

BUILDING, RUNNING, AND DEPLOYING A NODE.JS APPLICATION WITH DOCKER

Create a Dockerfile

- The Dockerfile is a blueprint for your Docker image. Let's set up a basic Node.js application:

Use the official Node.js image from the Docker Hub

FROM node:14

Create and set the working directory

WORKDIR /app

Copy package.json and install dependencies

COPY package*.json ./

RUN npm install

Copy the project files

COPY . .

Expose the port

EXPOSE 3000

Command to run the application

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

- This Dockerfile will create an image that runs a simple Node.js application.

Create the Application

- Create a file named app.js in the same directory:

```
const http = require('http');
```

```
const server = http.createServer((req, res) => {
```

```
  res.statusCode = 200;
```

```
  res.setHeader('Content-Type', 'text/plain');
```

```
res.end('Hello, Docker!\n');  
  
});  
  
server.listen(3000, () => {  
  
  console.log('Server running on port 3000');  
  
});
```

5. Build the Docker Image

- In the terminal, build the Docker image with the following command:

`docker build -t docker_demo_app .`

6. Run the Docker Container

- Once the image is built, you can run a container using this image:

`docker run -p 3000:3000 docker_demo_app`

- This will start the container, and you should see Server running on port 3000 in the terminal.

7. Test the Application

- Open a browser or use curl to access the application at <http://localhost:3000>.
- You should see the message Hello, Docker!.