

PRACTICAL EXERCISE - DOCKER

TASK:-1

#Image Should have proper name and tag

—> sudo docker build -t node-app:latest .

TASK:-2

#Application should be running on port 5000

—> sudo docker run -d -p 5000:3000 --name node-app node-app:latest

TASK:-3

#Logs folders should be mounted on the local disk - that means if someone restarts the container logs folder's data should be persisted.

—> sudo docker run -v nodeapp-volume:/uv/nodeapp -d --name node-app node-app:latest

TASK:-4

#Application should run in “Production Mode” not in “Local Mode”.

—> sudo docker run -e ENV_NODE=production

TASK:-5

#Dockerfile

```
# Latest version of node as bash image
FROM node:latest

# Expose the port that is used in server file
EXPOSE 3000

# The working directory is used to creat the directory and it will creat
location inside container as working directory.
WORKDIR /uv/nodeapp

# We want to run our server file in production mode
ENV ENV_NODE = production

# It will copy the .json file from local directory to container directory
COPY package.json package-lock.json* ./

# To run server file we need to install npm inside container directory
```

```
RUN npm install --production

# It will copy all the from local directory to container directory
COPY . .

# Start node with its default content
ENTRYPOINT [ "node" ]

# It will run the server file
CMD ["server.js" ]
```

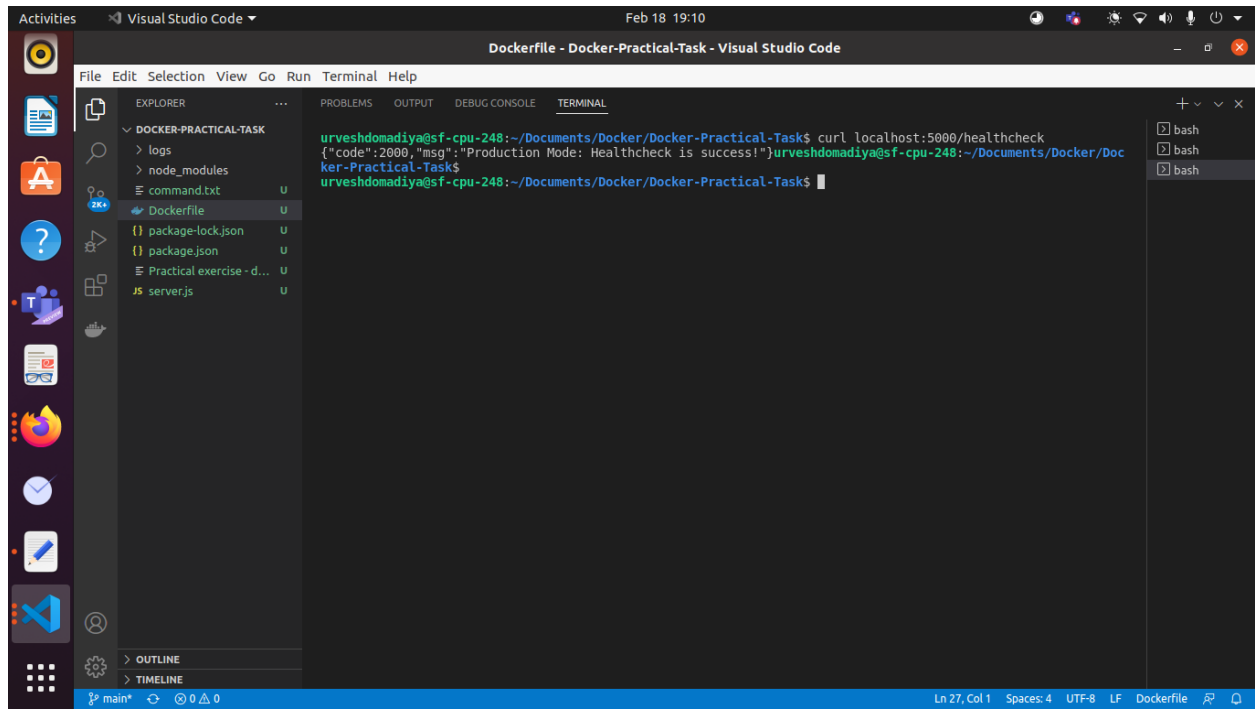
#Creat a docker image and Run it in your local environment.

—> sudo docker run -p 5000:3000 -v nodeapp-volume:/uv/nodeapp -e ENV_NODE=production
-d --name node-app node-app:latest

#Execution command for local environment

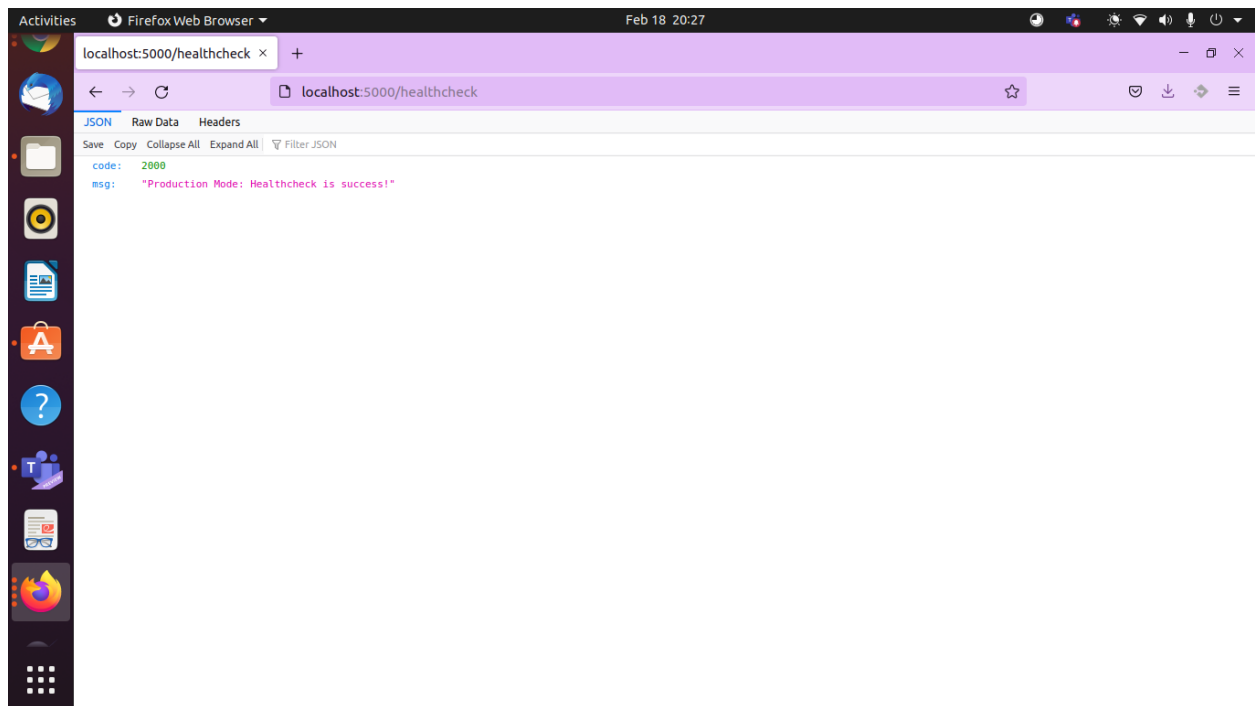
—> curl localhost:5000/healthcheck

Output:-



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left displays the file structure of a project named 'DOCKER-PRACTICAL-TASK'. The files listed are 'logs', 'node_modules', 'command.txt', 'Dockerfile', 'package-lock.json', 'package.json', 'Practical exercise - d...', and 'server.js'. The Dockerfile is selected. The terminal window at the bottom shows the following commands and output:

```
urveshdomadiya@sf-cpu-248:~/Documents/Docker/Docker-Practical-Task$ curl localhost:5000/healthcheck
{"code":2000,"msg":"Production Mode: Healthcheck is success!"}urveshdomadiya@sf-cpu-248:~/Documents/Docker/Doc
ker-Practical-Task$
urveshdomadiya@sf-cpu-248:~/Documents/Docker/Docker-Practical-Task$
```



The screenshot shows the Firefox Web Browser interface. The address bar displays 'localhost:5000/healthcheck'. The page content shows the following JSON response:

```
{
  "code": 2000,
  "msg": "Production Mode: Healthcheck is success!"
}
```