

Installing ReactJS and creating your first application

To install and use ReactJS, we need two main dependencies: **Node.js** and **NPM**. **Node.js** is a Javascript run-time environment that allow us to execute Javascript code like if we were working on a server.

NOTE: That every web application is meant to be executed in a server if we work on computers.

And second is **NPM** is a package manager for Javascript, that is, **NPM** allows us to install Javascript libraries.

Now we are going to see how to install **react js**

These instructions are for Linux systems.

Step:1

First, we need a tool called curl, if you don't have it, you can install it typing the following command in a terminal:

```
sudo apt-get install curl
```

You are going to be prompt to enter your user password to grant sudo access. After the installation is finished, now let's install Node by typing the following command:

Step:2

```
curl -sL https://deb.nodesource.com/setup\_6.x | sudo -E bash -
```

I'm going to install Node.js v6 LTS Version. If you want, you can install the newest version (v8) by replacing setup_6.x for setup_8.x. Either version will work fine.

Step:3 The command you just type added the Node.js to your repository list, now, let's install Node.js by typing:

```
sudo apt-get install -y nodejs
```

Step:4 If everything is alright, you can check that **Node.js** and **NPM** are installed (NPM is included with Node.js) by typing **node -v** and **npm -v**, and you should see something like this:

```
swecha@guptam-lenovo-g50-80:~$ node -v
```

v6.11.1

```
swecha@guptam-lenovo-g50-80:~$ npm -v
```

3.10.10

Step:5 Also, let's install build-essential package, which is needed for Node to work properly:

```
sudo apt-get install -y build-essential
```

Step:6 Now, let's install React by typing:

```
sudo npm install -g create-react-app
```

```
swecha@guptam-lenovo-g50-80:~$ sudo npm install -g create-react-app
[sudo] password for swecha:
/usr/local/bin/create-react-app -> /usr/local/lib/node_modules/create-react-app/index.js
/usr/local/lib
-- create-react-app@2.1.8
swecha@guptam-lenovo-g50-80:~$
```

create-react-app is a package that we can find in **NPM** and that will make our task easier. It takes care of every configuration. We only need to use it. Now that we have create-react-app installed,

Step:7 let's create our first app by typing:

create-react-app hello-world

```
swеча@guptan-lenovo-g50-80:~$ create-react-app hello-worl
Creating a new React app in /home/swеча/hello-worl.
Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts...

loadDep:yargs -> get      \ |#####-----
hello-worl@0.1.0 /home/swеча/hello-worl
+-- react@16.8.6
| +-- loose-envify@1.4.0
| | -- js-tokens@4.0.0
| +-- object-assign@4.1.1
+--

Success! Created hello-worl at /home/swеча/hello-worl
Inside that directory, you can run several commands:

  npm start
    Starts the development server.

  npm run build
    Bundles the app into static files for production.

  npm test
    Starts the test runner.

  npm run eject
    Removes this tool and copies build dependencies, configuration files
    and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd hello-worl
  npm start

Happy hacking!
swеча@guptan-lenovo-g50-80:~$
```

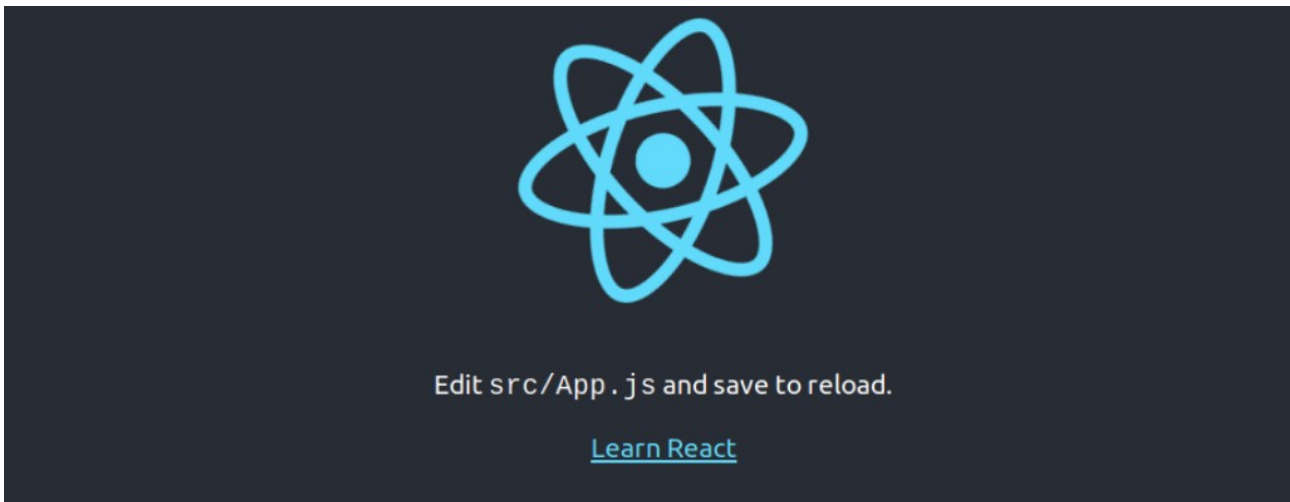
Step:8 After it finishes creating and setting up the application, we should move into the directory that was just created and that is named like our application by typing **cd hello-worl**.

I have created hello-world, yours can be anything you want. After a few seconds, a message will appear saying that our application was created successfully, and gives us a few instructions.

Step:9

npm start

This command will start a local server and will open our application in the browser, just like this:



Now we are moving to install the app and going to work on the app . so here we already build an app named as water cycle.so to access the app and to work on follow the steps as mentoned below:

Step:1

login to your code.swecha.org

Step:2

Goto **water-cycle** project

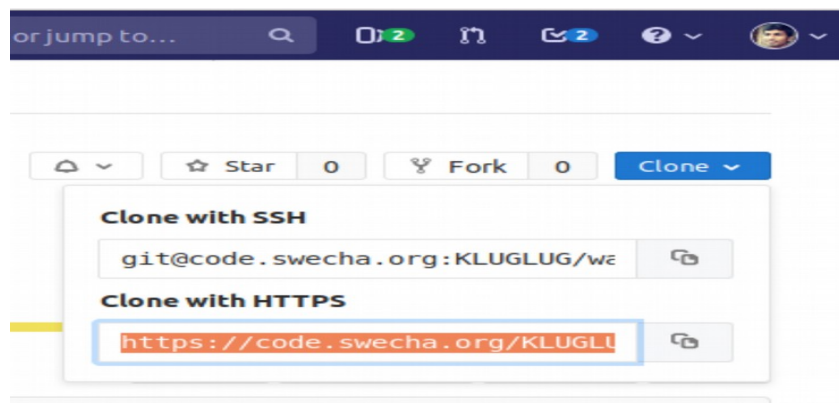
Step:3

Git Clone:-

git clone s a **Git** command line utility which is used to target an existing repository and create a **clone**, or copy of the target repository.

Cloning a local or remote repository.**Cloning** a bare repository.

Here clone the project (**clone with HTTPS**).



Step:4 After copying with (clone with https) and paste the url in the terminal

git clone "<https://code.swecha.org/KLUGLUG/water-cycle.git>"

```
swecha@guptam-lenovo-g50-80:~$ cd Videos/
swecha@guptam-lenovo-g50-80:~/Videos$ git clone https://code.swecha.org/KLUGLUG/water-cycle.git
Cloning into 'water-cycle'...
remote: Enumerating objects: 271, done.
remote: Counting objects: 100% (271/271), done.
remote: Compressing objects: 100% (171/171), done.
remote: Total 271 (delta 137), reused 214 (delta 97)
Receiving objects: 100% (271/271), 38.71 MiB | 41.00 KiB/s, done.
Resolving deltas: 100% (137/137), done.
swecha@guptam-lenovo-g50-80:~/Videos$
```

Step:5 After cloning the project .

1.Go to the working floder watercycle (cd watercycle)

2.Sudo npm install (it is used for installing the whole working dependencies which are used in watercycle)

```
swecha@guptam-lenovo-g50-80:~/Videos$ cd water-cycle/
swecha@guptam-lenovo-g50-80:~/Videos/water-cycle$ sudo npm install
npm WARN deprecated circular-json@0.3.3: CircularJSON is in maintenance only, flatled is its successor.
npm WARN deprecated kleur@2.0.2: Please upgrade to kleur@3 or migrate to 'ansi-colors' if you prefer the old syntax. Vis
/lukeed/kleur/releases/tag/v3.0.0 for migration path(s).
npm WARN deprecated core-js@2.6.4: core-js@<2.6.5 is no longer maintained. Please, upgrade to core-js@3 or at least to a
js@2.
loadDep:yargs -> addNameR \ |#####-----
npm WARN deprecated joi@11.4.0: This version has been deprecated in accordance with the hapi support policy (hapi.im/sup
to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this ti
available for older versions (hapi.im/commercial).
npm WARN deprecated hoek@4.2.1: This version has been deprecated in accordance with the hapi support policy (hapi.im/sup
to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this ti
available for older versions (hapi.im/commercial).
npm WARN deprecated topo@2.0.2: This version has been deprecated in accordance with the hapi support policy (hapi.im/sup
to the latest version to get the best features, bug fixes, and security patches. If you are unable to upgrade at this ti
+-- @semantic-ui-react/event-stack@3.1.0
|   -- exenv@1.2.2
+-- classnames@2.2.6
+-- keyboard-key@1.0.4
+-- lodash@4.17.11
+-- react-is@16.8.6
+-- shallowequal@1.1.0
npm WARN optional Skipping failed optional dependency /react-scripts/fsevents:
npm WARN notsup Not compatible with your operating system or architecture: fsevents@1.2.4
npm WARN optional Skipping failed optional dependency /chokidar/fsevents:
npm WARN notsup Not compatible with your operating system or architecture: fsevents@1.2.8
npm WARN optional Skipping failed optional dependency /sane/fsevents:
npm WARN notsup Not compatible with your operating system or architecture: fsevents@1.2.8
npm WARN react-marquee@1.0.0 requires a peer of react@~0.14.0 || ~15.0.0 || ~16.0.0 but none was installed.
npm WARN react-marquee@1.0.0 requires a peer of react-dom@~0.14.0 || ~15.0.0 || ~16.0.0 but none was installed.
npm WARN ts-pnp@1.0.1 requires a peer of typescript@* but none was installed.
swecha@guptam-lenovo-g50-80:~/Videos/water-cycle$
```

Step:6

After completion of installation, To run the app type the following command

sudo npm start

```
swecha@guptam-lenovo-g50-80:~/Videos/water-cycle$ sudo npm start
[sudo] password for swecha:

> e-litmus@0.1.0 start /home/swecha/Videos/water-cycle
> react-scripts start
```

```
./src/components/Aiup.js
Line 2: 'SmoothImage' is defined but never used no-unused-vars
Line 11: Do not use <marquee> elements as they can create visual accessibility issues and are deprecated jsx-a11y/no-distracti
-elements
Line 11: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

./src/components/Cloud.js
Line 3: 'cloud5' is defined but never used no-unused-vars
Line 13: Do not use <marquee> elements as they can create visual accessibility issues and are deprecated jsx-a11y/no-distracti
-elements
Line 14: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text
Line 15: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

./src/components/Cloud5.js
Line 3: 'cloud5' is defined but never used no-unused-vars
Line 13: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text
Line 14: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

./src/App.js
Line 6: 'Cloud2' is defined but never used no-unused-vars
Line 7: 'Cloud3' is defined but never used no-unused-vars
Line 8: 'Cloud4' is defined but never used no-unused-vars
Line 9: 'Cloud5' is defined but never used no-unused-vars
Line 11: 'Mountain' is defined but never used no-unused-vars
Line 103: No duplicate props allowed react/jsx-no-duplicat
-props
Line 171: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text
Line 194: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text
Line 216: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text
Line 240: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

./src/components/Mountain.js
Line 8: img elements must have an alt prop, either with meaningful text, or an empty string for decorative images jsx-a11y/alt-text

Search for the keywords to learn more about each warning.
To ignore, add // eslint-disable-next-line to the line before.
```

