

- **Note:** When running “npm start”, if at any time you get a message stating “...
*Something is already running on port 3000. Would you like to run the app on
another port instead? › (Y/n)...*”, please just type “y” without the quotations to
basically allow your React App to listen to another port
- Create a folder called, “**commonResource**” in your “**src**” folder
 - Go to your “**Resources**” folder, copy the “**css**”, “**js**” and “**images**” folders and
paste these folders into your “**commonResource**” folder
- Now, start creating different functional components under the “**src**” folder
 - **Note:** You will need to think how to divide your apple.com’s homepage section
by section and decide how many components you need before you start creating
components. For instance, you can organize your components like below:
 - Header.js - a component for your navbar/header
 - Main.js - a component for all the sections without the navbar and footer
 - You can also create different components for this by creating one
component for each section. **Example:** SectionOne.js,
SectionTwo.js, ...etc.
 - Footer.js - a component for your footer
 - Make sure all your components’ names start with Uppercase letters
- Open one of your components and start typing the code you need to create functional
components
 - To start, open one of your components and type the “**rfce**” abbreviation/snippet
without the quotation marks. This is basically a shortcut to create a functional
component structure/boilerplate.
 - **Note:** Make sure you have the “**ES7+ React/Redux/React-Native snippets**”
extension installed on your VSC to access the above shortcut from your
component.
 - The “**rfce**” shortcut will import the React library for you. In addition, it will create
your functional component and export this component for other components to
use it
- Use the “**index.html**” file from the Apple replica code to create your components. In
doing so, please watch out for the following:

- Make sure to rename all of your HTML “**class**” attributes with “className” in your components
- Make sure non-closing HTML tags we used in our replica code, such as <image>, and
 are closed when using them in your components
- Make sure all of your images are imported within the respective components
- Please remember to import the “**css**” folder we used in the replica into your components, preferably in your App.js
 - There are background images used when building the apple.com website using HTML/CSS. You have to make sure these images have the correct path if for any reason, you changed your folder/file structure
- Make sure to include/link Bootstrap and jQuery libraries in the “**index.html**” file found under the “public” folder. This is to allow your components to implement the features Bootstrap and jQuery provide for your Footer and Header component
- **Note:** You need to ALWAYS read the error on your terminal/browser, research the error online and fix it by yourself before asking anyone for help. You must develop the culture of searching for errors online and fixing your code issues by yourself

Happy coding 😊