# Disclaimer: This output contains AI-generated content; user is advised to review it before consumption.  
  
#\*Start of AI Generated Content\*

# **Docker Deployment Steps for Anomaly Detection Code**

## **Step 1: Create a Dockerfile**

Create a new file named Dockerfile in the root directory of your project with the following contents:

# Use an official Python image as the base  
FROM python:3.9-slim  
  
# Set the working directory to /app  
WORKDIR /app  
  
# Copy the requirements file  
COPY requirements.txt .  
  
# Install the dependencies  
RUN pip install --no-cache-dir -r requirements.txt  
  
# Copy the application code  
COPY . .  
  
# Expose the port  
EXPOSE 8000  
  
# Run the command to start the development server  
CMD ["python", "main.py"]

## **Step 2: Create a requirements.txt File**

Create a new file named requirements.txt in the root directory of your project with the following contents:

pandas  
numpy  
matplotlib  
scikit-learn

## **Step 3: Build the Docker Image**

Run the following command to build the Docker image:

docker build -t anomaly-detection .

## **Step 4: Run the Docker Container**

Run the following command to start a new container from the anomaly-detection image:

docker run -p 8000:8000 anomaly-detection

## **Step 5: Verify the Deployment**

Open a web browser and navigate to http://localhost:8000 to verify that the application is running correctly.

## **Step 6: Push the Image to Docker Hub (Optional)**

If you want to share your image with others or deploy it to a cloud platform, you can push it to Docker Hub. First, create a Docker Hub account and then run the following command:

docker tag anomaly-detection:latest <your-username>/anomaly-detection:latest  
docker push <your-username>/anomaly-detection:latest

Replace <your-username> with your actual Docker Hub username.

## **Step 7: Deploy to a Cloud Platform (Optional)**

You can deploy your containerized application to a cloud platform like AWS, Google Cloud, or Azure. The deployment process will vary depending on the platform you choose.

#\*End of AI Generated Content\*