**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/logs -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/logs -e ENV\_NODE=production -d --name node-app node-app:latest

#Execution command for local environment

—> curl localhost:5000/healthcheck

**Output:-**



