

WORKSHOP #4

# CREATE-REACT-APP

---

# LETS BECOME FRIENDS WITH THE COMMAND PROMPT.


- ▶ Open the command prompt by clicking in the search bar and typing "cmd" no quotes.
- ▶ 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 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 Windows Installer button highlighted below

LTS  
Recommended For Most Users

  
Windows Installer  
node-v10.13.0-x86.msi

Current  
Latest Features

  
macOS Installer  
node-v10.13.0.pkg

  
Source Code  
node-v10.13.0.tar.gz

Windows Installer (.msi)

Windows Binary (.zip)

macOS Installer (.pkg)

macOS Binary (.tar.gz)

Linux Binaries (x64)

Linux Binaries (ARM)

Source Code

32-bit	64-bit	
32-bit	64-bit	
64-bit		
64-bit		
64-bit		
ARMv6	ARMv7	ARMv8
node-v10.13.0.tar.gz		

# INSTALLING NODE

- ▶ Follow along the prompts to install node on your computer. Once the installer is complete it might have you restart your computer (hopefully not).
- ▶ When all is said and done we want to go back into our command prompt 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"

- ▶ So now we want to set up our create-react-application to be installed globally. 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. Lets call this directory "react-applications".
- ▶ From the command prompt type "mkdir react-applications" without the quotes.

- ▶ Now that we have a directory that we can install our react applications. Lets create our rock scissor paper 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 rockscissorspaper"
- ▶ It'll take a minute to set up the correct files.

## 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
```


# CREATE-REACT-APP

---

## ▼ rockscissorspaper

>  .git

>  node\_modules


>  public

>  src

 .gitignore

 package-lock.json

 package.json

 README.md

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



# CREATE-REACT-APP

---

## src

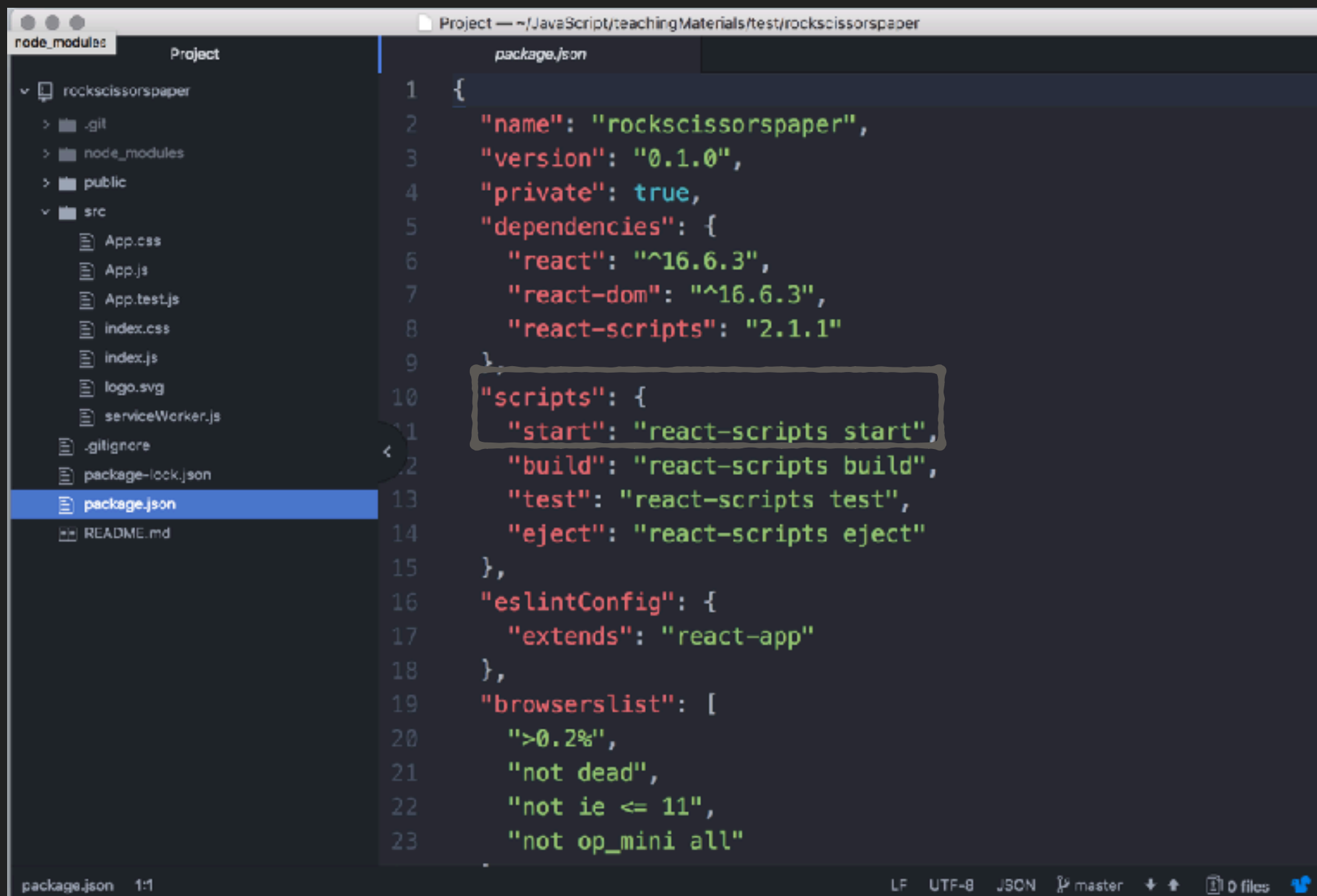
- App.css
- App.js
- App.test.js
- index.css
- index.js
- logo.svg
- serviceWorker.js

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.

# CREATE-REACT-APP

- Now its time to start a server. Select the package.json file.



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with a file named `package.json` selected. The code editor displays the content of `package.json`, which is a JSON object. A box highlights the `"scripts"` section of the file.

```
1 {
2   "name": "rockscissorspaper",
3   "version": "0.1.0",
4   "private": true,
5   "dependencies": {
6     "react": "^16.6.3",
7     "react-dom": "^16.6.3",
8     "react-scripts": "2.1.1"
9   },
10  "scripts": {
11    "start": "react-scripts start",
12    "build": "react-scripts build",
13    "test": "react-scripts test",
14    "eject": "react-scripts eject"
15  },
16  "eslintConfig": {
17    "extends": "react-app"
18  },
19  "browserslist": [
20    ">0.2%",
21    "not dead",
22    "not ie <= 11",
23    "not op_mini all"
24  ]
25 }
```

- ▶ We have a script that will allow us to locally run our application, and we run this script by typing in our project directory “npm run start”
- ▶ If successfully compiled you should see this in your console.

```
Compiled successfully!
```

```
You can now view rockscissorspaper in the browser.
```

```
Local: http://localhost:3000/
```

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

```
Note that the development build is not optimized.
```

```
To create a production build, use npm run build.
```



## CREATE-REACT-APP

---

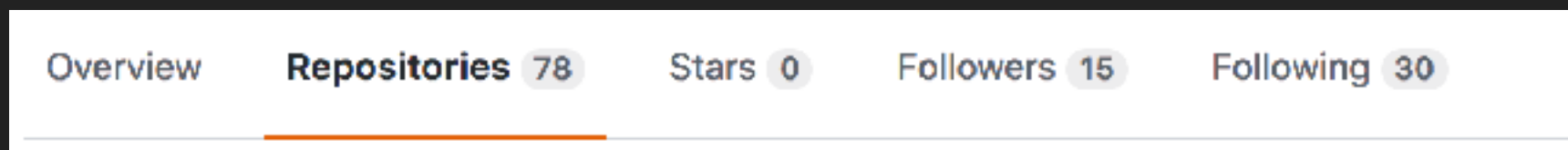
If you open your browser and go to <http://localhost:3000/> you should see the create-react-app boilerplate page loading.



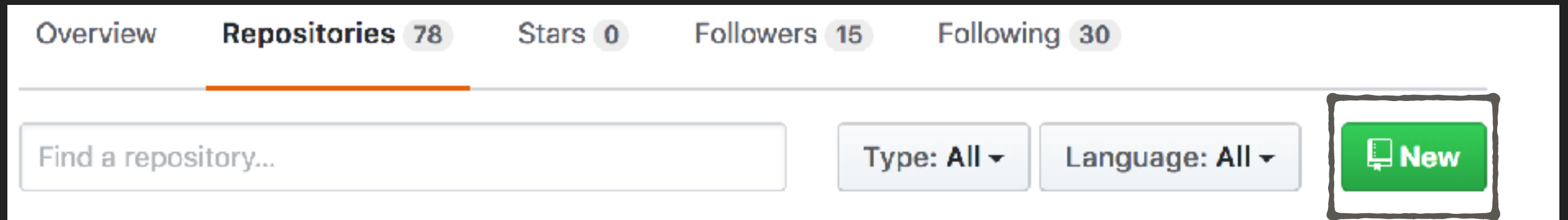
Edit `src/App.js` and save to reload.

[Learn React](#)

- ▶ 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](https://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.

### Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name

 Voodooobrew ▾

 /

Great repository names are short and memorable. Need inspiration? How about **fluffy-barnacle**.



Description (optional)

☒  **Public**  
Anyone can see this repository. You choose who can commit.

☐  **Private**  
You choose who can see and commit to this repository.

- ▶ 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 rockscissorspaper 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.
- ▶ You want to run the command.  
**git remote add origin <https://github.com/YOURNAMEHERE/rock-scissors-paper.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>