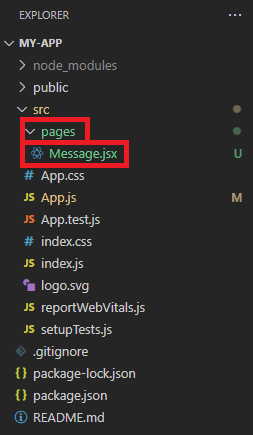
Step 1: First component

# Introduction:

In this first step we’re going to better learn the basics of React by creating our **first component** and rendering it in App.js.

Start by creating a new folder called **pages**. Inside that folder create a new file called **Message.jsx**.



First of all, you have to **import** **React** in your file like shown below. **Imports go to the very top of the file!**

import React from "react";

In React you can render multiple components inside of other components. The root component is usually named App. Components are mainly in their own files which are imported to the display component one by one.

A very important thing to remember while creating components is that you **must** export the component, otherwise you can’t render it. You need to import every component to you App.js file, kind of like you imported React. Example below:

import { Example } from "./pages/Example";

# First component

Now we’ll start writing our first component. A very bare-bones component is an **arrow function with React JSX code as the return value**. More complex components might have JavaScript code for things like data handling. Start by creating an **exportable** arrow function, named **Message**.

export const Message = () => {

    return (

    );

}

Like so. Next, **inside the return tags** create a normal HTML div element with header **Hello World!**.

export const Message = () => {

    return (

        <div>

            <h1>Hello World!</h1>

        </div>

    );

}

We have now created our first working component! Now we’ll move to the App.js file and **render it**.

# Rendering:

Your App.js file should look like this. What we have here is our empty **root component**. Root component is the one responsible of rendering your code. Just like in our previous file, inside the return tags is a div element.

import React from "react";

function App() {

    return (

        <div>

        </div>

    );

}

export default App;

**Remember to import** the component at the top of the file.

import { Message } from "./pages/Message";

Next you have to go inside your div tags and render the component. It’s simple, just write your component name inside HTML tags, kind of like a HTML element, just like shown below.

<div>

    <Message />

</div>

Now our component should display the text inside the message component. Launch your developer server (**developer server being the local web page that we launch with cmd**) and take a look. It should look something like this:

Graphical user interface, text, application, chat or text message

Description automatically generated

Now we’ve learned a bit about the basics of React and how to render content, in the next exercise we’ll create a second component that has a bit more functionality. **See you there!** 😊