Program 2

lab2-devops/
│
├── Dockerfile
├── package.json
├── package-lock.json
├── server.js
├── node_modules/
└── README.md

Dockerfile:

```dockerfile
FROM node:18-alpine AS builder

ENV APP_PORT=3000

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

RUN npm run build

EXPOSE ${APP_PORT}

ENV PORT=${APP_PORT}

CMD ["node", "server.js"]


## developer stage
FROM node:18-alpine AS tester

ENV APP_PORT=3000

WORKDIR /app

COPY --from=builder /app /app
```

```
RUN adduser -D app

USER app

EXPOSE ${APP_PORT}

ENV PORT=${APP_PORT} NODE_ENV=development

CMD ["node", "server.js"]
```

server.js:
```
const express = require('express');
const app = express();
const port = 3000;

app.get('/', (req, res) => {
 res.send('Hello World from docker container!');
});

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

Setup and Installation:
Verify Node.js and npm installation:
Node –version
Npm –version

Initialize npm:
Npm init -y
(This creates a package.json file with default settings)

Install express:
Npm install express
(Express will be added to dependencies in package.json)

Build Builder Image:
sudo docker build –target builder -t jsapp-bui .

Build Tester Image:

sudo docker build –target tester -t jsapp-tes .

(This creates the final image using the built files from the first stage.)

(singh⊛singh)-[~/lab2-devops]
└─$ sudo docker run -d -p 3000:3000 --name jsbui jsapp-bui

Open:
http://localhost:3000

Hello World from docker container!