

## Hands-On - #2

Q12. Your challenge!

Part 1: Create a page of your own using a custom function component by name 'Page'

It should return an ordered list with the reasons why you're excited to be learning React. Like

- \* React is flexible and efficient
- \* React developers are in demand
- \* React doesn't take long to learn

Ref: <https://www.codecademy.com/resources/blog/3-reasons-to-learn-react/>

Render your list to the page.

Q13. Your next challenge!

Part 2:

- Add a ``header`` element with a nested ``nav`` element. Inside the ``nav``, include a ``img`` element with the image of the React logo inside (`src="react-logo.png"`) and make sure to set the width to something more manageable so it doesn't take up the whole screen
- Add an ``h1`` with some text describing the page. (E.g. "Why learn React?").

Place it above the ordered list.

- Add a ``footer`` after the list that says:  
**"© 20xx <last name here> development. All rights reserved."**

You can decide on what names you would like to have for the above function components.

Render your function components

Q14. Let's implement Composability.

This question is a continuation of the previous challenge.

NOTE: Don't attempt this question if you have not done Q12 and Q13.

- [a] Move the 'header' element from the Page component into its own component called 'Header'
- [b] Move the 'footer' element into its own component called 'Footer'
- [c] Move the 'h1' and 'ol' elements together into another component called "MainContent"
- [d] Render the above components inside the Page component.

Identify the parent and the child components.

Q15. Your next Challenge!

- Add an ``ul`` inside the Header's ``nav`` and create the following ``li`s`:

"Tutorial", "About", & "Contact"

- The un-ordered list will be displayed right below the image with bullets
- We want provide styling for this un-ordered list and remove the list style. i.e. in other words set to 'none'

HINT: CSS style-sheets are static entities and created inside the 'public' folder.

- [a] Create a CSS file by name 'style.css' in the 'public' folder.
- [b] Link the CSS file in the 'index.html' file.
- [c] Give a className 'nav-items' to our un-ordered list
- [d] Set the 'list-style' property to 'none' for the

`'nav-items'` class in the CSS file

Save the `style.css`, `index.html` and the `index.js` files

Refresh the check the change in your browser. The bullets for the un-ordered list should have now gone.

- Using flexbox, line up the nav items horizontally, and put them inline with the React logo.

- [a] Set the `'display'` property to `'flex'`
- [b] Set the `'justify-content'` property to `'space-between'`
- [c] Align items to `'center'`

NOTE: You should only use `'className'`

- Ensure we have a padding of 10px for list items
- Change the image styling to happen in CSS instead of in-line

For practice, add a new class to the image in order to style it

Set the image width to `'60px'`

- Change the footer style, such that the following properties are set

```
position: fixed;
bottom: 0;
background-color: grey;
width: 100%;
height: 40px;
```

Save the changes made to the files and refresh the browser.

## Q16. Organizing Components

- [a] To organize our components, let's first create a folder by name `'components'` inside our `'src'` directory.

[b] Create a file by name 'Header.js' inside the 'components' folder  
Copy the Header() function from the 'index.js' file.

Don't forget THREE things:

- (1) To import the 'React' module in the component's file
- (2) To export the component(s)
- (3) To import the component in the 'index.js' file

Best practices:

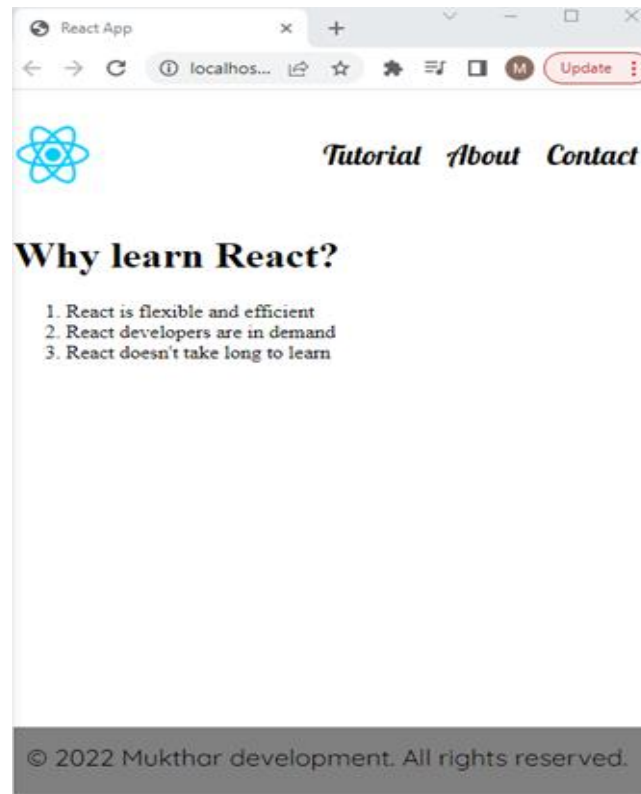
- \* The name of the JS file should be same as the component name
- \* The filename should be Pascal Case

[c] Save your work and check if the app is functional.

[d] You next Challenge:

Move the Footer and MainContent components into their own files.

[e] Save your work and check if the app is functional.



Q17. It is time now to do a sample React application which will help us in consolidating the topics which we have covered so far.

You are supposed to design an app which looks like the one shown below.

## Fun facts about React

- Was first released in 2013
- Was originally created by Jordan Walke
- Has well over 100K stars on GitHub
- Is maintained by Facebook
- Powers thousands of enterprise apps, including mobile apps

This resembles, something similar to what is done in the previous set of questions.

**NOTE:** It is not necessary, that it should look exactly the same. However, if it resembles something similar is fine.

You are at liberty to use your own font, styles.