Cloning a Repository

Let's move our Git repository to our VM instance.

For the purpose of this lesson, we will move a simple React app to our VM and access it through the internet. Here is the repository on GitHub:

📁 public	Initial commit from Create React App	2 hours ago
📁 src	Initial commit from Create React App	2 hours ago
📄 .gitignore	Initial commit from Create React App	2 hours ago
📄 README.md	Initial commit from Create React App	2 hours ago
📄 package.json	Update package.json	8 minutes ago

In the `package.json` file, we edit this part,

```
"scripts:"{
  "start": "react-scripts start",
```

and add,

```
"scripts:"{
  "start": "PORT=80 react-scripts start",
```

We are specifying the port to 80 because React usually runs on 3000, whereas normal websites run on 80. Therefore, for simplicity's sake, we'll keep port uniform.

The cloning URL for this app is

`https://github.com/rauhaanrizvi/simpleReactApp.git`

Go into the VM instance SSH console:

	Name ∧	Zone	Recommendation	Internal IP	External IP	Connect	
☐	✅ instance-1	us-east1-b		10.142.0.2 (nic0)	35.196.64.74	SSH ▾	⋮

Gain root access using `sudo su` and install **NodeJS** and **npm** in the environment.

Let's clone the repository into the root directory using

```
git clone https://github.com/rauhaanrizvi/simpleReactApp.git
```

Use `ls` to check if the repository has been copied. The console should display this:

```
root@instance-1:/home/rauhan_ahmed# git clone https://github.com/rauhaanrizvi/simpleReactApp.git
Cloning into 'simpleReactApp'...
remote: Enumerating objects: 18, done.
remote: Counting objects: 100% (18/18), done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 18 (delta 0), reused 18 (delta 0), pack-reused 0
Unpacking objects: 100% (18/18), done.
root@instance-1:/home/rauhan_ahmed# ls
simpleReactApp
root@instance-1:/home/rauhan_ahmed#
```

Move into the `simpleReactApp` folder and run `npm install`.

We can run the application using `npm start`, but how would users access it through the internet? We'll solve this problem in the following lessons.