

Rest Api With Typescript & Docker!

Let's Gooo.....



Create a File in Express

Ensure that you have set up your Nodejs application using the Express framework. Here is the index.js file:

```
import express, { Request, Response } from 'express';

const app = express();
const PORT = 3001;

app.get('/', (req: Request, res: Response) => {
  res.json({ message: 'Hello, Bayu Widia Santoso!' });
});

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

Create a Dockerfile for the application

Create a Dockerfile in the same directory as your index.js file:

```
# Stage 1: Build the app
FROM node:18-alpine as build

WORKDIR /app

COPY package*.json ./
RUN npm install

COPY . .
RUN npm run build

# Stage 2: Run the app
FROM node:18-alpine

WORKDIR /app

COPY --from=build /app/dist ./dist
COPY --from=build /app/package*.json ./

RUN npm install --production

EXPOSE 3001
CMD ["node", "dist/index.js"]
```

Set up a Docker Compose configuration.



Create a docker-compose.yml file in the same directory as your index.js file :

```
version: '3.8'
services:
  app:
    build:
      context: .
      dockerfile: Dockerfile
    ports:
      - "3001:3001"
    environment:
      NODE_ENV: production
```

Docker Desktop

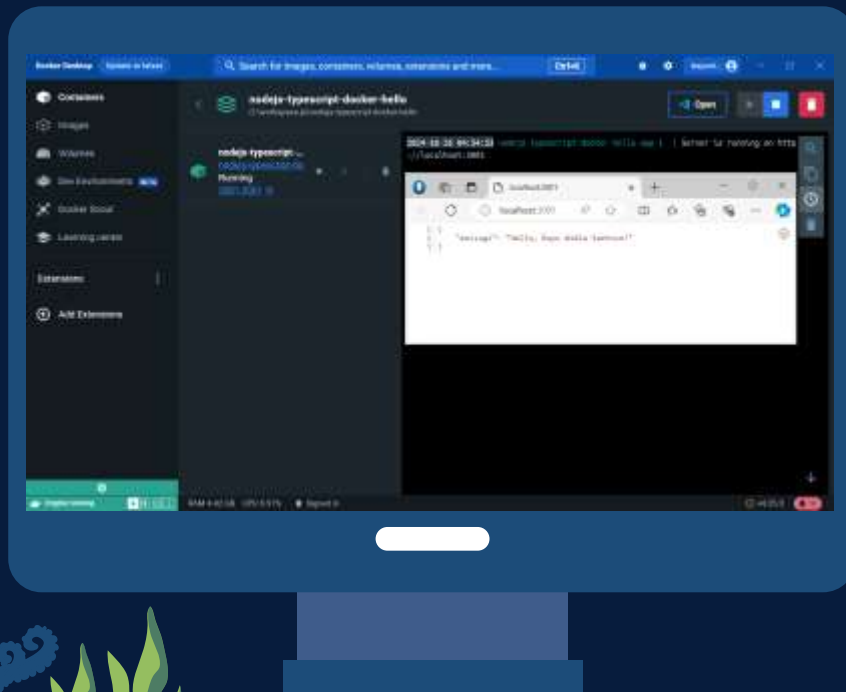
TS

Rebuild the Docker image

`docker-compose down`

`docker-compose build --no-cache`

`docker-compose up`





SEE YOU...

