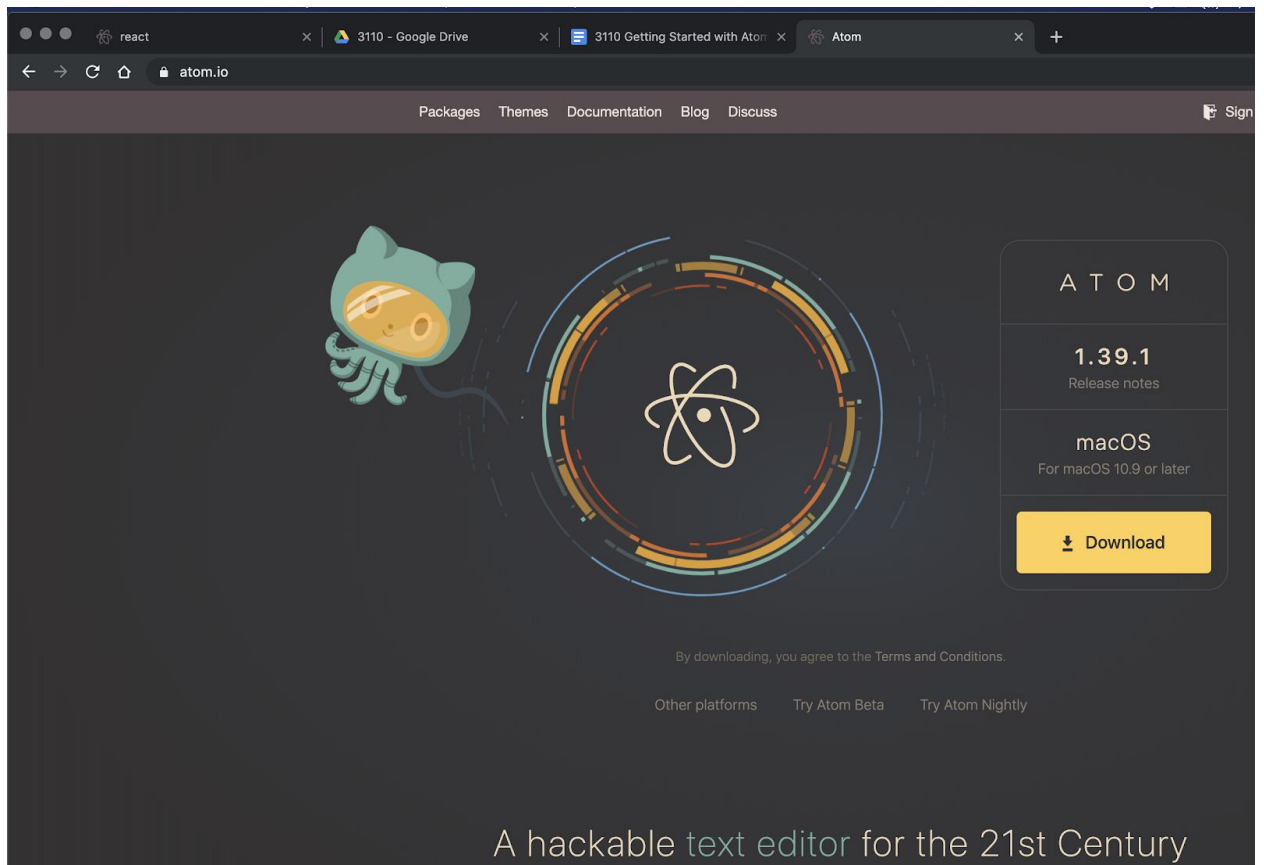
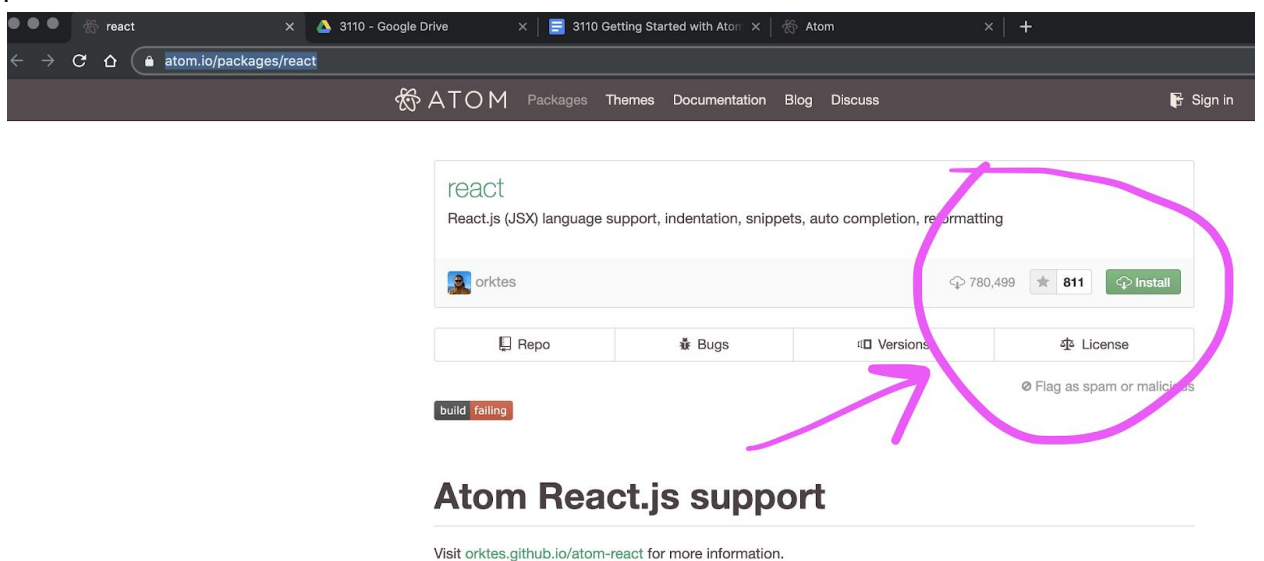


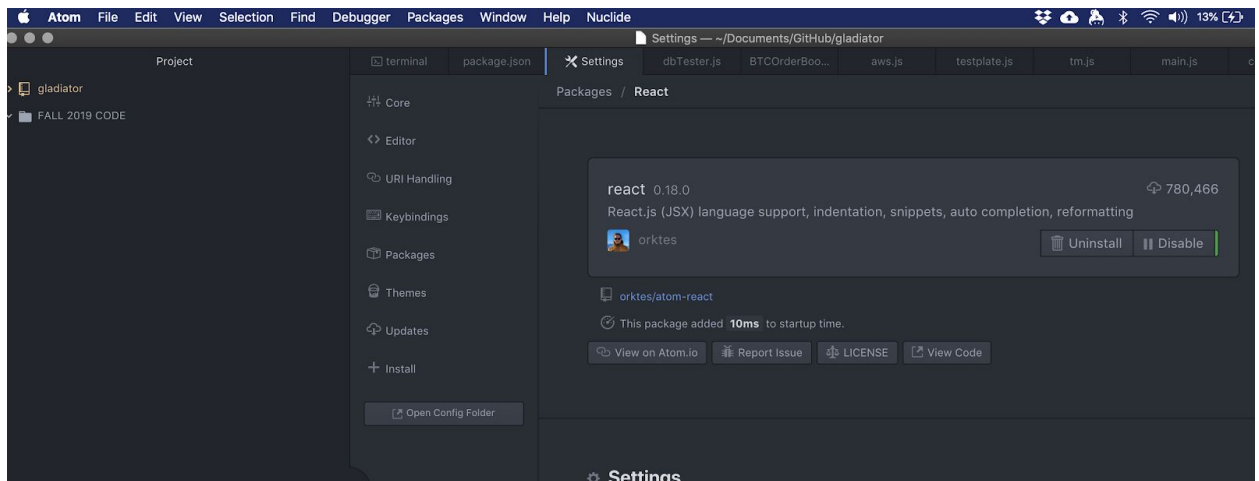
1. Get atom IDE at <https://atom.io/> :



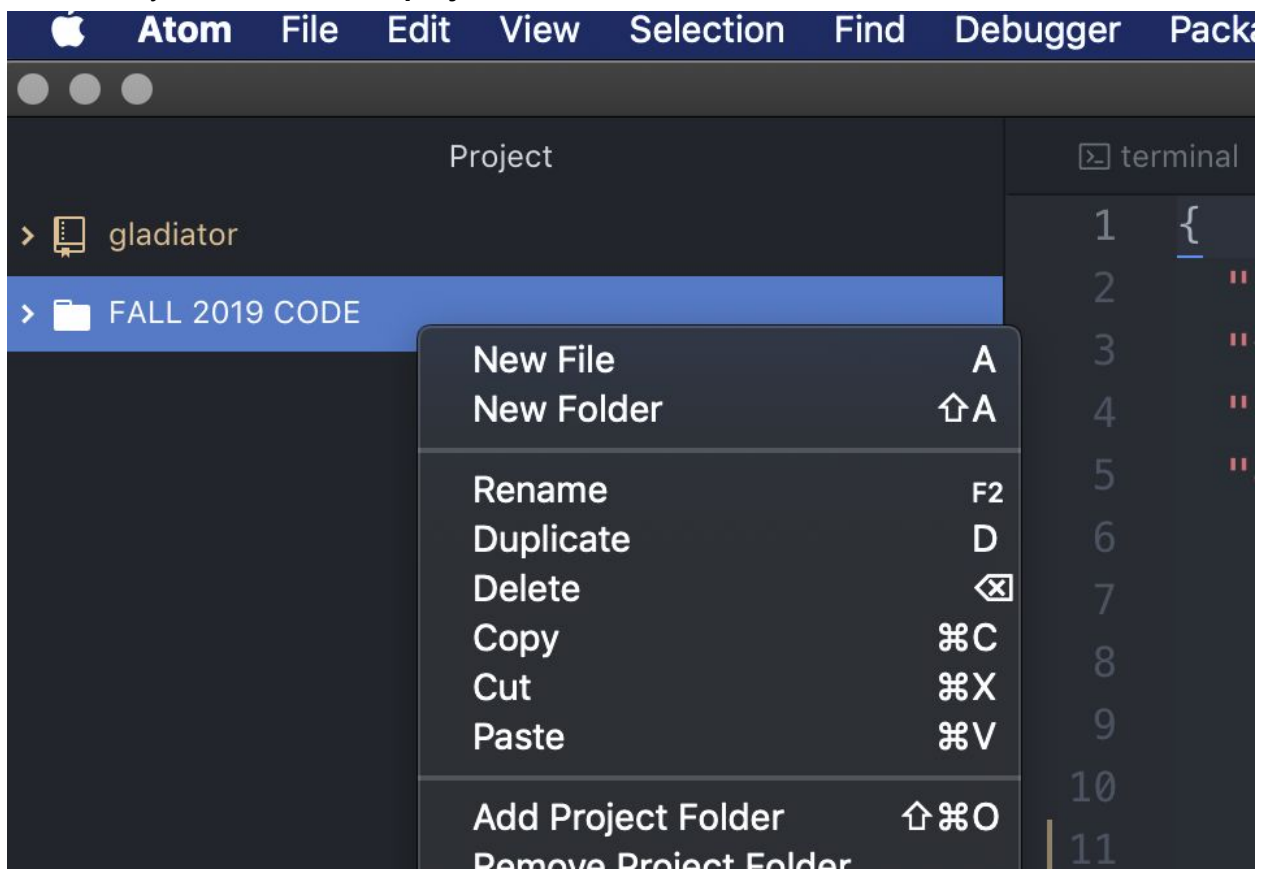
2. Install atom's package for react.js by visiting <https://atom.io/packages/react> , then press Install button:



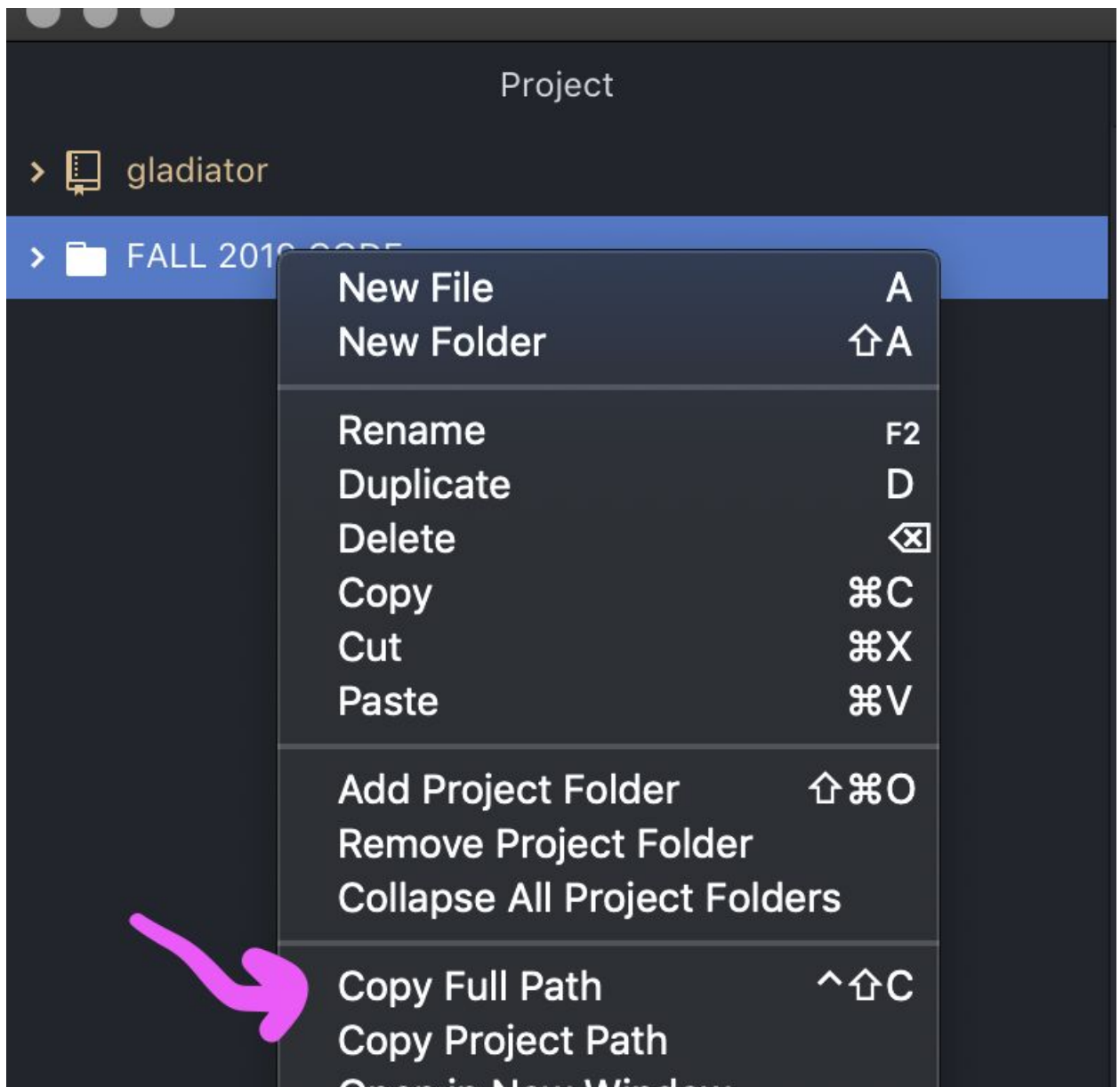
3. The install button will open atom ide, then begin the installation of the react.js package, which helps debug and build react.js programs.



4. In atom, you will **add a new project folder**



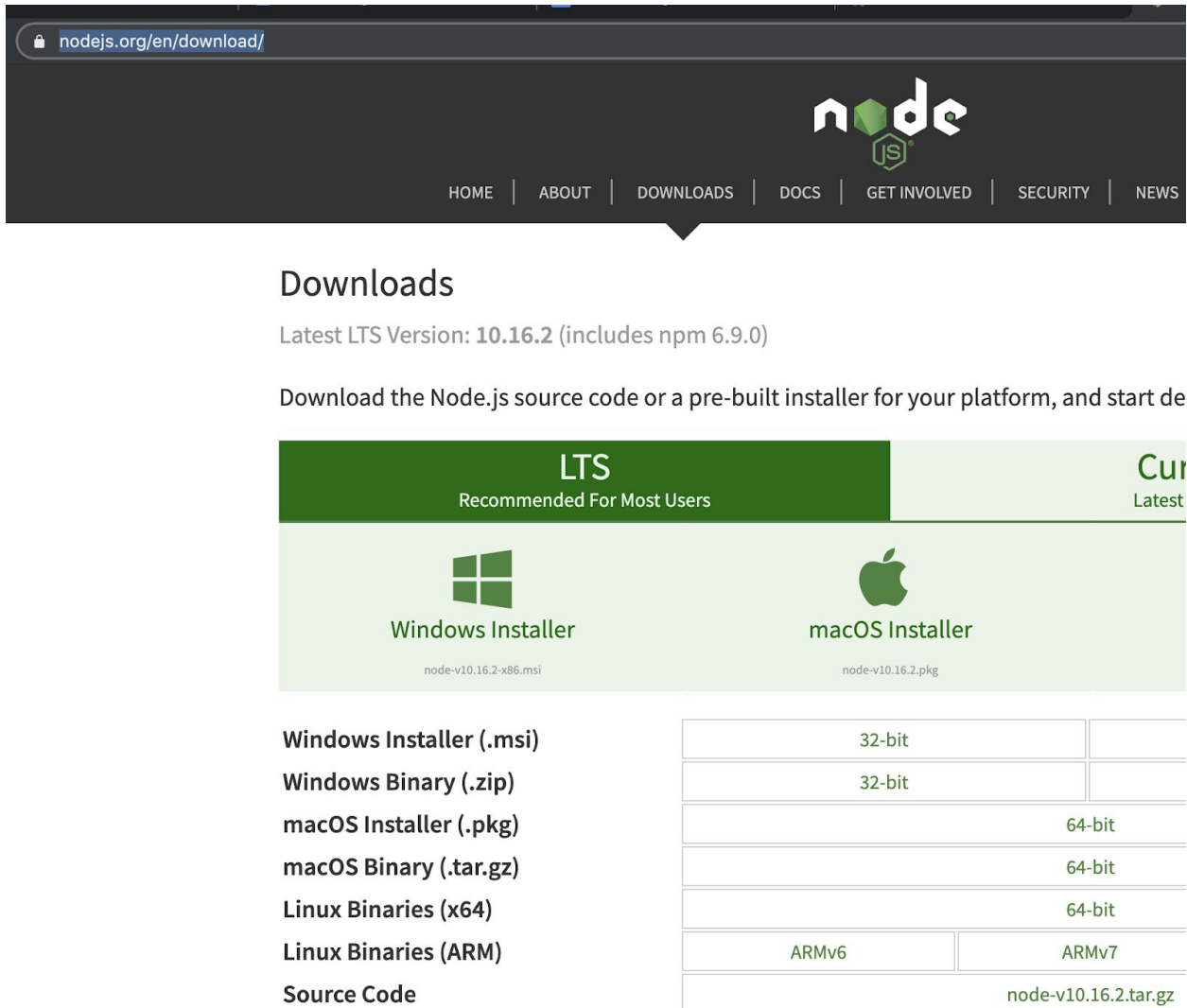
5. Then navigate to that new folder in your terminal or command prompt window, after getting the absolute pathname, or exact location of the new folder on your filesystem. Use the copy full path command in atom, after right clicking the new folder.



6. Once you've copied the full path, or absolute path name from atom, you can navigate to the folder using your terminal. Just open the command prompt, then copy/paste the path, after typing 'cd' into the command prompt or terminal window cli

```
FALL 2019 CODE — ec2-user@ip-172-30-0-226:~ — bash — 164x39
stefans-MacBook-Pro:aws ec2 maximus key pair stefanbund$ cd /Users/stefanbund/Documents/GitHub/FALL*
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ pwd
/Users/stefanbund/Documents/GitHub/FALL 2019 CODE
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$
```

7. Time to install node.js, which is a framework for javascript that helps bundle react, and enable react development. Node is available at <https://nodejs.org/en/download/> and you will download the proper installer for your OS:



The screenshot shows the Node.js download page for the LTS version. The page has a dark header with the Node.js logo and navigation links: HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, and NEWS. Below the header, the 'Downloads' section is highlighted. It states 'Latest LTS Version: 10.16.2 (includes npm 6.9.0)' and 'Download the Node.js source code or a pre-built installer for your platform, and start de'. The main content area is divided into two columns: 'LTS Recommended For Most Users' and 'Current Latest'. Under the LTS column, there are two installers: 'Windows Installer' (node-v10.16.2-x86.msi) and 'macOS Installer' (node-v10.16.2.pkg). Below these, a list of download options is provided: Windows Installer (.msi), Windows Binary (.zip), macOS Installer (.pkg), macOS Binary (.tar.gz), Linux Binaries (x64), Linux Binaries (ARM), and Source Code. To the right of this list is a table showing the available binaries for different architectures.

32-bit	
32-bit	
	64-bit
	64-bit
	64-bit
ARMv6	ARMv7
	node-v10.16.2.tar.gz

8. Run this command, once you've installed node. You'll need to be in administrator to install on windows, or login to your root account, using sudo, if you're on mac or linux.

```
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ sudo npm install -g create-react-app
Password:
/usr/local/bin/create-react-app ->
/usr/local/lib/node_modules/create-react-app/index.js
+ create-react-app@3.0.1
added 33 packages from 29 contributors, removed 9 packages and updated 22 packages in
6.576s
```

Sudo: helps you login as root for a limited set of commands, necessary for creating the installable's directories, and enabling execution of the npm installer.

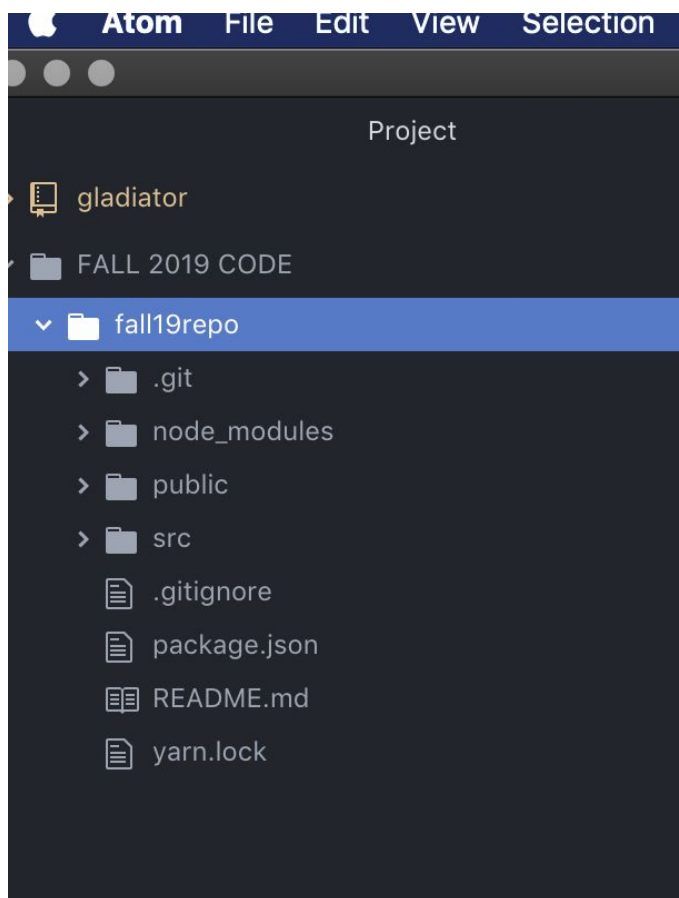
Npm: node package manager, responsible for installing node libraries, extends your javascript programming capabilities

9. Run create-react-app, which initializes a project workspace for react projects.

```
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ npx create-react-app fall19repo
```

Fall19repo is the name of the project we'll build.

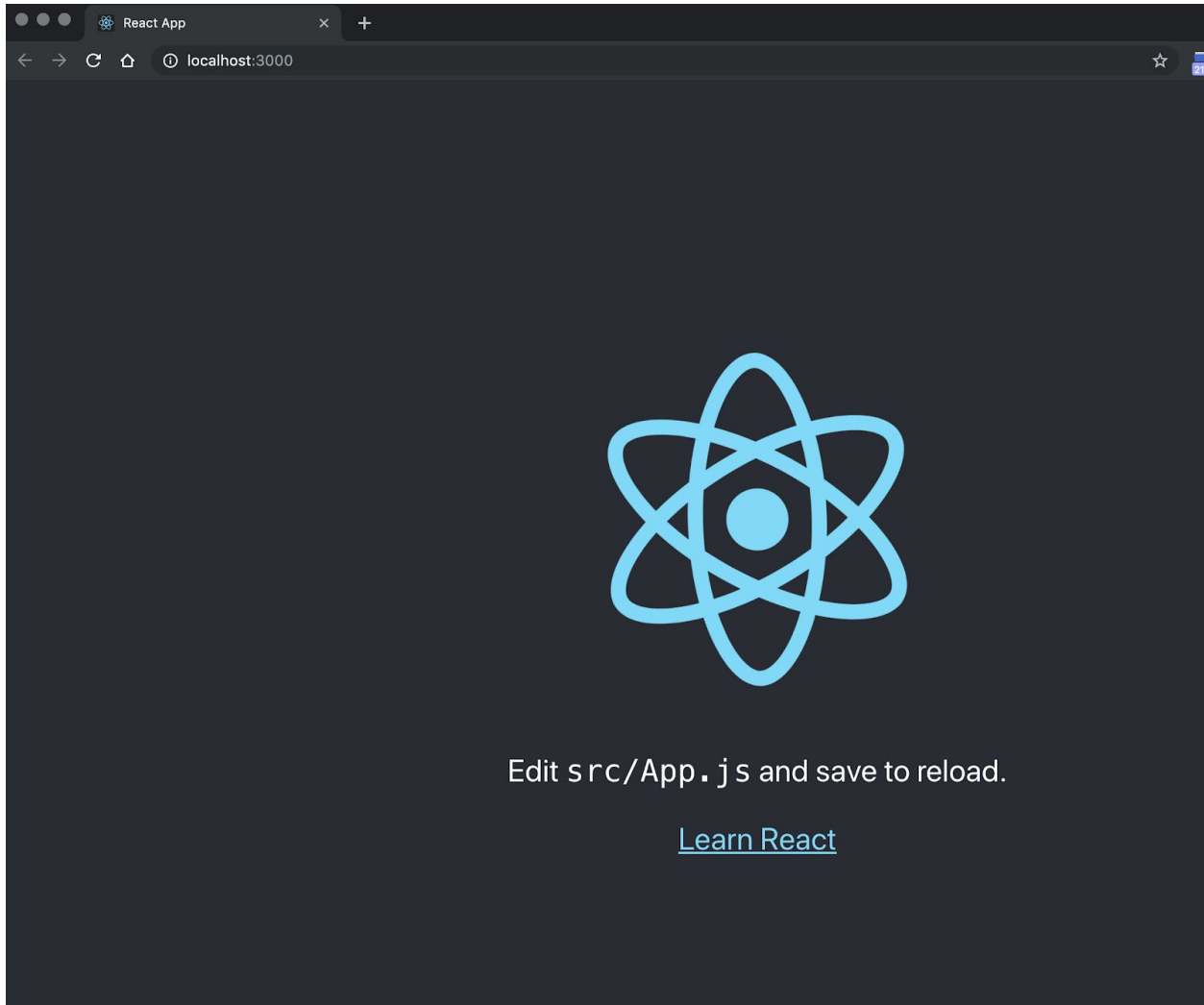
10. Return to atom: you now have a populated project workspace, ready for coding react.js apps. Congrats in order/Klondike bar incoming



11. Time to run react's web server, and observe this project online, using your local machine:

```
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ pwd
/Users/stefanbund/Documents/GitHub/FALL 2019 CODE
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ ls
fall19repo
stefans-MacBook-Pro:FALL 2019 CODE stefanbund$ cd fall19repo
stefans-MacBook-Pro:fall19repo stefanbund$ npm start
```

Notice how I navigate to the new react project, formed underneath our initial directory. These commands work because they are done within the project folder (as shown -- I navigate using `cd`, then verify my position using `pwd`, then a quick `ls`, to verify I am where I need to be). Npm will trigger react's web server, and deploy my current work to a live web server:



Here is what your terminal will show:
Compiled successfully!

You can now view **fall19repo** in the browser.

Local: `http://localhost:3000/`
On Your Network: `http://192.168.2.13:3000/`

Note that the development build is not optimized.
To create a production build, use `yarn build`.

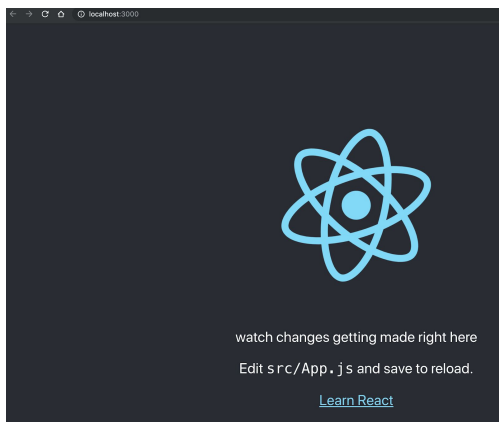
It should spin up and open an instance of your default browser, after asking your permission.

12. Navigate to the /src folder, then open App.js in atom.
Adjust line 11 so it reads as so:

```
import React from 'react';
import logo from './logo.svg';
import './App.css';

function App() {
  return (
    <div className="App">
      <header className="App-header">
        <img src={logo} className="App-logo" alt="logo" />
        <p>
          <p>watch changes getting made right here </p>
          Edit <code>src/App.js</code> and save to reload.
        </p>
        <a
          className="App-link"
          href="https://reactjs.org"
          target="_blank"
          rel="noopener noreferrer"
        >
          Learn React
        </a>
      </header>
    </div>
  );
}

export default App;
```



You can see it register changes when you save App.js in atom, then reselect the browser. It will automatically reload the changed code.

From here we will create new react project folders, each with a full configuration, which will demonstrate new concepts.