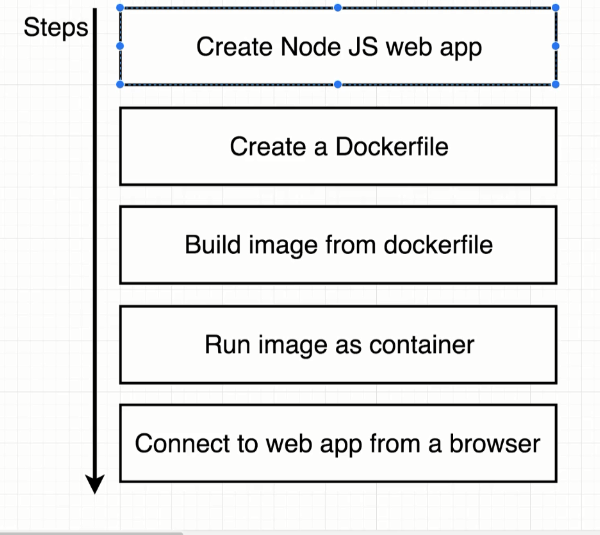
**V--39:(Project Outline):**

The goal of this project is to create a tiny nodejs web application wrap it inside of a docker contianer and then be able to access that web application from a browser running on your local machine

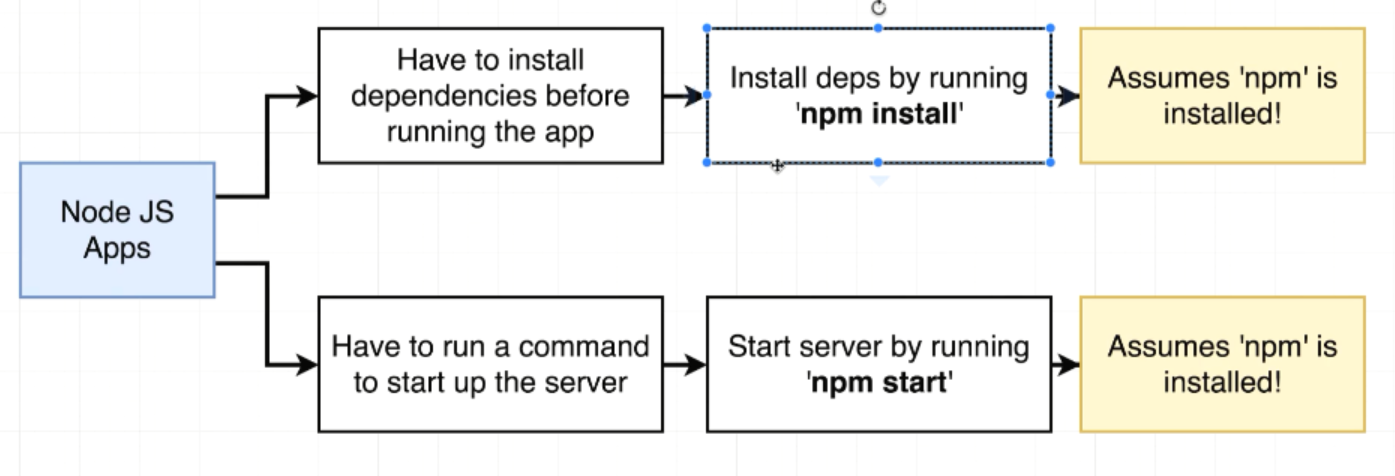


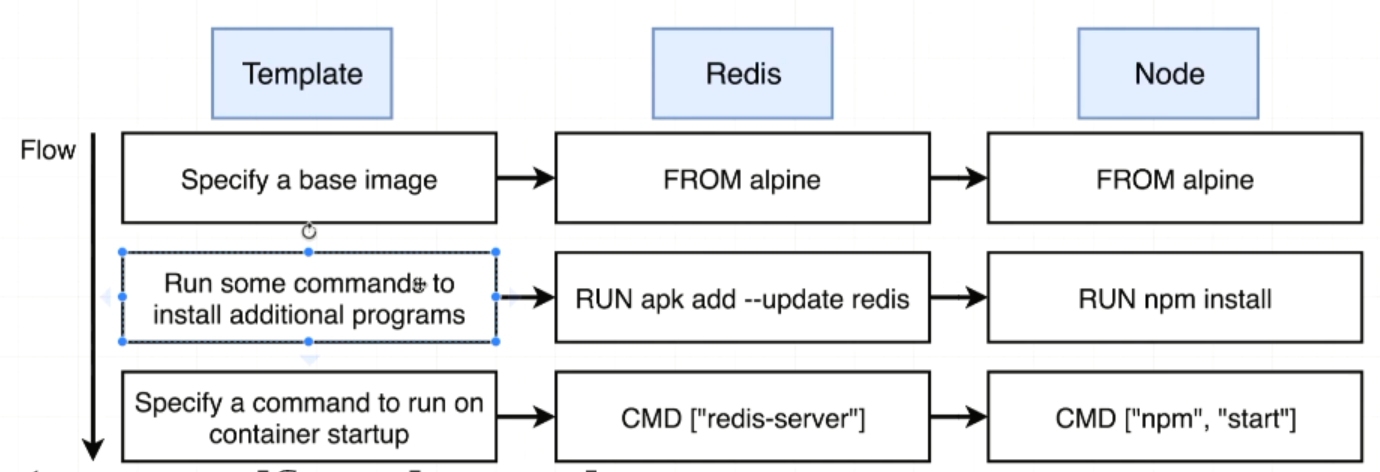
V—40:(NodeServer Setup):



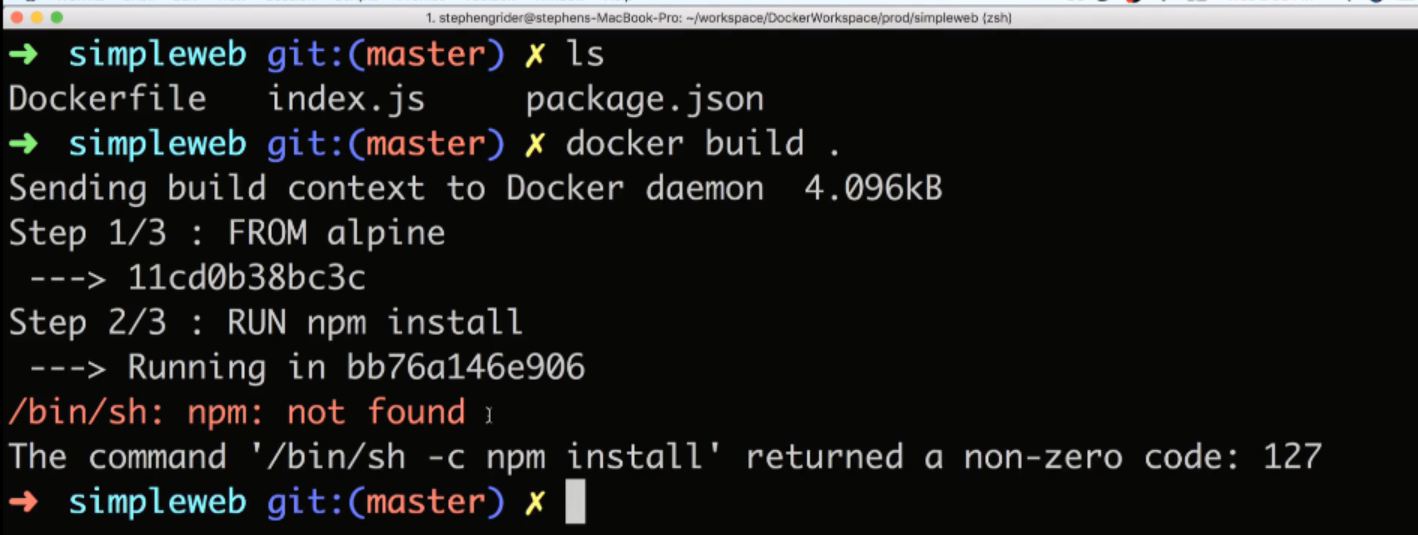


V—41:(Few Planned Errors):

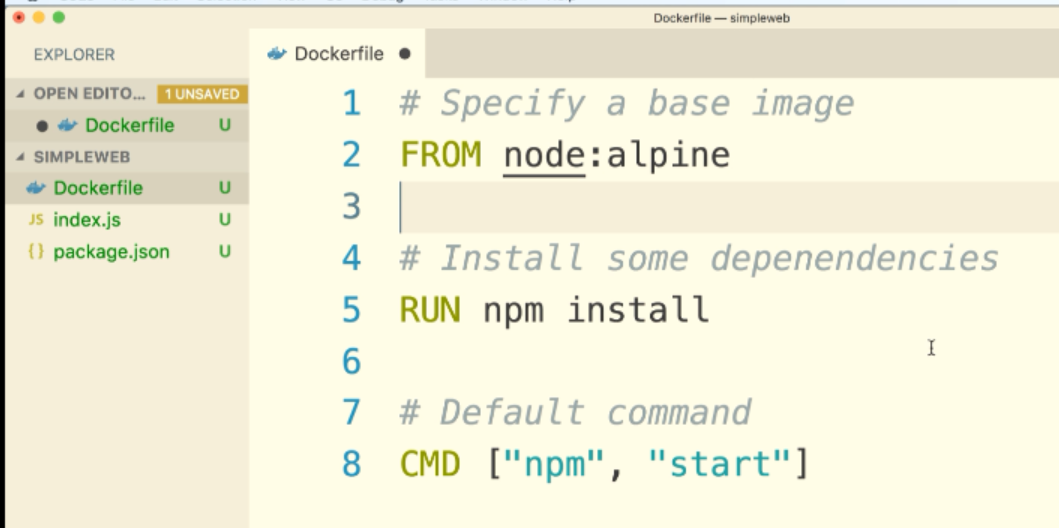


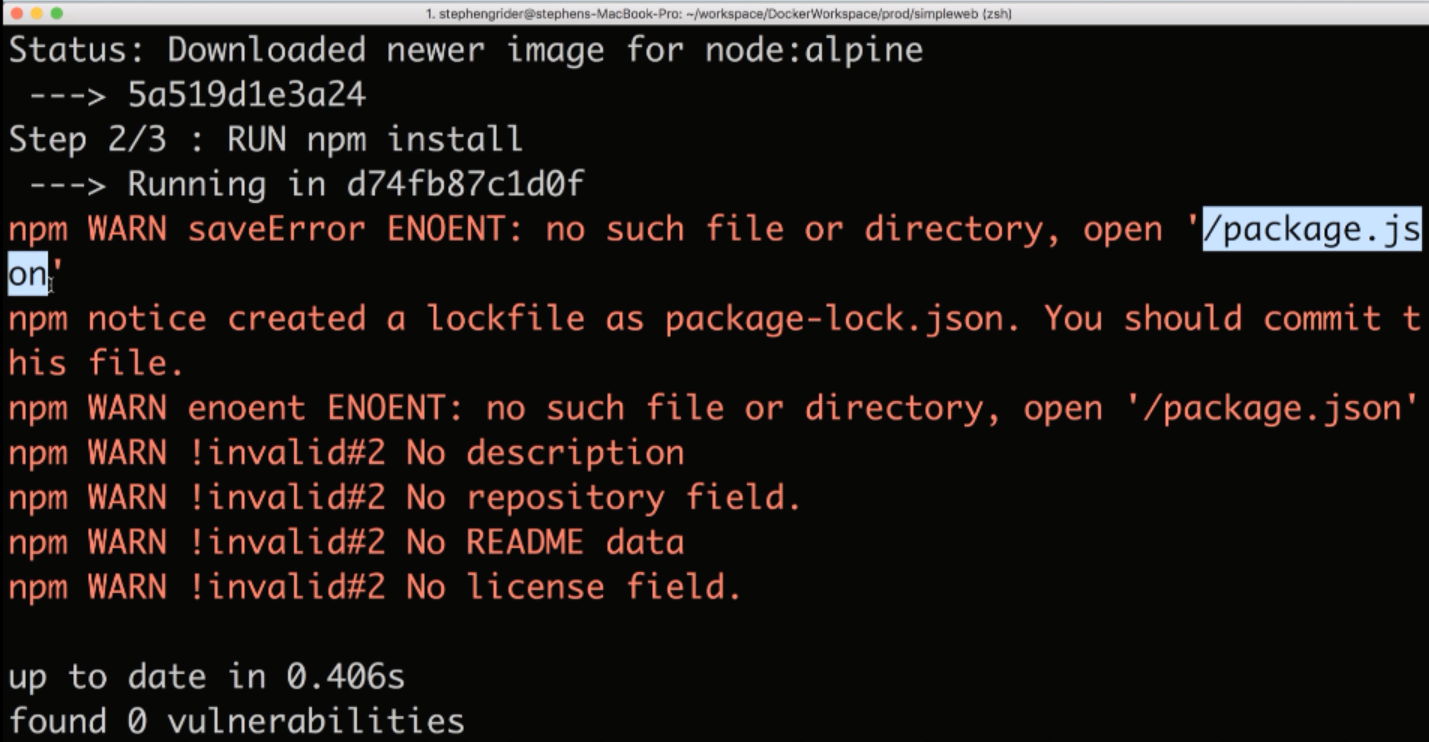






V—43:(Base Image Issue):

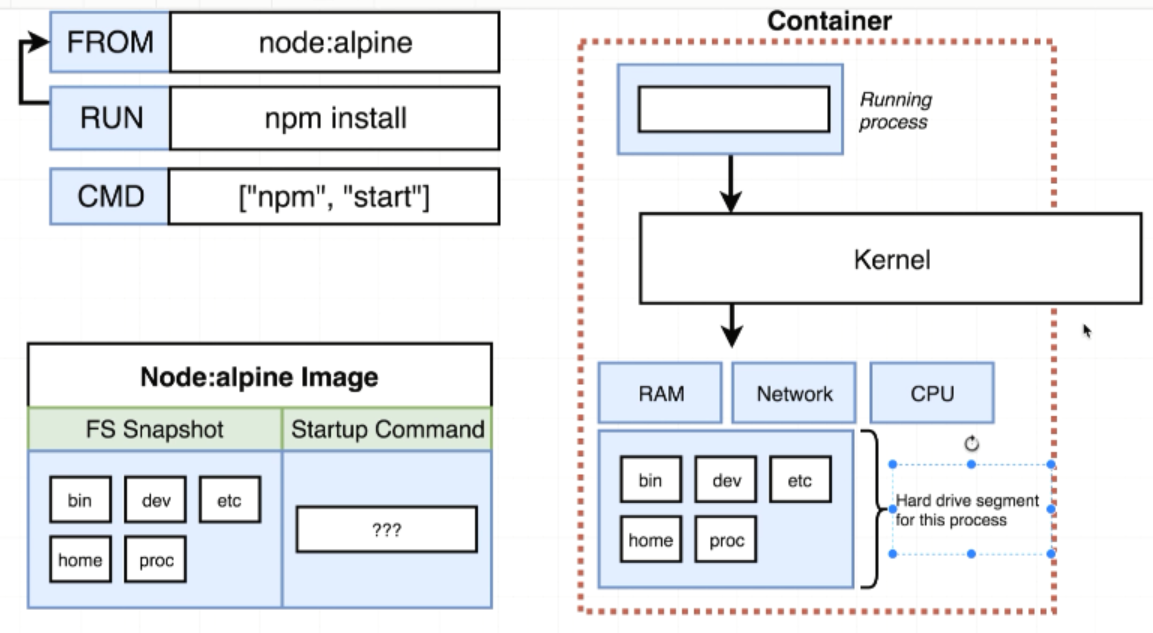


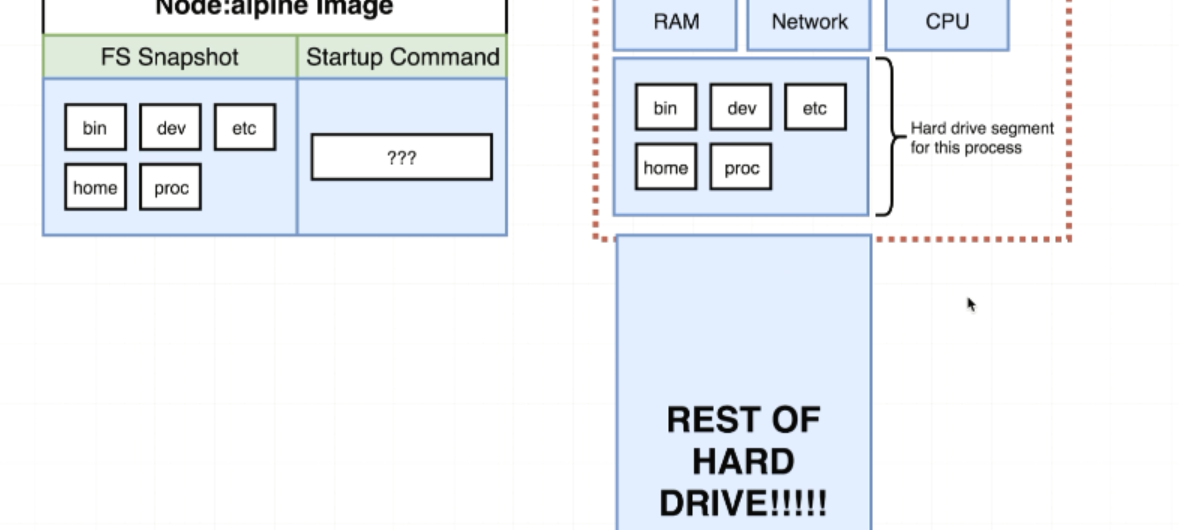


V—44:(A Few Missing Files):

Files package.json and index.js files are missing in the container.

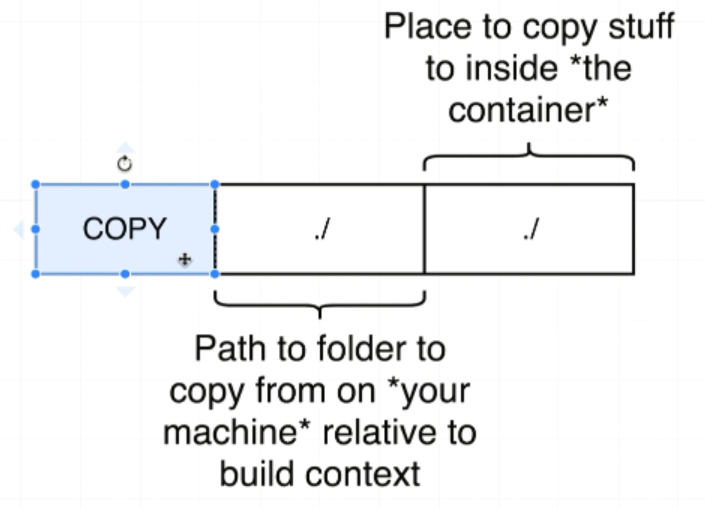
They are outside of the container. Outside of the container means rest of the hard drive.





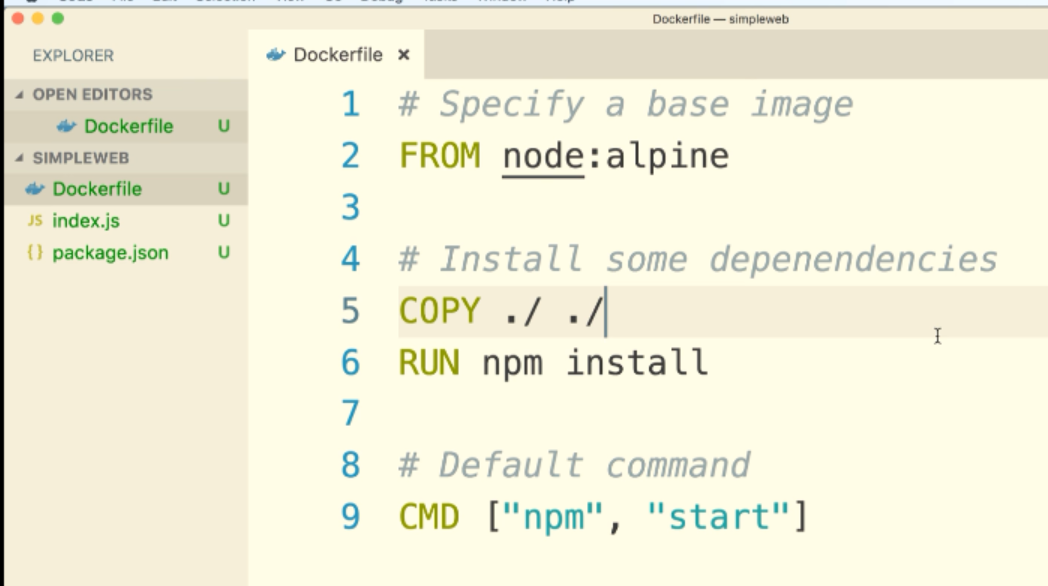
V—45:(Copying Build File):

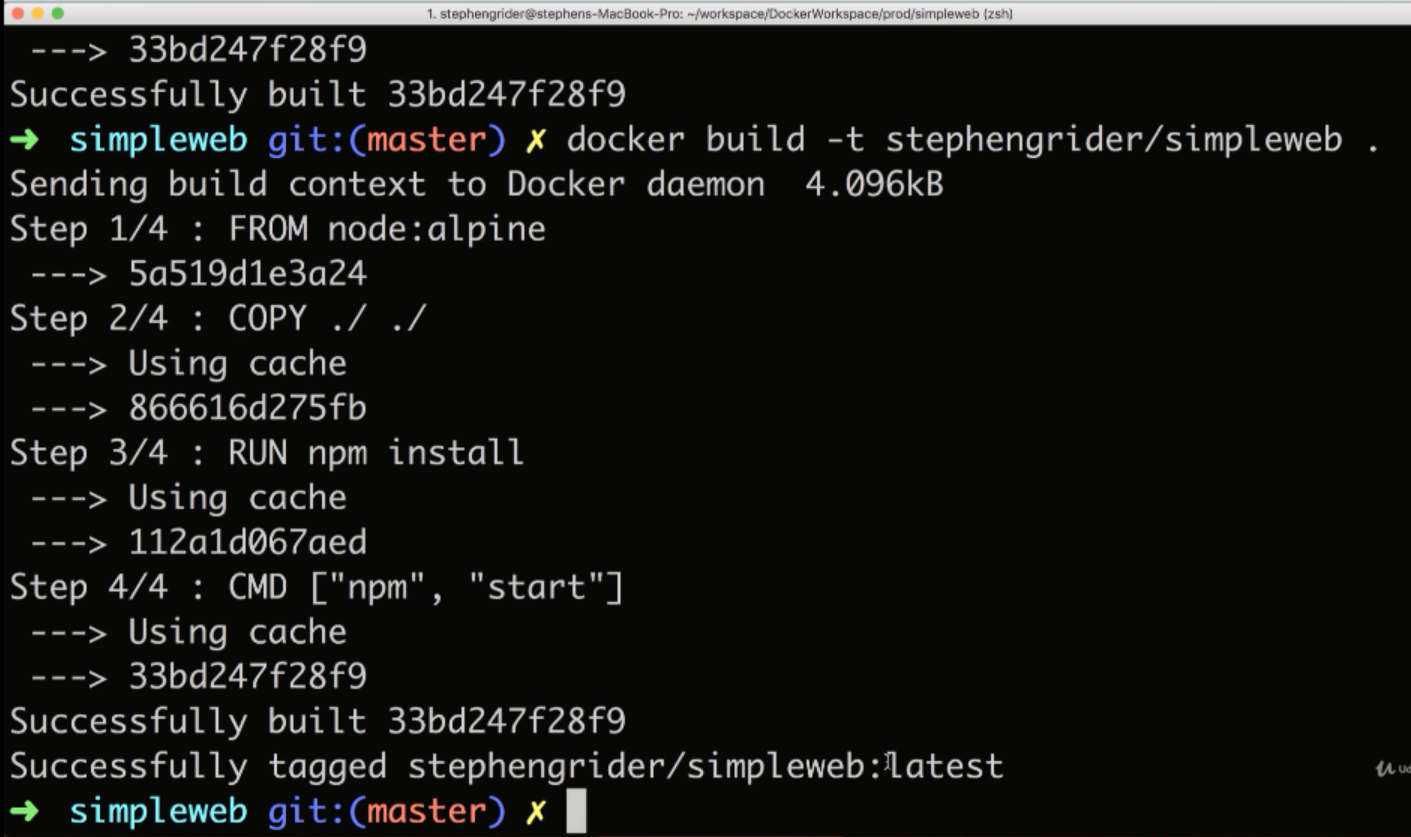
How we can make sure that package.json file is available before NPM install is executed inside the container.



package.json file should be available before npm install command executes.

COPY ./ ./ 🡪 MEANS Everything from the current working directory of simpleweb to the current working directory inside the container.







V—47:(Container Port Mapping):

In order to make sure that any request from either your computer or some outside computer will be redirected into the container. We have to set up a explicit port mapping

Port mapping essentially says anytime that someone makes a request to a given port on your local network take that request in automatically and forward it to some port inside the container.

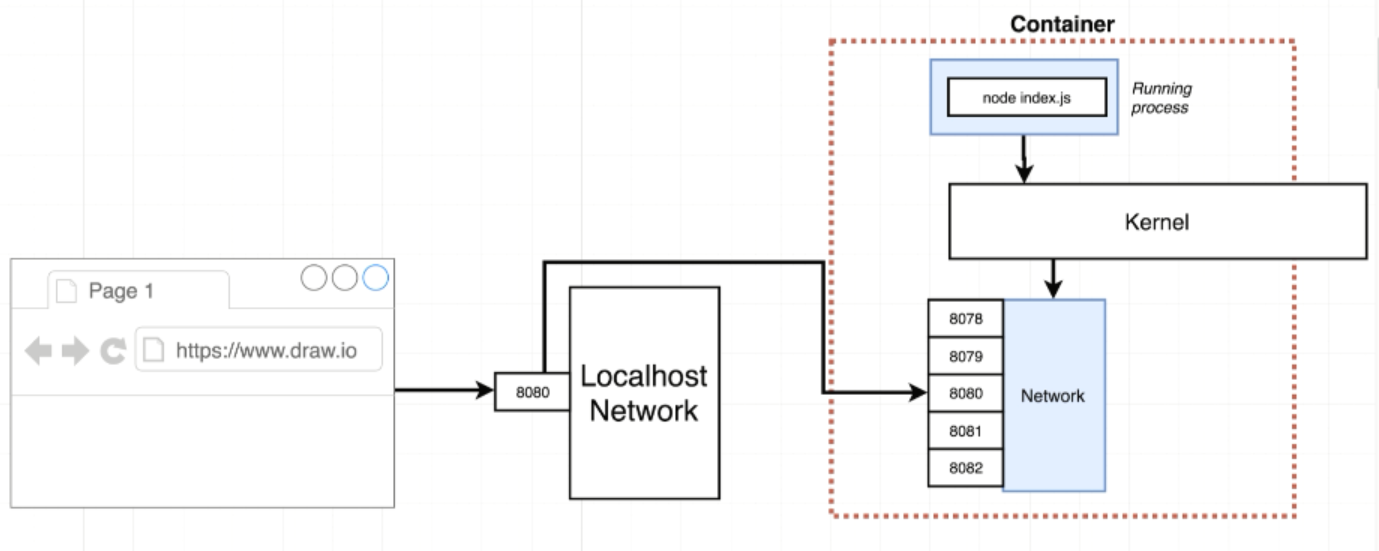
If anyone makes a request to localhost port 8080, take that request automatically forward into the container on port 8080 where the node application can then receive it and process the request and eventually respond to it.

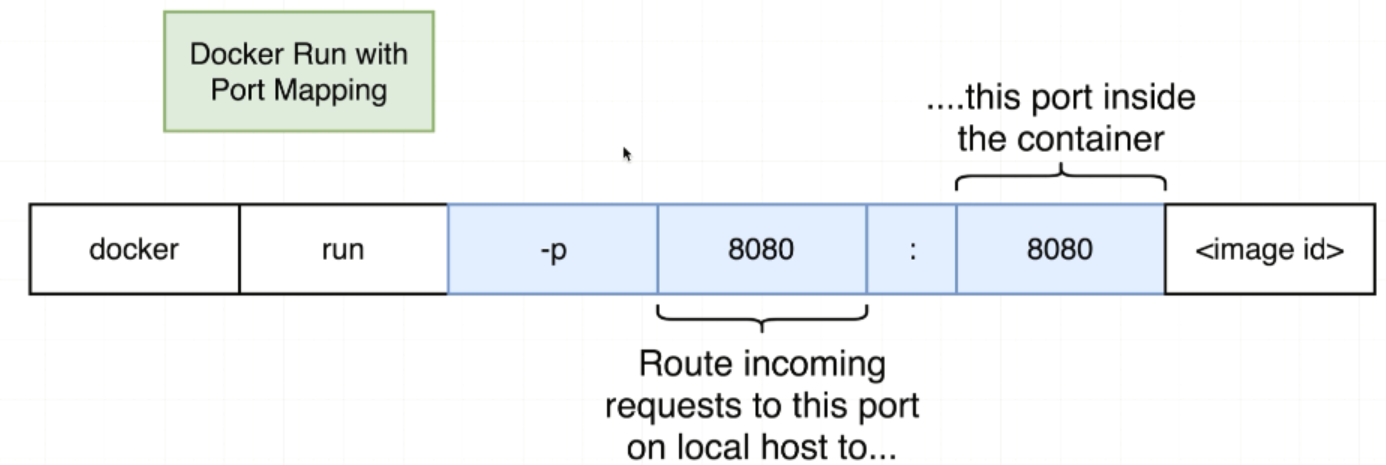
This is only talking about incoming request.

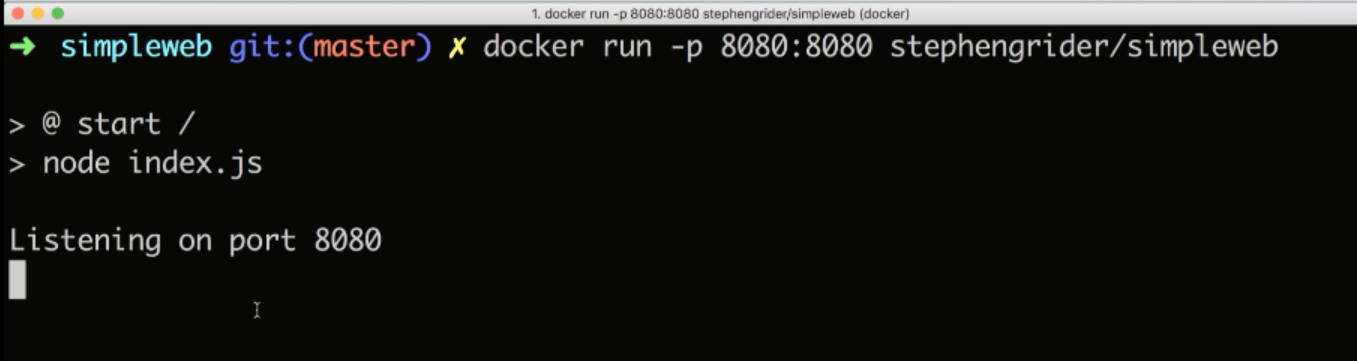
Your docker container can by default make requests on its own behalf to the outside world.

When we ran npm install during the docker build process, npm reaches out to the outside world. It reaches out across the internet and so there is no limitations by default on your containers ability to reach out.

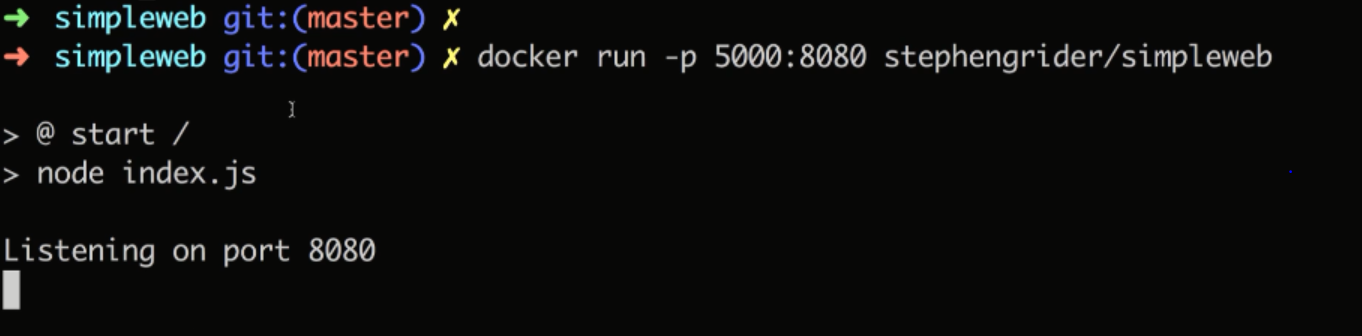
It’s strictly limitation on the ability for incoming traffic to get into the container.





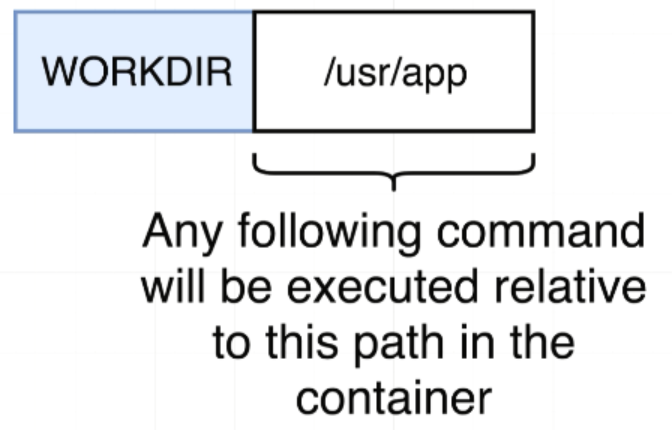


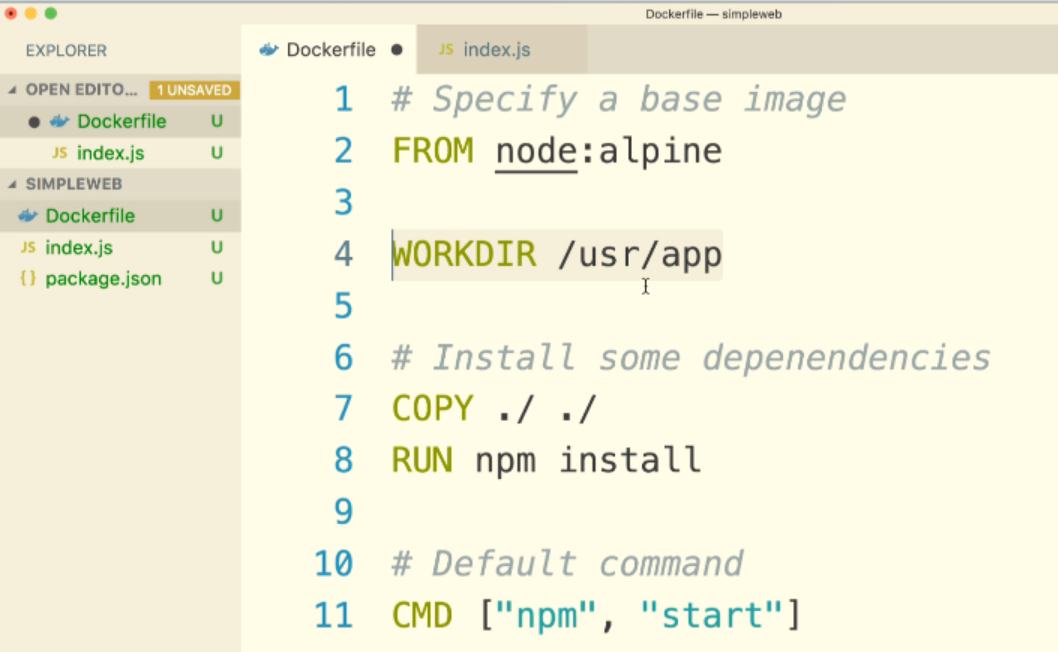




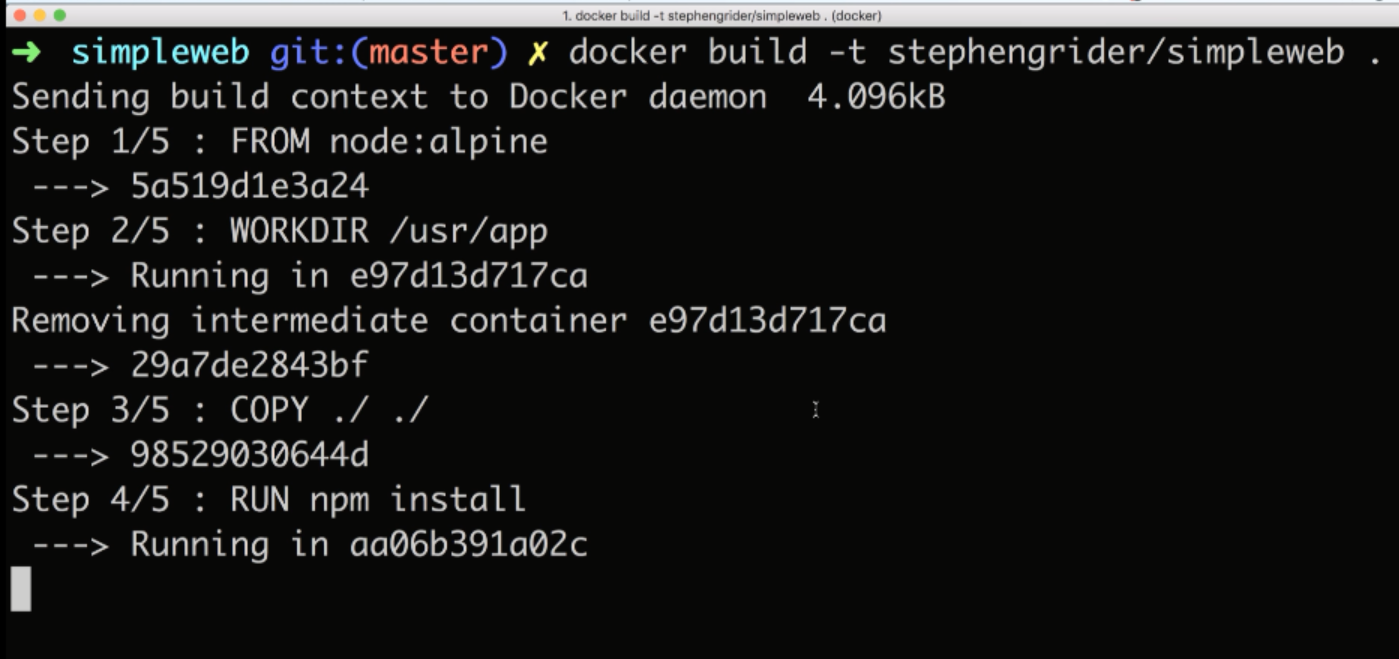


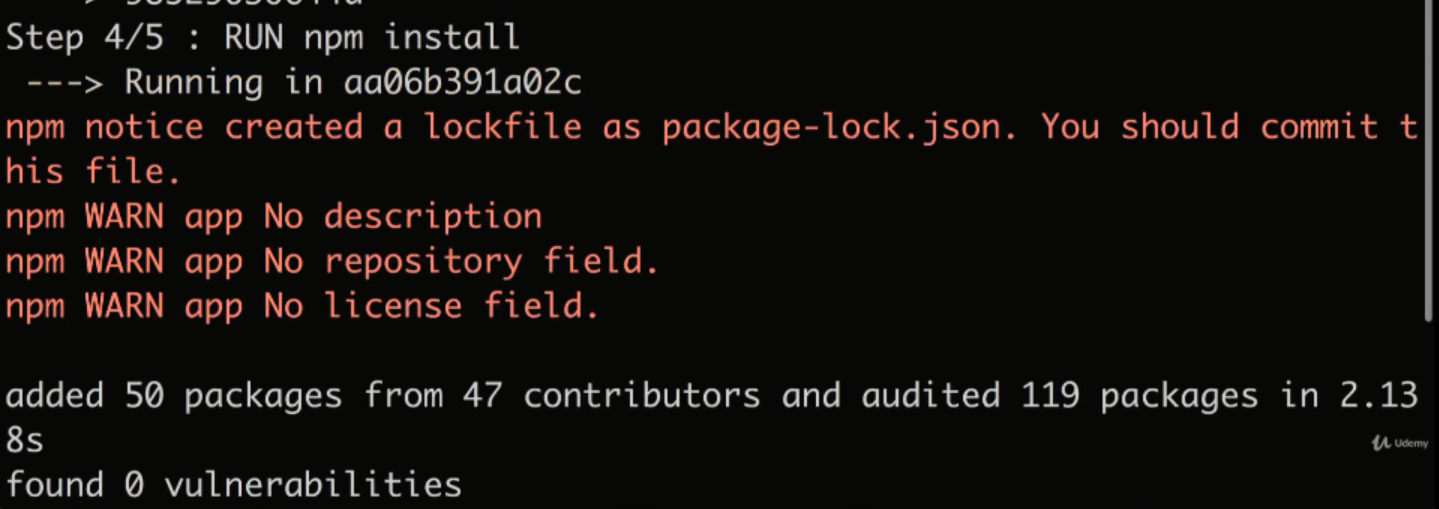
V—48:(Specifying a Working Directory):

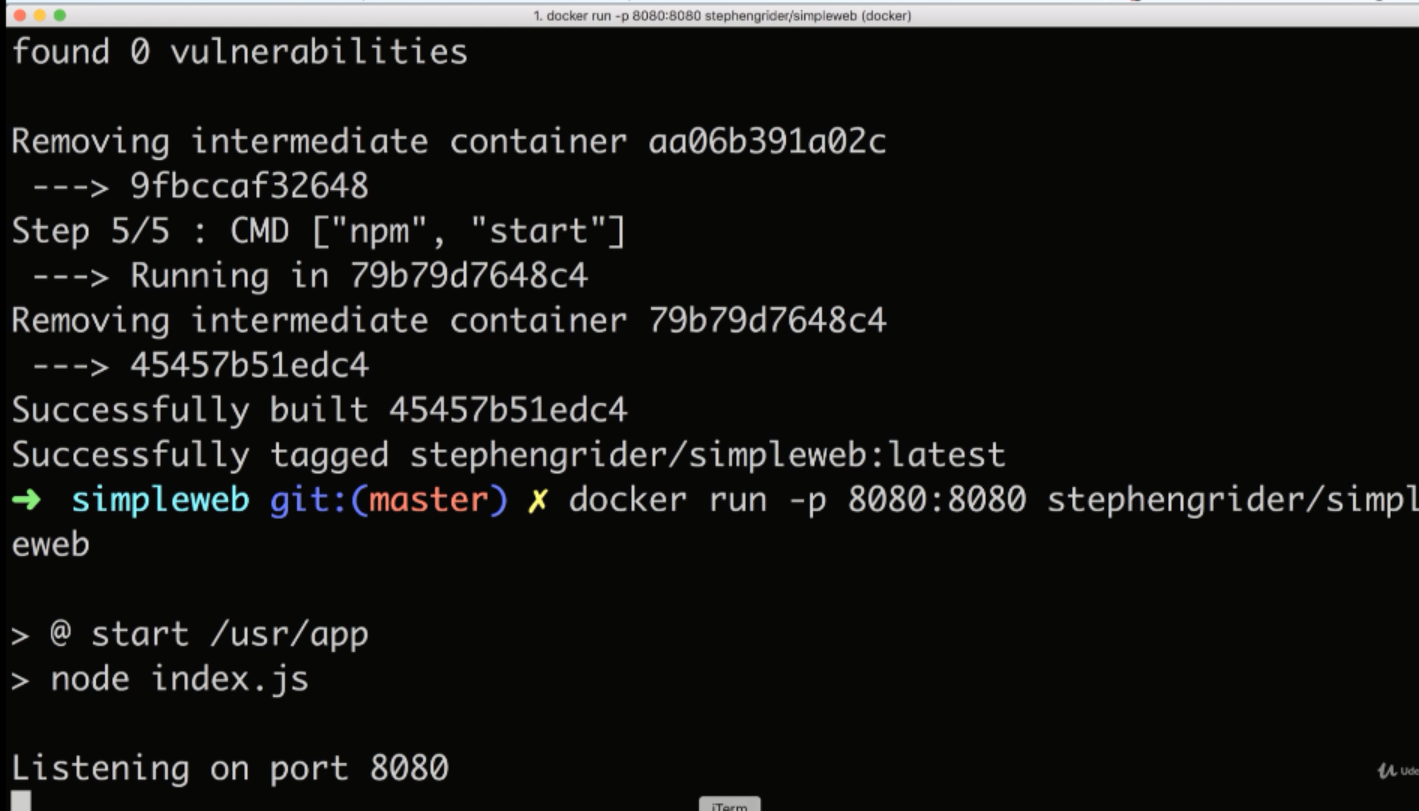




First terminal window:

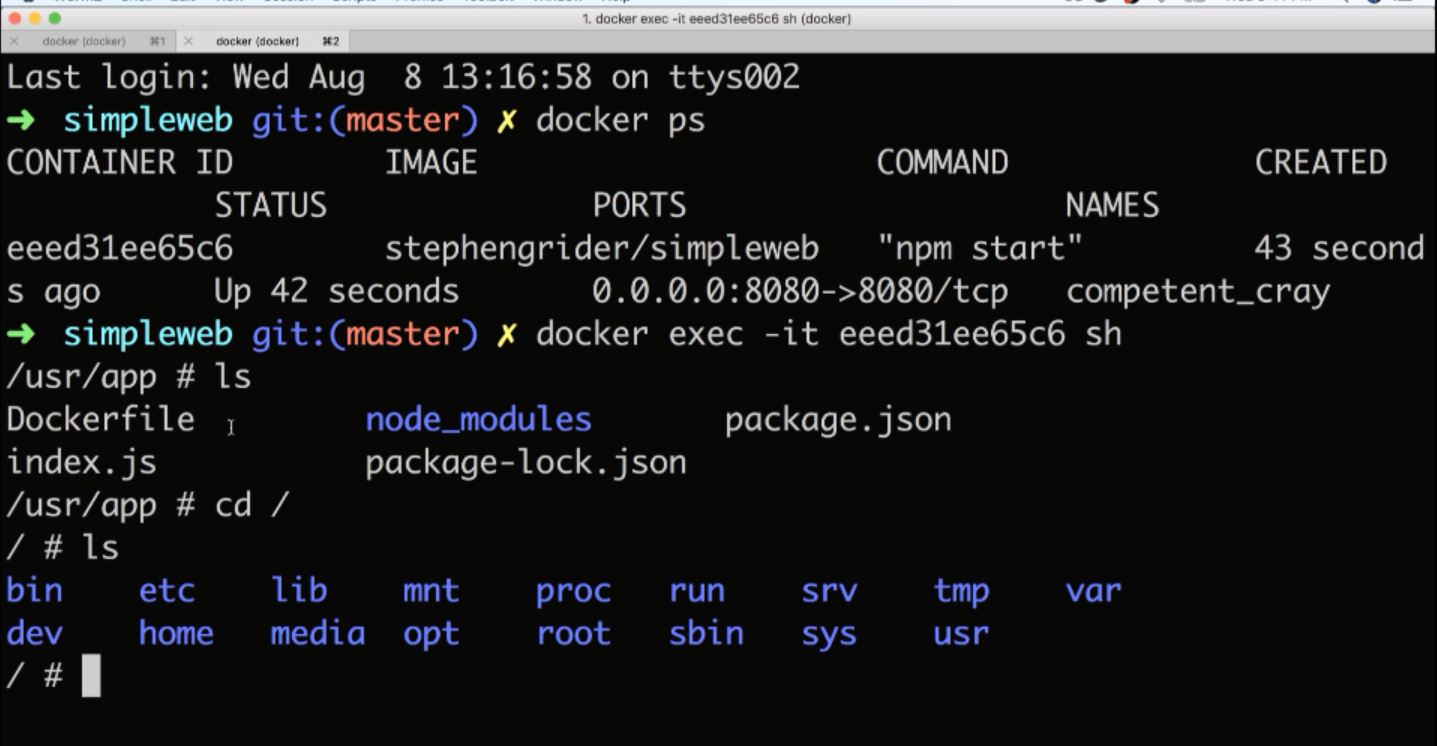






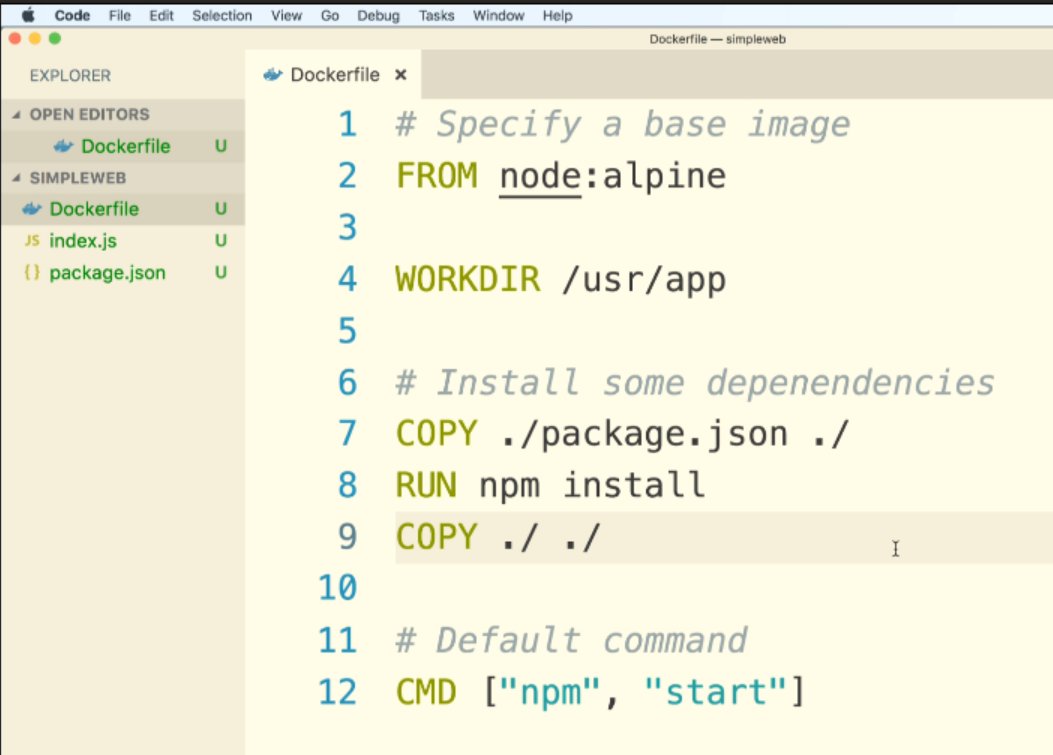
Second terminal window:

To open shell inside of the container:



V—49:(Unnecessary Rebuilds):

V—50:(Minimizinf cahce busty and rebuilds):



Now any change to index.js file will not re install all dependencies. It means It will not run line no :8 again.