

2. Develop a Multi-stage Dockerfile for container Orchestration

Directory Structure:

```
P2
|--- Dockerfile
|--- package.json
|--- / src/ index.js
|--- / dist / index.js
|--- node_modules
|--- / build
```

Step-1 : Create a Dockerfile using nano in the pwd

→ Dockerfile

```
// Stage-1
FROM node:20-alpine AS builder
WORKDIR /app
COPY package.json package-lock.json ./
RUN npm install
COPY . .
RUN npm run build

// Stage-2
FROM node:20-alpine
WORKDIR /app

COPY --from=builder /app/package.json ./
COPY --from=builder /app/package-lock.json ./
COPY --from=builder /app/dist ./dist
COPY --from=builder /app/node_modules ./node_modules

EXPOSE 5000
CMD ["node", "dist/index.js"]
```

Step-2: Create a express script inside src folder

→ / **src / index.js**

```
const express= require('express');
const app = express();
const PORT = 5000;

app.get('/', (req, res)=>{
res.send('Hello from Multi-stage Docker!!');
});

app.listen(PORT, ()=>{
console.log('Server running on port ${PORT}');
});
```

Step-3: Create a custom package.json file or you can also run **npm init -y** it'll generate package.json for you and you can modify the content according to you.

→**package.json**

```
{
  "name": "p2",
  "version": "1.0.0",
  "description": "A Node js app with multi-staged Dockerfile",
  "main": "dist/index.js",
  "scripts": {
    "start": "node dist/index.js",
    "build": "mkdir -p dist && cp -r src/* dist/"
  },
  "keywords": [],
  "author": "Niranjan",
  "license": "No-Licence-yet",
  "dependencies":{
    "express": "^4.18.2"
  }
}
```

```

1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps$ mkdir program2
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps$ cd program2
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ mkdir src
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ cd src
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2/src$ nano index.js
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2/src$ cd ..
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ nano package.json
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ nano Dockerfile
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ docker build -t program2 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 5.12kB
Step 1/14 : FROM node:20-alpine AS builder
--> 2b56f2779663
Step 2/14 : WORKDIR /app
--> Using cache
--> 24c674d87b3f
Step 3/14 : COPY package.json package-lock.json .
COPY failed: file not found in build context or excluded by .dockerignore: stat package-lock.json: file does not exist
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ npm install

added 69 packages, and audited 70 packages in 3s

14 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

```

Step-4 : install Node modules using below command

`npm install`

Step-5 : Build a Docker image using

`sudo docker build program2 .`

```

--> Removed intermediate container 06a07dfa84f2
--> 2e200b4c3b76
Step 5/14 : COPY . .
--> ef862f7f516a
Step 6/14 : RUN npm run build
--> Running in 4c65eae688d0

> program2@1.0.0 build
> mkdir -p dist && cp -r src/* dist/

--> Removed intermediate container 4c65eae688d0
--> 90a0da7ee828
Step 7/14 : FROM node:20-alpine
--> 2b56f2779663
Step 8/14 : WORKDIR /app
--> Using cache
--> 24c674d87b3f
Step 9/14 : COPY --from=builder /app/package.json .
--> ef5fa963187d
Step 10/14 : COPY --from=builder /app/package-lock.json .
--> 3b03c2e4ced3
Step 11/14 : COPY --from=builder /app/dist ./dist
--> 2c1259d75fd4
Step 12/14 : COPY --from=builder /app/node_modules ./node_modules
--> 6e952db0fb34
Step 13/14 : EXPOSE 3000
--> Running in 8eb81193b7d0
--> Removed intermediate container 8eb81193b7d0
--> 5e7c32f22f9c
Step 14/14 : CMD ["node", "dist/index.js"]
--> Running in 3c7fdb8a1dea
--> Removed intermediate container 3c7fdb8a1dea
--> e5a587ea8e16
Successfully built e5a587ea8e16
Successfully tagged program2:latest
1RV24MC029_CHINMAYAKAGOLLI@CHINMAY:~/DevOps/program2$ docker run -p 3000:3000 program2
Server running on port http://localhost:8000

```

Step-6 : Run the image to build a container

```
sudo docker run -t -p 3000:3000 program2
```

```
1 found 0 vulnerabilities
1RV24MC029_CHINNAYAKAGOLLI@CHINMAY:~/DevOps/program2$ docker build -t program2 .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
Install the buildx component to build images with BuildKit:
https://docs.docker.com/go/buildx/

Sending build context to Docker daemon 2.846MB
Step 1/14 : FROM node:20-alpine AS builder
--> 2b56f2779663
Step 2/14 : WORKDIR /app
--> Using cache
--> 24c674d87b3f
Step 3/14 : COPY package.json package-lock.json .
--> 0d848ebfa4e4
Step 4/14 : RUN npm install
--> Running in 06a07dfa84f2
added 69 packages, and audited 70 packages in 1s

14 packages are looking for funding
  run `npm fund` for details

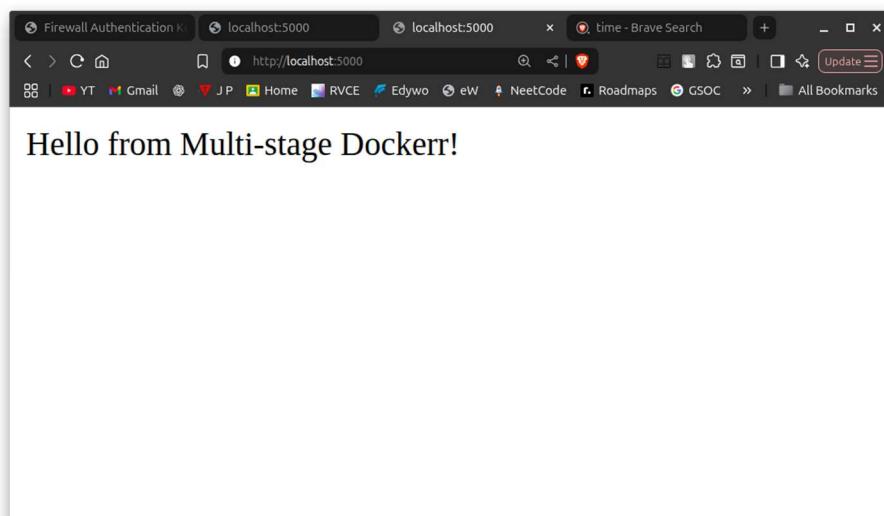
found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 10.8.2 => 11.6.2
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.6.2
npm notice To update run: npm install -g npm@11.6.2
npm notice
--> Removed intermediate container 06a07dfa84f2
--> 2e200b4c3b76
Step 5/14 : COPY .
--> ef862f7f516a
Step 6/14 : RUN npm run build
--> Running in 4c65eae688d0

> program2@1.0.0 build
> mkdir -p dist && cp -r src/* dist/
--> Removed intermediate container 4c65eae688d0
--> 90a0da7ee828
Step 7/14 : FROM node:20-alpine
--> 2b56f2779663
Step 8/14 : WORKDIR /app
--> Using cache
--> 24c674d87b3f
--> 0d848ebfa4e4
Step 9/14 : COPY . . . . .
```

Step-7 : check the status of the container using

Sudo docker ps

<http://localhost:3000> on any browser



Step-8 : Stop the container , remove the container and delete the image

```
sudo docker container ls -a   or   docker ps -a  
sudo docker container stop <container-id>  
sudo docker container rm <container-id>  
sudo docker image rm <image-id>   or   docker rmi <image-id>
```