



ROITRAINING
MAXIMIZE YOUR TRAINING INVESTMENT™

Cheat Sheet: Docker

Dockerfile

- The following Dockerfile will work for both **external** and **internal** services
 - Create a file called Dockerfile **in both the internal and external folders:**

```
# Use base image for NodeJS
FROM node:14-alpine

# Copy application code.
COPY . /app/

# Change the working directory
WORKDIR /app

# Install dependencies.
RUN npm install

# Start the Express app
CMD ["node", "server.js"]
```

.dockerignore

- Create a file called `.dockerignore` **in both the internal and external folders:**

```
node_modules  
npm-debug.log
```

Build

- To build external, from the **external** folder:
 - `docker build -t external:v1.0 .`
- To build internal, from the **internal** folder:
 - `docker build -t internal:v1.0 .`

Run Locally

- Verify you have stopped the services from earlier (use <CTRL>+C)
- To run internal:
 - `docker run -d -p 8082:8082 -e SERVICE_PORT=8082 internal:v1.0`
- To run external:
 - `docker run -d -p 8080:8080 -e SERVER='http://localhost:8082' --network="host" external:v1.0`
- Test your app by previewing on port 8080
- Other commands to try:
 - `docker images`
 - `docker ps -a`
 - `docker stop <ContainerID>`
 - `docker rm <ContainerID>`

Save All Changes to GitHub

- Add, commit, and push your changes
 - For each repo

Success!

- **Congratulations!** You have containerized the case study