WORKSHOP #4

CREATE-REACT-APP

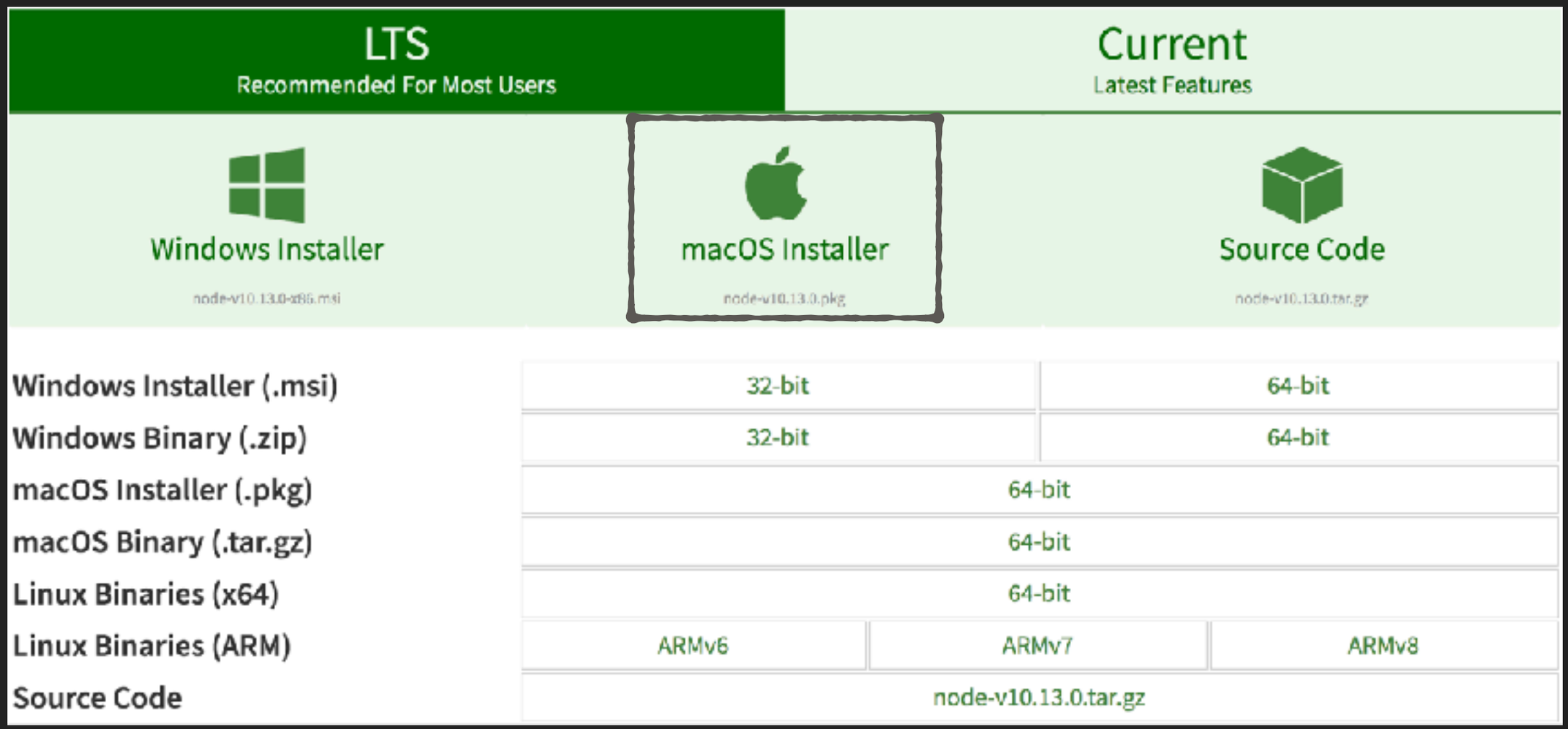
LETS BECOME FRIENDS WITH THE COMMAND PROMPT.

- Open the spotlight search and type in “terminal”
- This should open a lovely command prompt that will allow us to do all sorts of fun things.
- We first want to check if we have node installed. Type “npm -version” without the quotes in your command prompt. If we see a version that is above 5.0 we’re all set.
- If not we will see and error saying “npm is not a recognized program”, and we’ll want to follow the directions in the next few slides.

INSTALLING NODE

If node isn't installed on our computers, we need to navigate to https://nodejs.org/en/download/

We want to click on the macOS button highlighted below



INSTALLING NODE

- Once we click on the macOS button a download should start. Run the file once the d/l has completed.
- Follow along the prompts to install node on your computer. Once the installer is complete it might have you restart your computer (hopefully not).
- Close any command prompt windows, and reopen a new one, and type in “npm -version” without the quotes. Now we should see a version number and this will allow us to install packages with the “Node Package Manager”.

INSTALLING CREATE-REACT-APP

- So now we want to set up our create-react-application to be installed on our computer. This will allow us to use the create-react-app command to create projects and install the correct files necessary for our react application.

- In our command prompt, we want to make a directory that we will install our react applications into. First we need to navigate into our Documents folder. We do this by using the cd command (which stands for Change Directory).

- The format is cd directoryNameGoesHere.

- From the command prompt type “mkdir react-applications” without the quotes.

CREATE-REACT-APP

- Now that we have a directory that we can install our react applications. Lets create our project.
- In our command prompt make sure we are in our react-applications folder by using the command “cd” - this stands for change directory. The command “cd react-applications” should bring us into the correct directory.
- Now lets type “create-react-app ourprojectnamegoeshere”
- It’ll take a minute to set up the correct files.

- Once your installation is complete you should see a suggested message to change your directory into your project name, and run the command “npm start”. Follow these directions.

```
We suggest that you begin by typing:

  cd myproject
  npm start

Happy hacking!
```

- If successfully complied you should see this in your console.

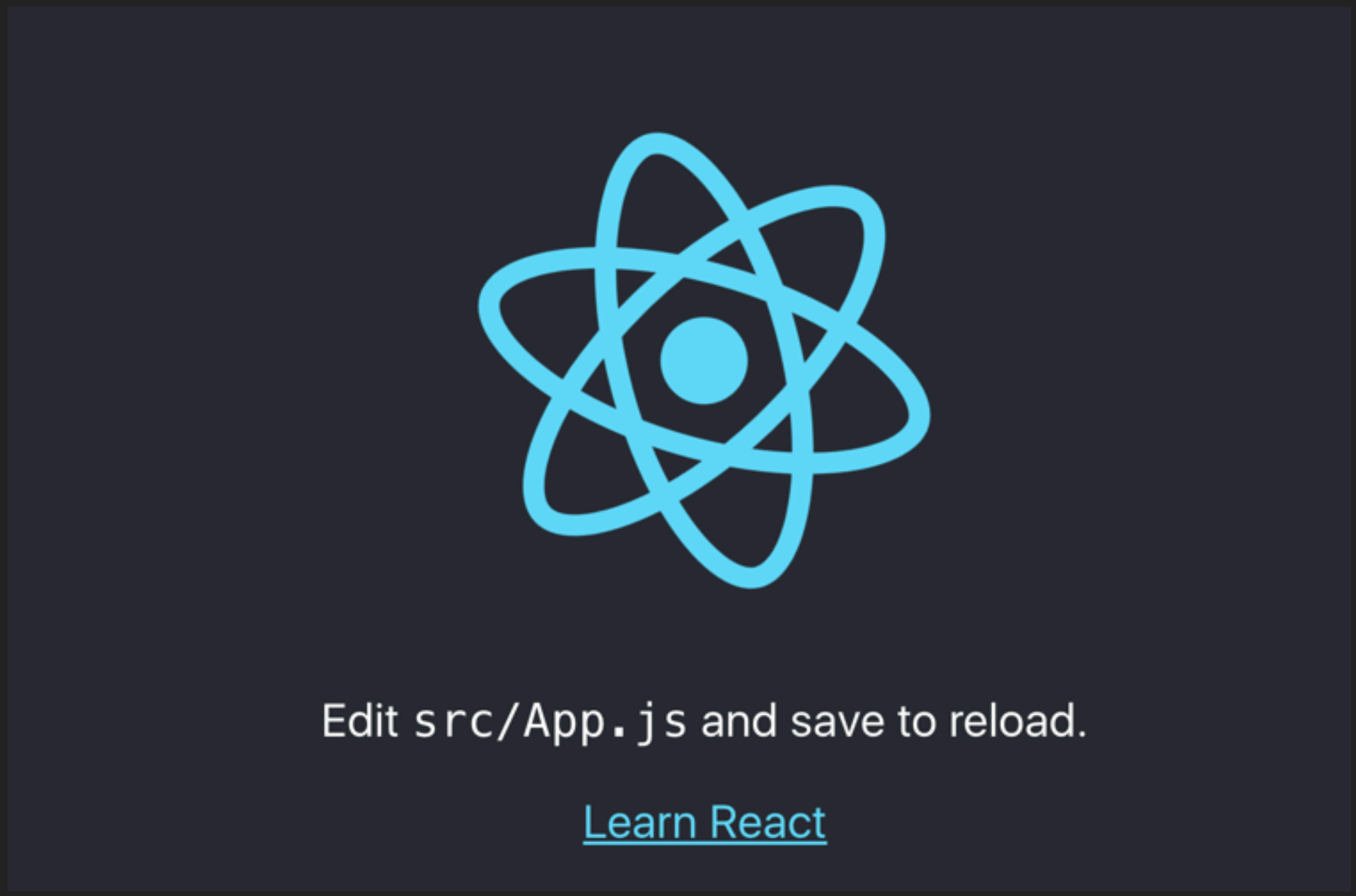
```
Compiled successfully!

You can now view myproject in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://192.168.1.183:3000/
```

CREATE-REACT-APP

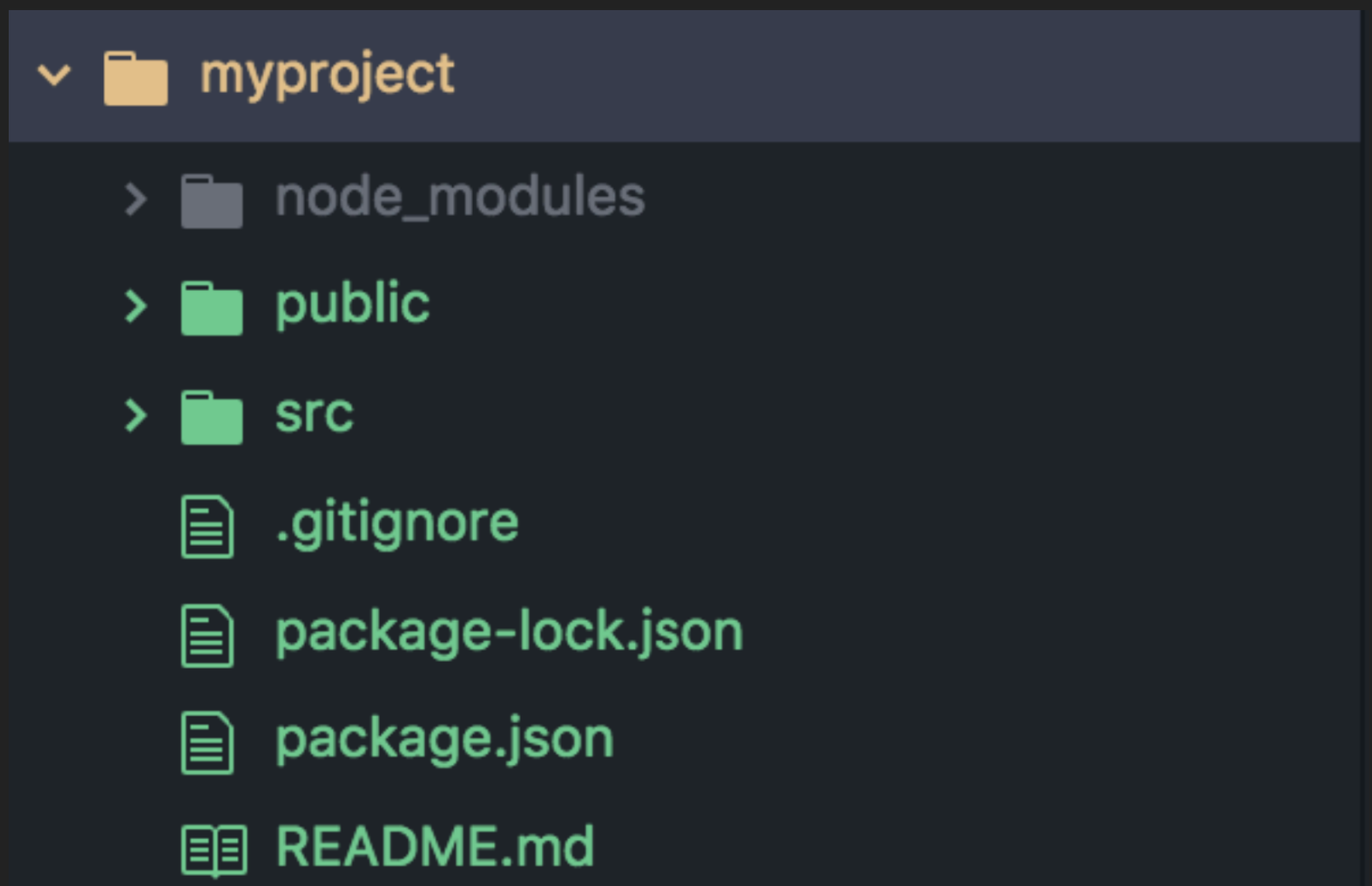
If you open your browser and go to http://localhost:3000/ you should see the create-react-app boilerplate page loading. (It should do it automatically when you first start your server.)



CREATE-REACT-APP

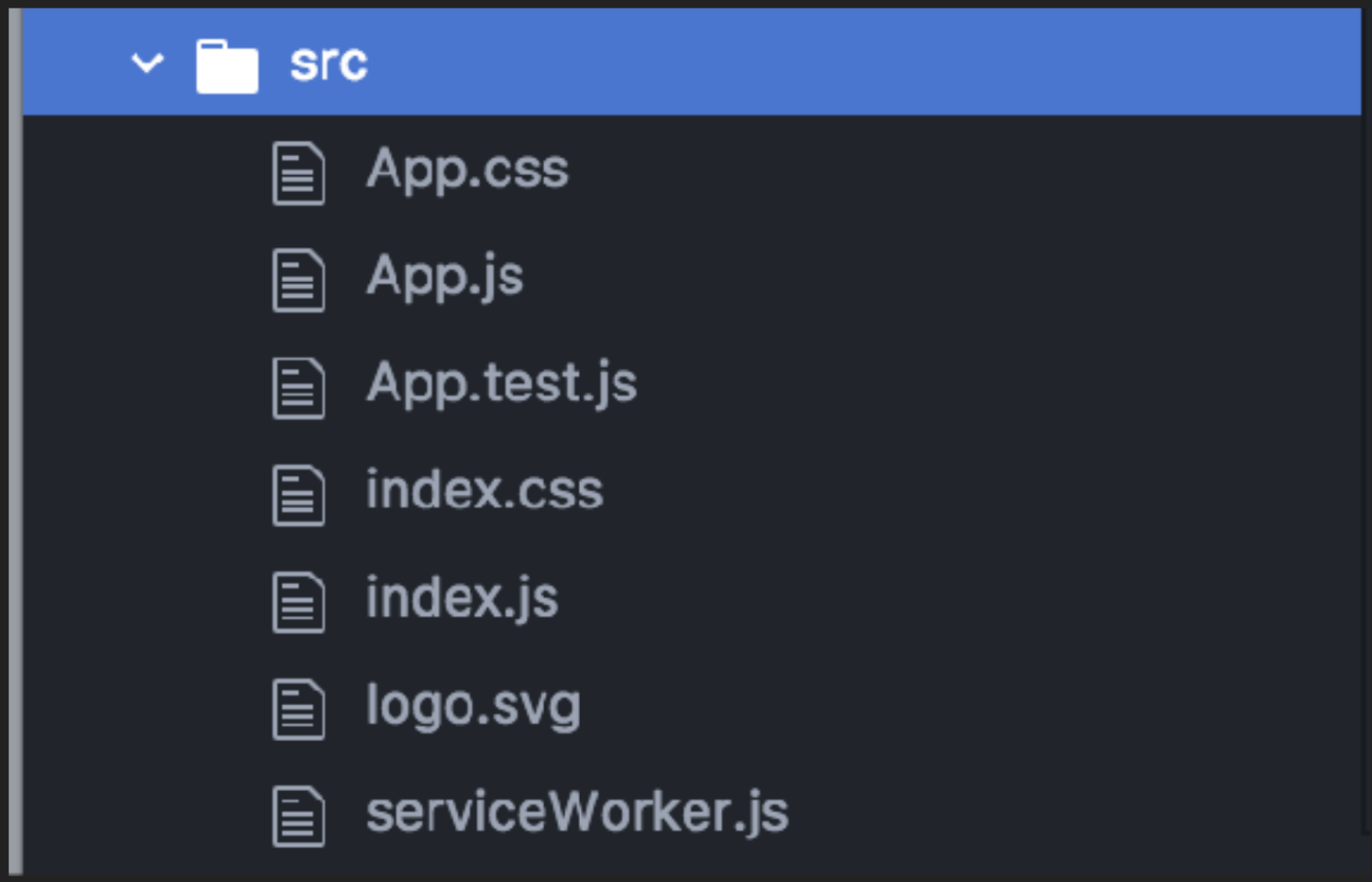
- Now that we have our application initialized. Lets open it up and check out the directory structure that create-react-app initially gives us. (Here is where you want to open the project in your local IDE to see the directory structure)

```
my-app
├── README.md
├── node_modules
├── package.json
├── .gitignore
├── public
│   ├── favicon.ico
│   ├── index.html
│   └── manifest.json
└── src
    ├── App.css
    ├── App.js
    ├── App.test.js
    ├── index.css
    ├── index.js
    ├── logo.svg
    └── serviceWorker.js
```



This is what the directory structure will look like in the atom IDE.

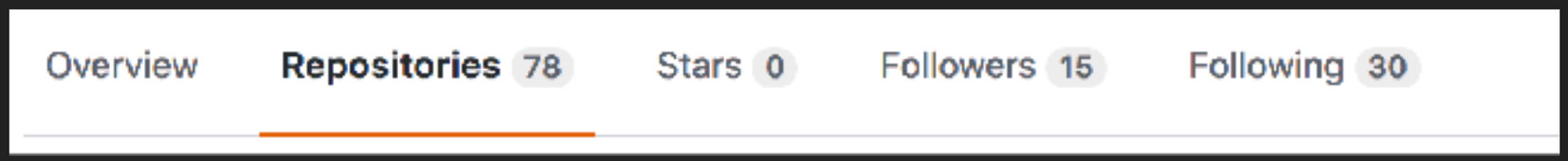
CREATE-REACT-APP



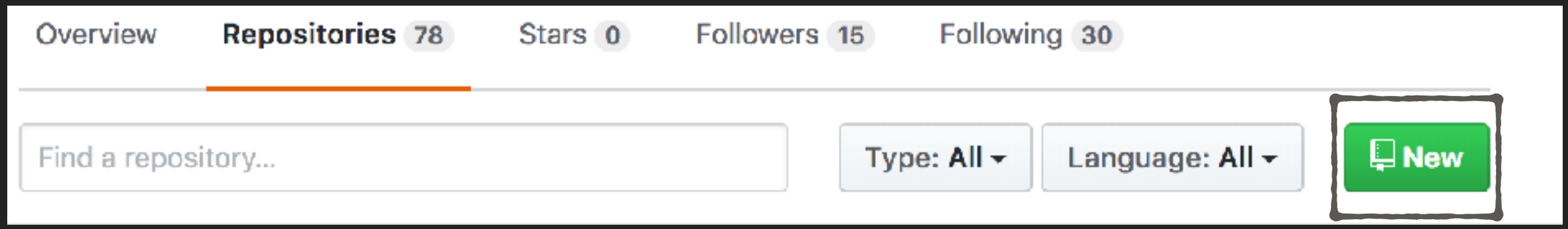
Our src directory is where we are going to be working 99% of the time.

This is where our main App.js file is as well as where we would want to create our component JS files.

- Now that we have our boilerplate and directory structure created. We want to add this to a repository on github. This way we start using version control to handle our projects.
- First we have to create this repo on github. Navigate to GitHub.com and log in.
- Then we want to navigate to our repositories tab at the top of your page

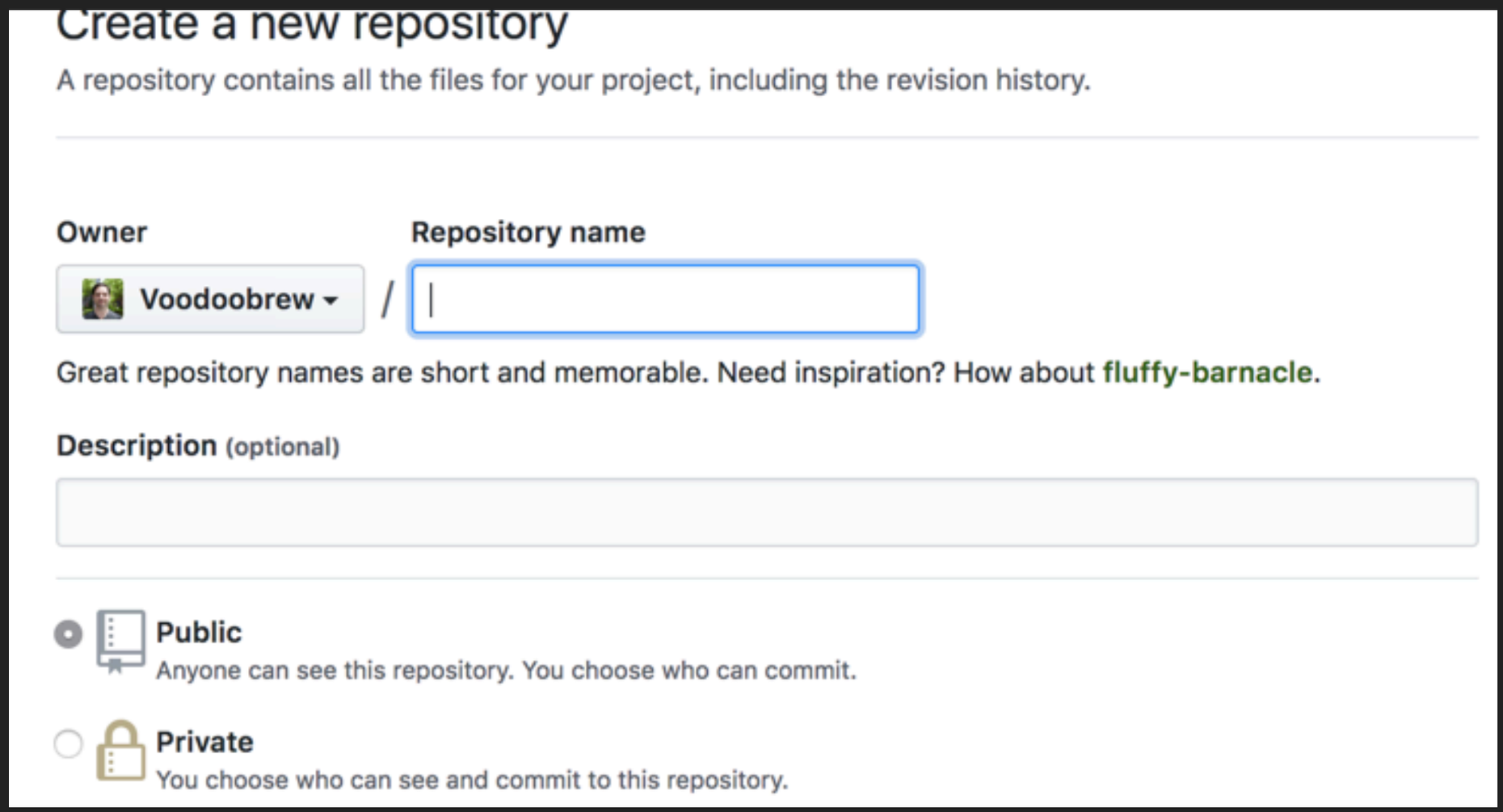


- We want to create a new repository by clicking the “New” button pictured.

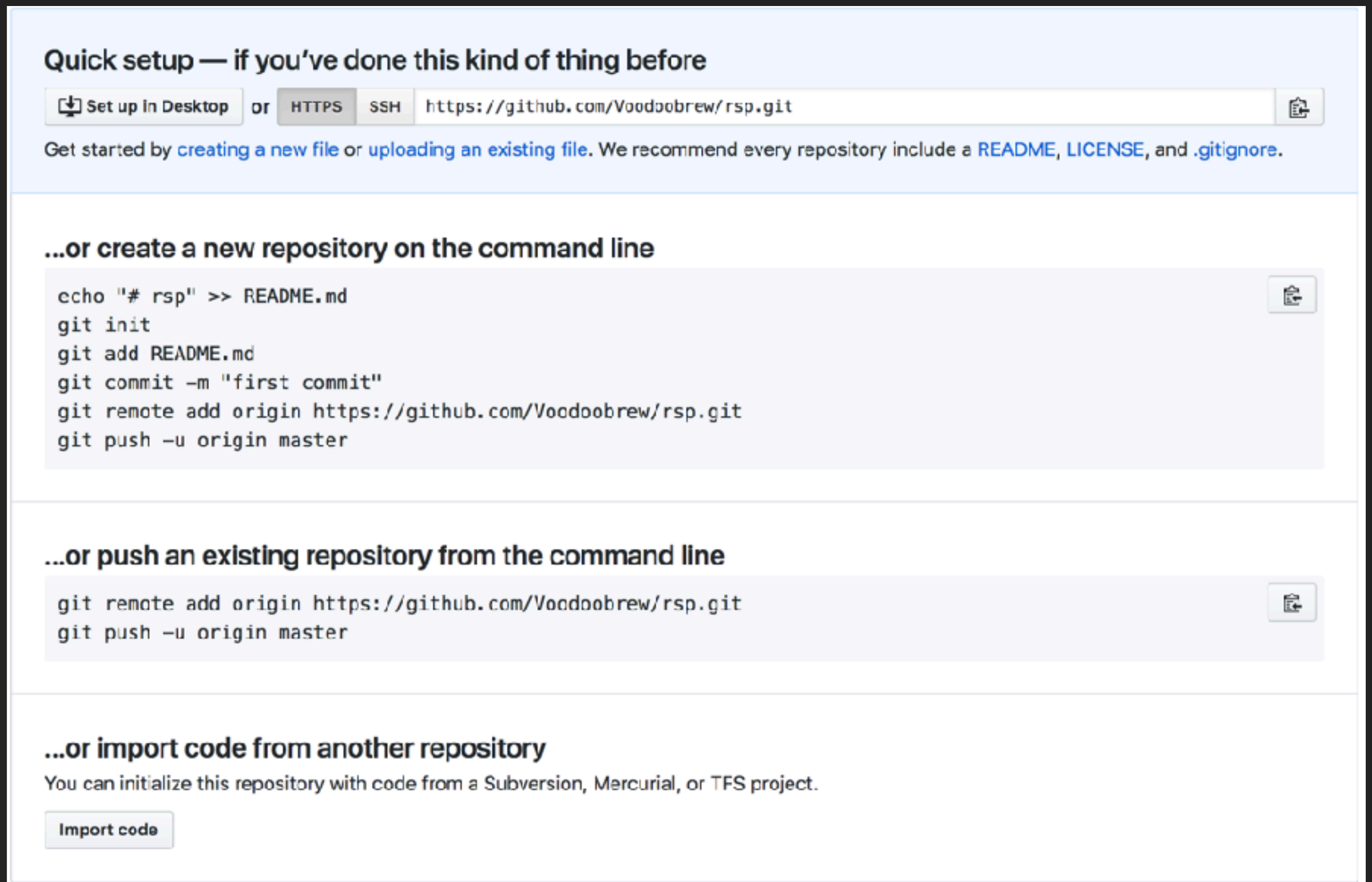


- Then we want to give our repo a declarative name that describes our project.

- Now lets give our repo a name of rock-scissors-paper in the repository name field.



- We should see this page if setup correctly.



Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH https://github.com/Voodoobrew/rsp.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# rsp" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/Voodoobrew/rsp.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/Voodoobrew/rsp.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

Import code

- Lets navigate to our react-applications directory and make sure we change directories into our projects folder.
- Now we want to initialize the project by typing in **git init** into our command prompt.
- After our repository is initialized. We want to add in all of our files that we have so far. Run the **git add .** (that's a period after the add word to make sure it adds everything in the directory initialized).
- Then we want to commit these files to our repo by running the command **git commit -m "First Commit"**

- Now we need to use that link up in the quick setup to add into our remote repository.
- Now we need to add the remote repository. Copy the link that is specific to your newly create repository. The command starts with "git remote add origin" followed by your specific link given in the Quick Setup guide. Example:

```
git remote add origin https://github.com/Voodoobrew/rsp.git
```

- Then we want to push these files to our repo's master branch. In the command prompt run
  git push -u origin master

- Here are some useful links to reference:
  https://help.github.com/articles/adding-an-existing-project-to-github-using-the-command-line/

  https://github.com/facebook/create-react-app