Created folder **elevatelab**

**Inside it**

Task1/

├── app.js

├── package.json

├── Dockerfile

└── .github/

└── workflows/

└── main.yml

--------------------------------------------------------------------------------------------------------------------------------------

**App.js**

const express = require("express");

const app = express();

const PORT = process.env.PORT || 5000;

app.get("/", (req, res) => {

    res.send("Pipeline is Successfully completed....!");

});

app.listen(PORT, () => {

    console.log(`Server is running on port ${PORT}`);

});

**---------------------------------------------------------------------------------------------------------------------------------**

**Dockerfile**

FROM node:18

WORKDIR /app

COPY package\*.json ./

RUN npm install

COPY . .

EXPOSE 5000

CMD ["npm", "start"]

**Package.json**

{

    "name": "cicd-task",

    "version": "1.0.0",

    "main": "app.js",

    "scripts": {

        "start": "node app.js",

        "test": "echo \"No tests yet\""

    },

    "dependencies": {

        "express": "^4.18.2"

    }

}

**---------------------------------------------------------------------------------------------------------------------------------**

**Main.yml**

name: CI/CD Pipeline

on:

  push:

    branches:

      - main  # Trigger the workflow when code is pushed to the main branch

jobs:

  build-and-deploy:

    runs-on: ubuntu-latest

    steps:

      # Checkout repository

      - name: Checkout code

        uses: actions/checkout@v3

      # Set up Node.js

      - name: Set up Node.js

        uses: actions/setup-node@v3

        with:

          node-version: "18"

      # Install dependencies

      - name: Install dependencies

        run: npm install

      # Run tests

      - name: Run tests

        run: npm test

      # Build Docker image

      - name: Build Docker image

        run: docker build -t ${{ secrets.DOCKERHUB\_USERNAME }}/elevatelab:latest .

      # Login to DockerHub

      - name: Login to DockerHub

        uses: docker/login-action@v3

        with:

          username: ${{ secrets.DOCKERHUB\_USERNAME }}

          password: ${{ secrets.DOCKERHUB\_TOKEN }}

      # Push Docker image to DockerHub

      - name: Push Docker image to DockerHub

        run: docker push ${{ secrets.DOCKERHUB\_USERNAME }}/elevatelab:latest





