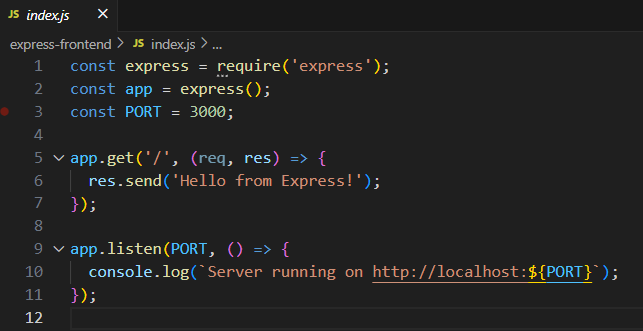
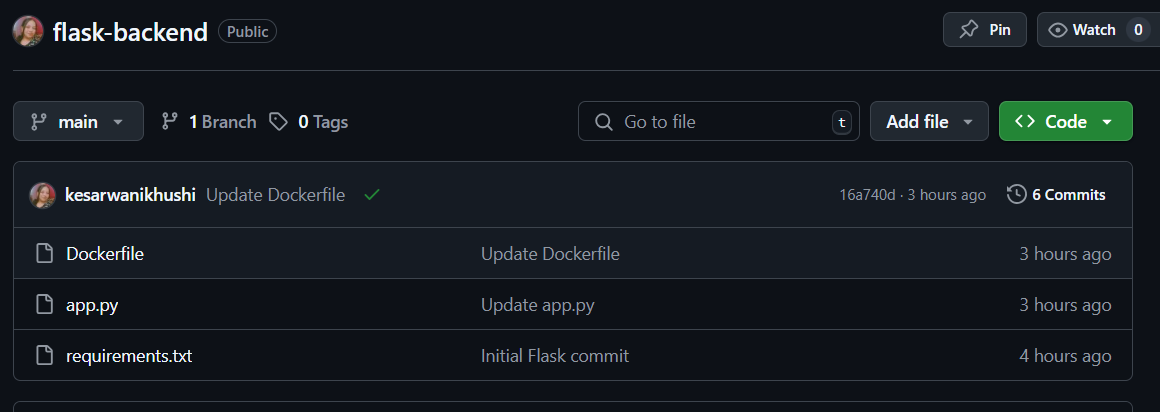
# AWS Assignment: Express Frontend Deployment

This document outlines the process of deploying an Express.js frontend application using AWS infrastructure (Docker-based deployment simulated on a platform compatible with Docker).

## Step 1: Prepare Your Project Repository

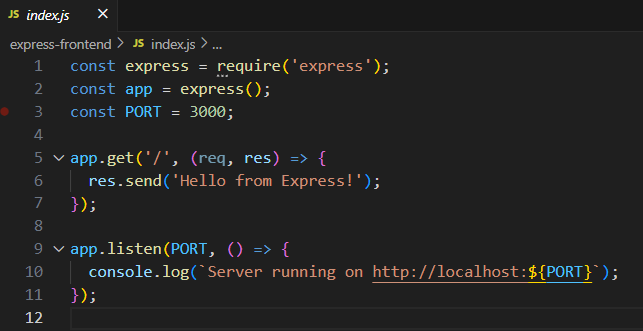
- Ensure your Express.js app runs locally using `npm start` or `node app.js`.  
- Include a valid `Dockerfile` in your project root.  
- Push your project to GitHub.

## Step 2: Deploy to AWS

- Log in to your AWS Docker-compatible hosting dashboard.  
- Click "New Project" → "Deploy from GitHub Repo".  
- Select your Express.js GitHub repository.

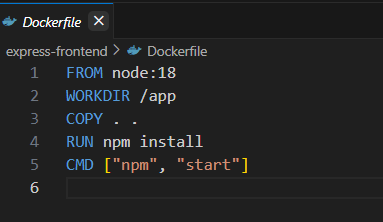
## Step 3: Configure Build and Service Settings

- No need to set install or start commands (Docker handles that).  
- Set the `PORT` environment variable to `3000` (or the port your Express app uses).



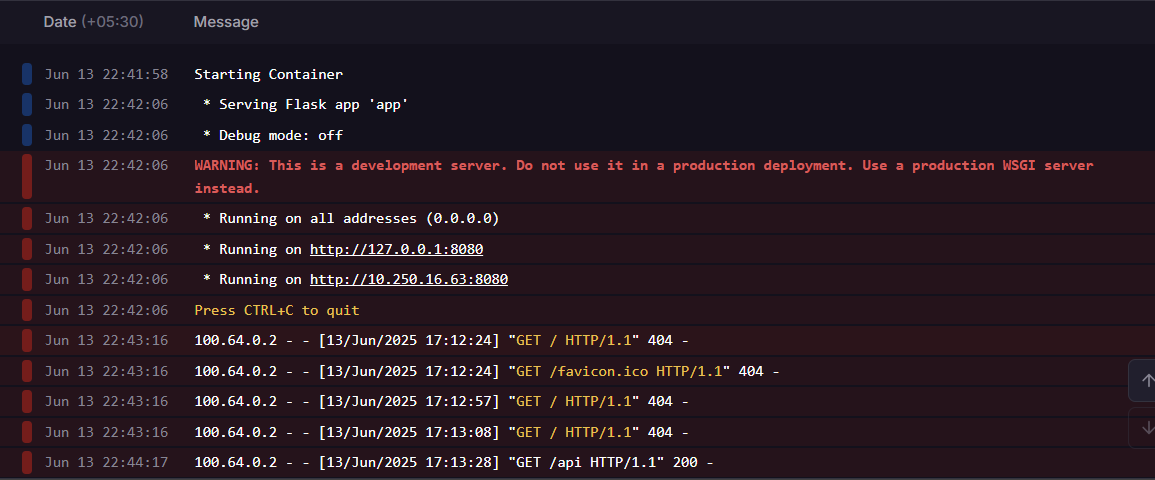
## Step 4: Dockerfile Configuration

Ensure your Dockerfile includes the following (if not already set):

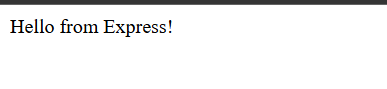
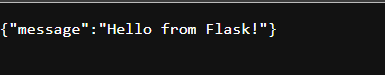


## Step 5: Build and Logs

- Once deployment starts, monitor build logs to ensure there are no errors.  
- After successful build, check container logs for output like:  
 - `Listening on port 3000`



## Step 6: Accessing Your App

- After deployment, a public URL will be provided.  
- Visit the URL to verify that your frontend is running correctly.

## Github Repo Link:

https://github.com/kesarwanikhushi/flask-backend

https://github.com/kesarwanikhushi/express-frontend