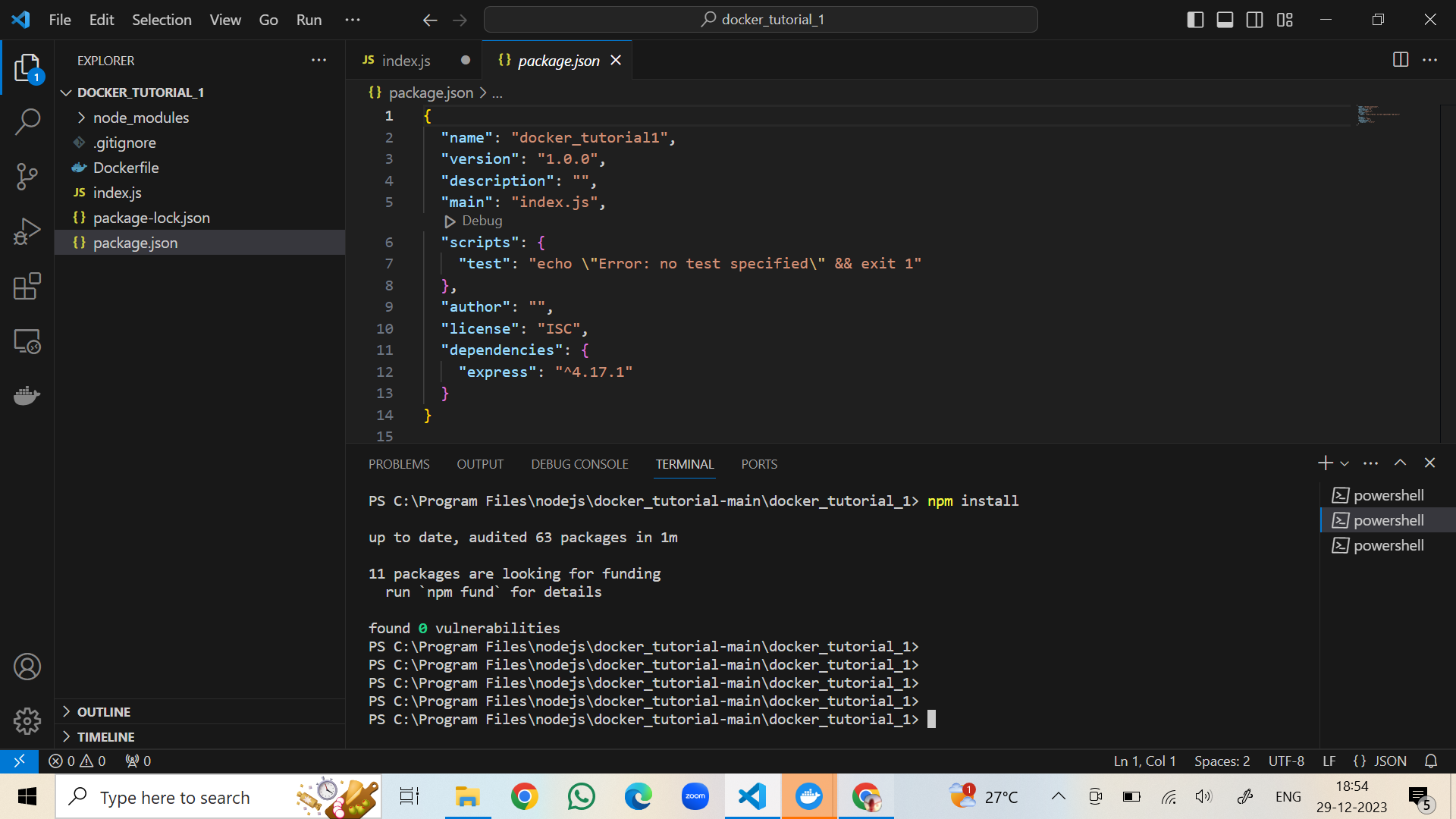
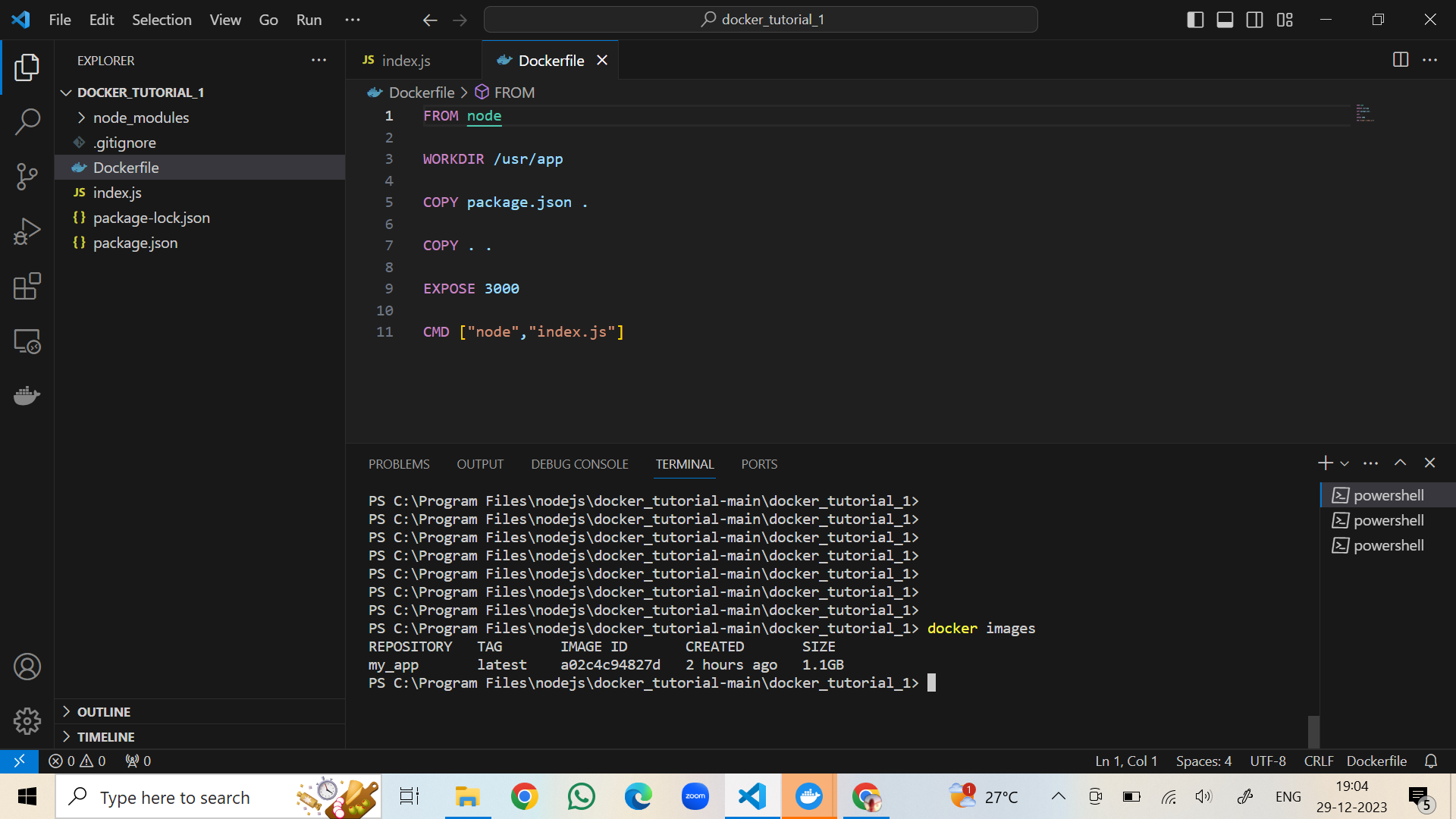
1.Docker Challenge:  
Task: Create a Dockerfile for a simple Node.js web application.

* Our apps run inside a container, container is a running instance of an image
* To build image we need dockerfile.
* First we have to create dockerfile file for nodejs app
* We can take linux os(alpine linux)
* On topof tat we can install nodejs
* I copied pacakage.json inside image
* Then npm install on dockerfile and node modules
* Then copied source code form host machine to the docker image
* Instead of using nodejs and alpine we can use nodejs image from docker repo
* I took nodejs image from docker hub and downloaded on local machine and opened in vs code
* To run application I run npm install (it will install all dependency for application)
* 
* Then I run node index.js
* We can see our app is running on port 3000 (<http://localhost:3000>)
* App is running
* I installed docker desktop
* Then I created dockerfile
* Dockerfile contains all the instruction to build an image
* Dockerfile
* 

FROM node(it always start with from then base image(node))

WORKDIR /usr/app (work directory-tat is going to contain all our code)

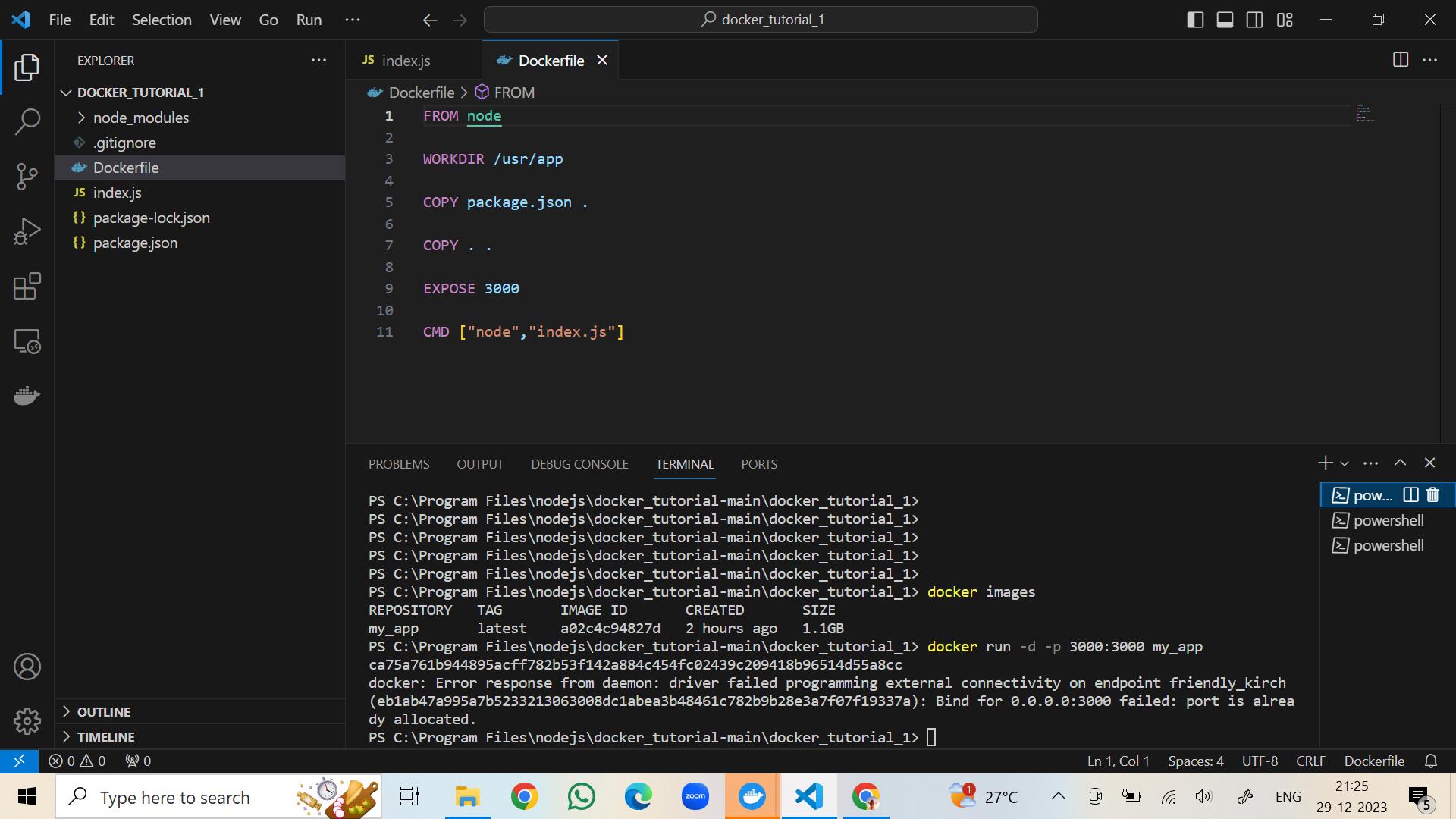
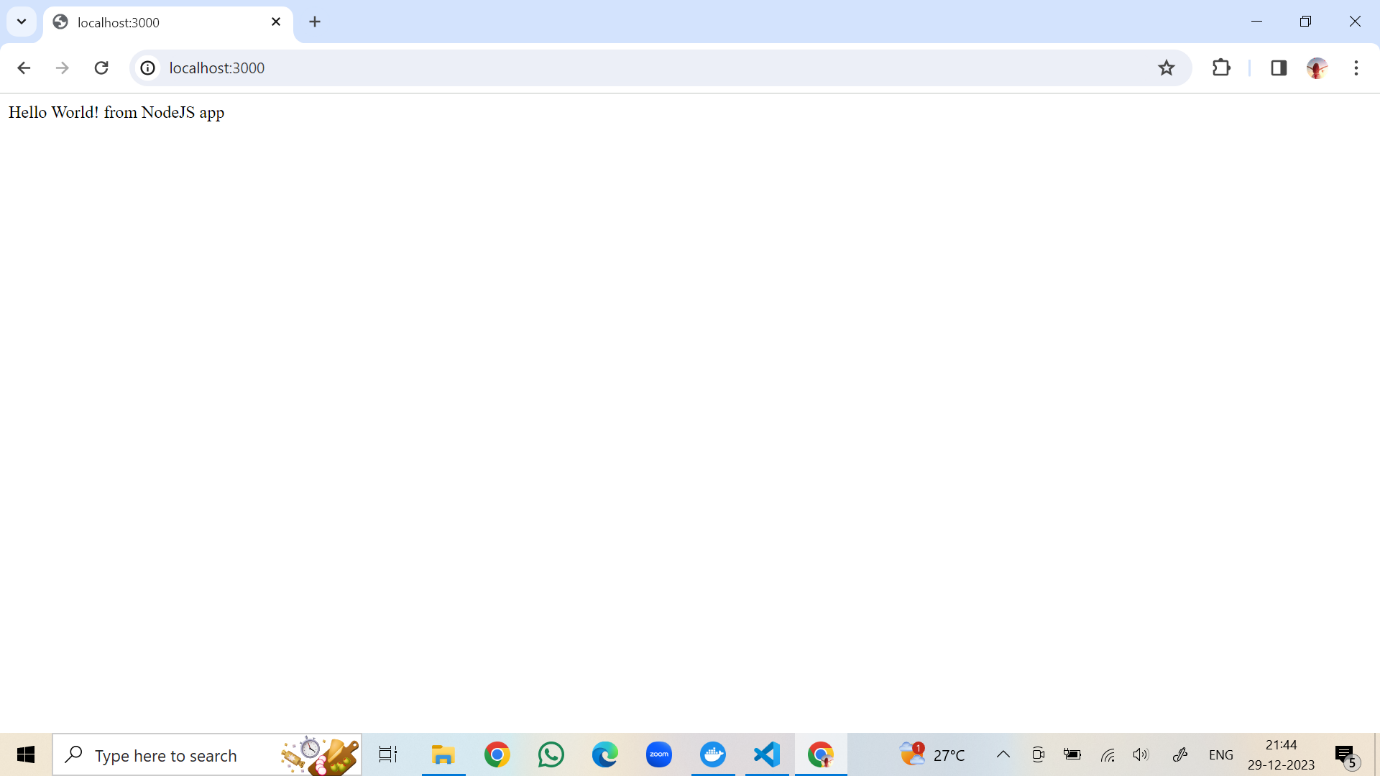
COPY package.json . (copy package.json from host machine inside our work directory)

RUN npm install (npm install to run inside our image )

COPY . . (Copying entire source code from our host machine inside our container )

EXPOSE 3000 (port 3000)

CMD[“node”,”index.js”] (cmd command defines runtime, main file is index.js)

* In terminal – going ti build image from dockerfile
* Docker build –t my\_app .
* Docker image created
* To check docker images
* 
* Now I run a container from image--- docker run –d –p 3000:3000 my\_app
* 
* 