

Srusti G C
1RV24MC106

Program 2: Develop a Multi-Stage Dockerfile for Container Orchestration.

Step 1: Create a project folder
`mkdir multi-stage-docker-demo`
`cd multi-stage-docker-demo`

Step 2: Create subfolder for the app
`mkdir app`
`cd app`

Step 3: Create your app files

a) Create package.json

`nano package.json`

Then paste this inside:

```
{  
  "name": "multi-stage-demo",  
  "version": "1.0.0",  
  "main": "server.js",  
  "scripts": {  
    "start": "node server.js"  
  },  
  "dependencies": {  
    "express": "^4.18.2"  
  }  
}
```

b) Create index.js

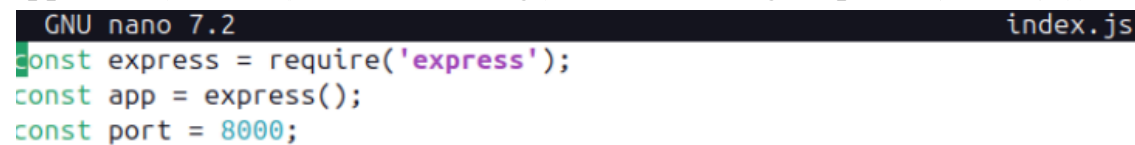
`nano index.js`

Paste this:

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

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

```
const PORT = process.env.PORT || 8080;
app.listen(PORT, () => console.log(`Server running on port ${PORT}`));
```



```
GNU nano 7.2 index.js
const express = require('express');
const app = express();
const port = 8000;
```

```
app.get('/', (req, res) => {
  res.send('Hello from Multi-Stage Dockerfile');
});

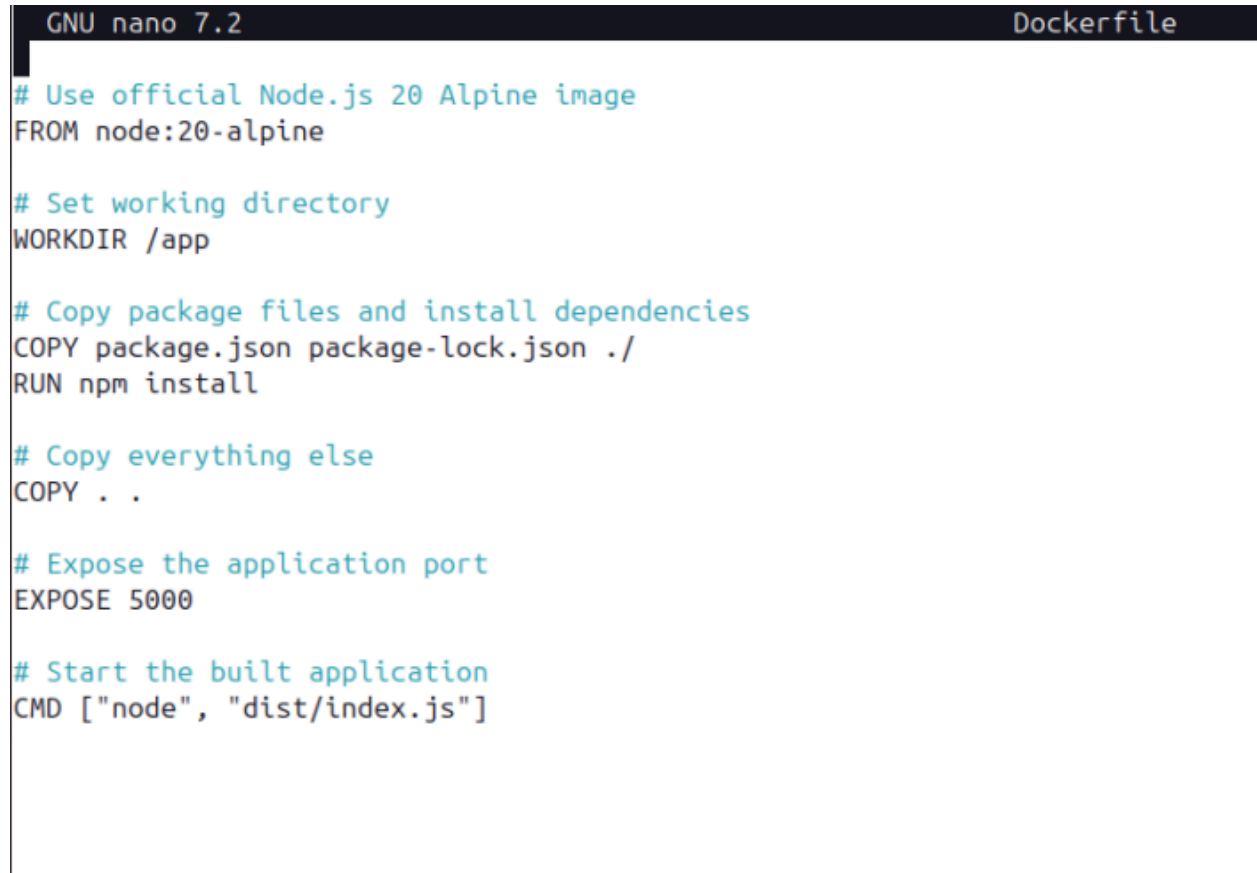
app.listen(port, () => {
  console.log(`Server running on port http://localhost:${port}`);
});
```

Step 4: Create Dockerfile

```
# ----- Stage 1: Builder -----
FROM node:18-alpine AS builder
WORKDIR /app
COPY app/package*.json ./
RUN npm install --only=production
COPY app/ .

# ----- Stage 2: Runtime -----
FROM node:18-alpine
```

```
WORKDIR /app
COPY --from=builder /app /app
EXPOSE 8080
CMD ["npm", "start"]
```

A screenshot of a terminal window showing a nano editor editing a file named 'Dockerfile'. The editor's title bar shows 'GNU nano 7.2' on the left and 'Dockerfile' on the right. The content of the file is a Dockerfile with the following instructions: '# Use official Node.js 20 Alpine image', 'FROM node:20-alpine', '# Set working directory', 'WORKDIR /app', '# Copy package files and install dependencies', 'COPY package.json package-lock.json ./', 'RUN npm install', '# Copy everything else', 'COPY . .', '# Expose the application port', 'EXPOSE 5000', and '# Start the built application', 'CMD ["node", "dist/index.js"]'.

```
GNU nano 7.2 Dockerfile
# Use official Node.js 20 Alpine image
FROM node:20-alpine

# Set working directory
WORKDIR /app

# Copy package files and install dependencies
COPY package.json package-lock.json ./
RUN npm install

# Copy everything else
COPY . .

# Expose the application port
EXPOSE 5000

# Start the built application
CMD ["node", "dist/index.js"]
```

Step 5: Build Docker Image

```
docker build -t program2-app .
```

Step 6: Run the Container

```
docker run -d -p 3000:3000 program2-app
```

Check running containers:

```
docker ps
```

Access the application in browser:

http://localhost:3000

