Steps to Run FastAPI on EC2 Ubuntu

Step 1: SSH into your EC2 instance

```
ssh -i "your-key.pem" ubuntu@your-ec2-public-ip
```

- Replace `"your-key.pem"` with your private key. - Replace `your-ec2-public-ip` with your EC2's public IP.

Step 2: Update packages

```
sudo apt update && sudo apt upgrade -y
```

Step 3: Install Python & pip

```
sudo apt install python3 python3-pip python3-venv -y
```

Step 4: Navigate to your project folder

```
cd /home/ubuntu/myproject
```

(Replace `myproject` *with your FastAPI project folder.)*

Step 5: Create a virtual environment

```
python3 -m venv venv
```

Step 6: Activate the virtual environment

```
source venv/bin/activate
```

- Your prompt should now show `(venv)` at the beginning.

Step 7: Install dependencies

```
pip install --upgrade pip
pip install -r requirements.txt
```

Step 8: Run FastAPI with nohup (keep it running in background)

1. Run Uvicorn with `nohup`:

   ```
   nohup uvicorn main:app --host 0.0.0.0 --port 8000 &
   ```

2. `nohup` keeps it running after logout.
3. `&` runs it in the background.
4. Output will be written to `nohup.out` by default.
5. Optional: Log output to a custom file:

   ```
   nohup uvicorn main:app --host 0.0.0.0 --port 8000 > fastapi.log 2>&1 &
   ```

6. Check running process:

   ```
   ps aux | grep uvicorn
   ```

7. Stop process if needed:

   ```
   kill <PID>
   ```

Step 9: Allow traffic through port 8000

- Go to your **EC2 Security Group** → Inbound rules → Add rule:
- Type: Custom TCP
- Port: 8000
- Source: Anywhere (0.0.0.0/0) *(for testing)*

- Access your app via:

```
http://your-ec2-public-ip:8000
```

Optional Step 10: Use Nginx as reverse proxy (Production)

- For production, serve your FastAPI app via **Nginx** on port 80 while Uvicorn runs on 8000. This removes the need to specify port 8000 in the URL.