Table of Contents

[Table of Contents 1](#_Toc161940386)

[Dockerized Django Server 2](#_Toc161940387)

[Requirements 2](#_Toc161940388)

[Installation 2](#_Toc161940389)

[1. Open Command Prompt 2](#_Toc161940390)

[2. Cloning the GitHub Repository 2](#_Toc161940391)

[3. Performing the specific steps in Docker Desktop 3](#_Toc161940392)

[Usage 7](#_Toc161940393)

[Open django script from container using Visual Code Studio 8](#_Toc161940394)

Dockerized Django Server

This project provides a Dockerized environment for running a Django server. By downloading the files onto your local machine and performing the specific steps in Docker Desktop (for Dev Environment), a Docker container will be created and initialized with the Django server. You can then modify the Django script using Visual Studio Code on your local machine.

Requirements

Make sure you have installed all of the following prerequisites on your development machine:

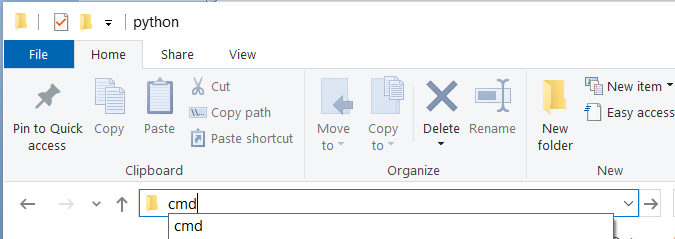
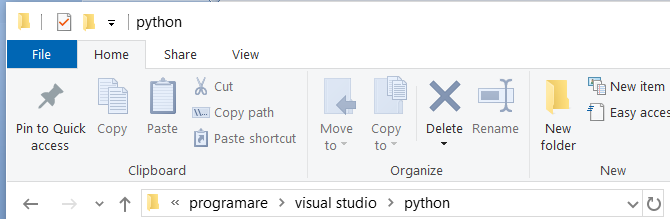
* Visual Code Studio (<https://code.visualstudio.com/>)
* Visual Studio Code Remote Containers Extension (<https://marketplace.visualstudio.com/items?itemName=ms-vscode-remote.remote-containers>)
* Docker Desktop (<https://www.docker.com/products/docker-desktop/>)
* Git (<https://git-scm.com/>)

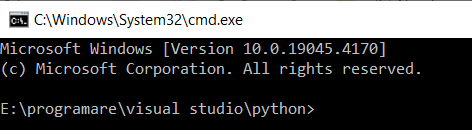
Installation

To get started with this project, follow these steps:

1. Open Command Prompt

Run Command Prompt and open installation folder:

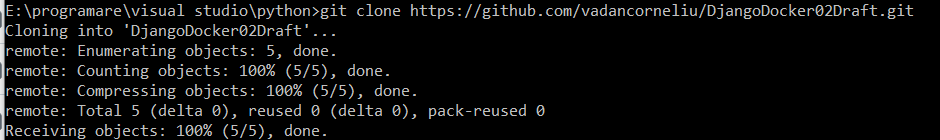




2. Cloning the GitHub Repository

The recommended way to get installation files is to use git to directly clone the repository:

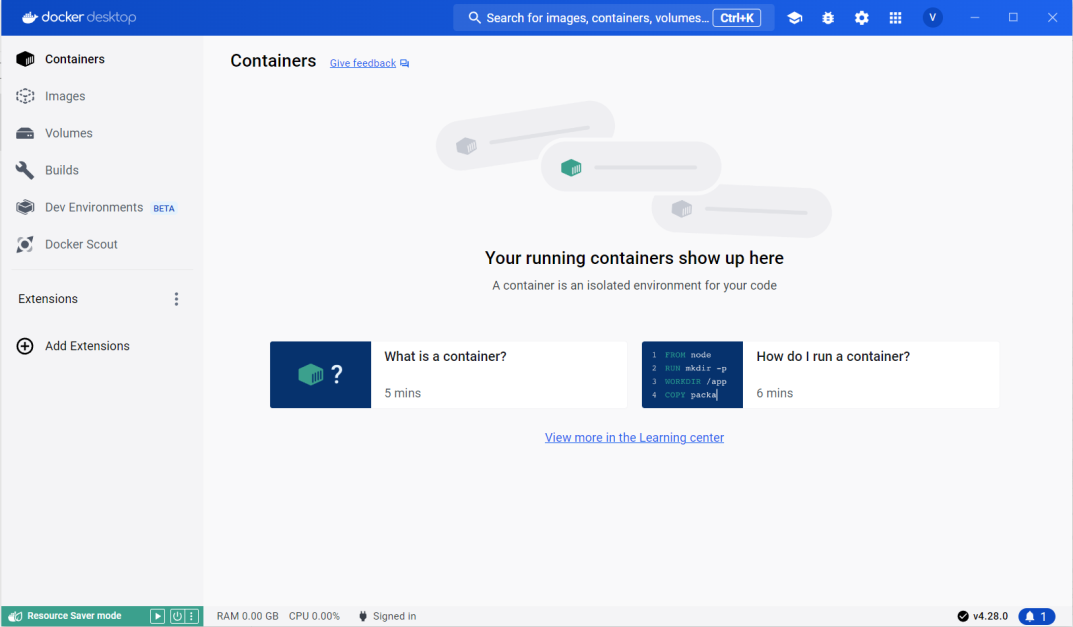
git clone https://github.com/vadancorneliu/DjangoDocker02Draft.git



3. Performing the specific steps in Docker Desktop

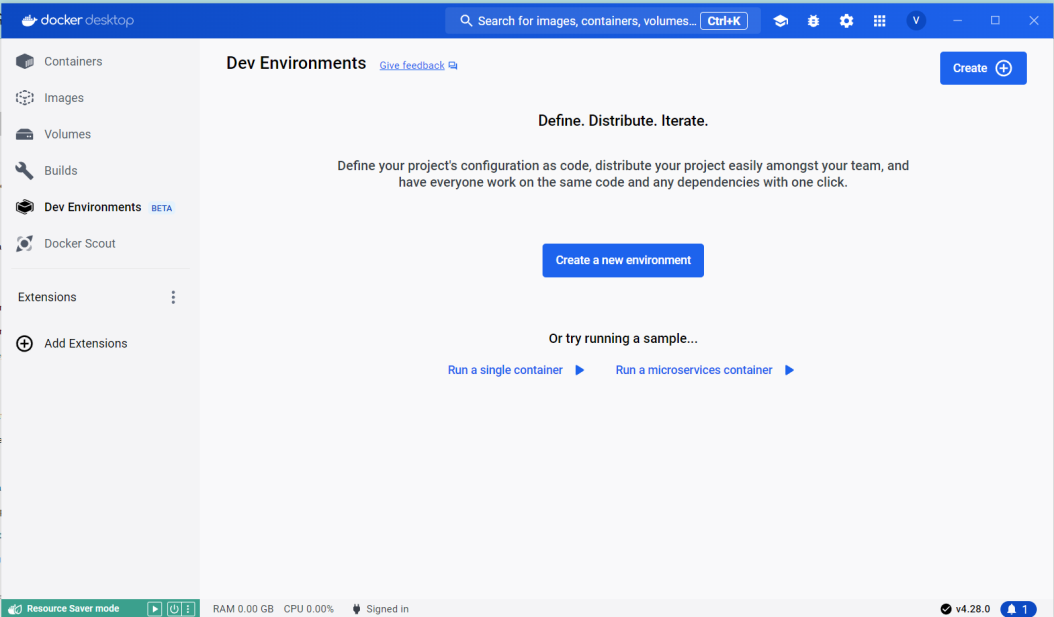
Launch the Docker Desktop application on your Windows system.

In the Docker Desktop interface, navigate to the "*Dev Environments*" section,





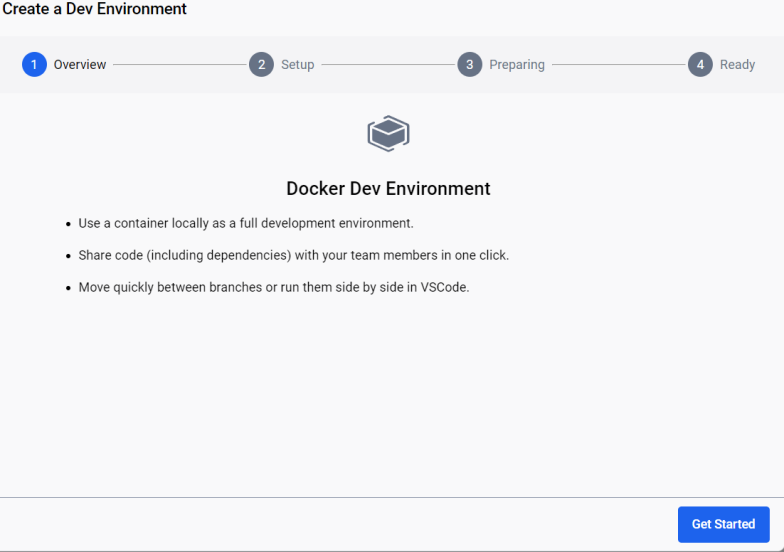
click on the "*Create a new environment*" button and follow these steps:





Step1 – Overview – Create a Dev Environment

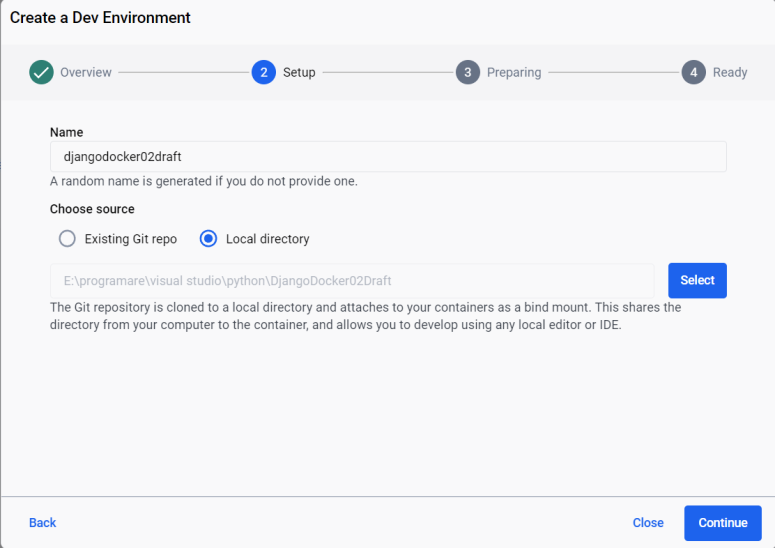
* click on the "*Get Started*" button





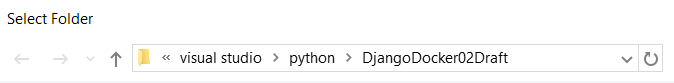
Step2 – Setup – Create a Dev Environment

* input dev environment name in the text field with label "*Name*"
* select "*Local directory*" option and click on the "*Select*" button





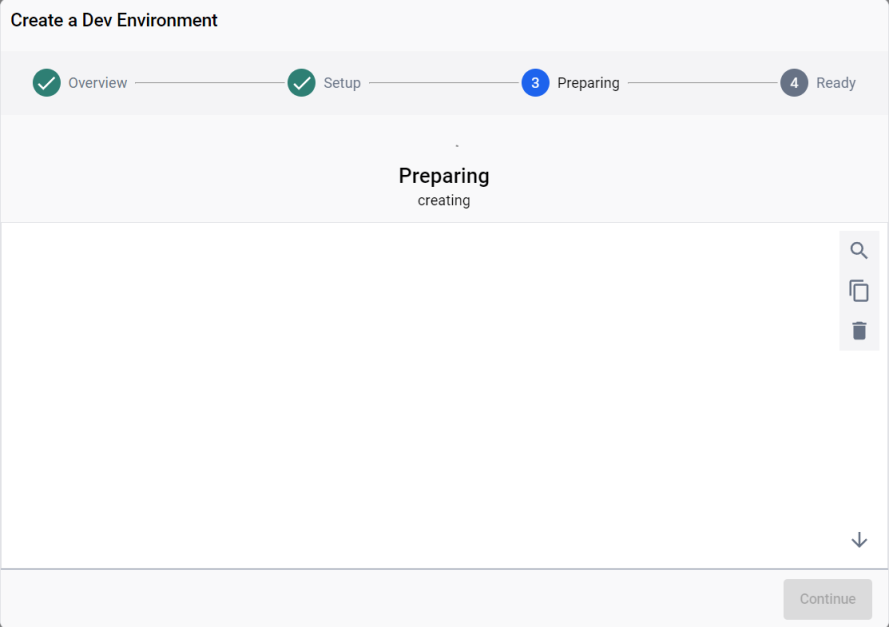
* select the project folder and click on the "*Select Folder*" button

* click on the "*Continue*" button



Step 3 – Preparing – Create a Dev Environment



* the following messages are displayed in the Docker Desktop terminal for:
  + create a custom image from Dockerfile (name and tag image is defined in the compose-dev.yaml file)

Detecting main repo language...

#0 building with "default" instance using docker driver

#1 [app internal] load build definition from Dockerfile

#1 DONE 0.1s

#1 [app internal] load build definition from Dockerfile

#1 transferring dockerfile:

#1 transferring dockerfile: 607B 0.2s done

#1 DONE 0.9s

#2 [app internal] load metadata for docker.io/library/python:3.11-slim

#2 ...

#3 [app auth] library/python:pull token for registry-1.docker.io

#3 DONE 0.0s

#2 [app internal] load metadata for docker.io/library/python:3.11-slim

#2 DONE 1.8s

#4 [app internal] load .dockerignore

#4 transferring context: 2B done

#4 DONE 0.1s

#5 [app 1/4] FROM docker.io/library/python:3.11-slim@sha256:a2eb07f336e4f194358382611b4fea136c632b40baa6314cb27a366deeaf0144

#5 DONE 0.0s

#6 [app internal] load build context

#6 transferring context: 265B done

#6 DONE 0.1s

#7 [app 2/4] WORKDIR /django-app

#7 CACHED

#8 [app 3/4] COPY requirements.txt ./

#8 CACHED

#9 [app 4/4] RUN pip install --no-cache-dir -r requirements.txt

#9 CACHED

#10 [app] exporting to image

#10 exporting layers done

#10 writing image sha256:fd9400d5952a644fd2ebf3c4e98e8fd1d7c355cf9fbf68e73511ff7d6527d3a6 done

#10 naming to docker.io/library/django02draft\_img:1.0 0.0s done

* + create container and run the service defined in the compose-dev.yaml file,

#10 DONE 0.1s

Network djangodocker02draft-naughty\_nash\_default Creating

Network djangodocker02draft-naughty\_nash\_default Created

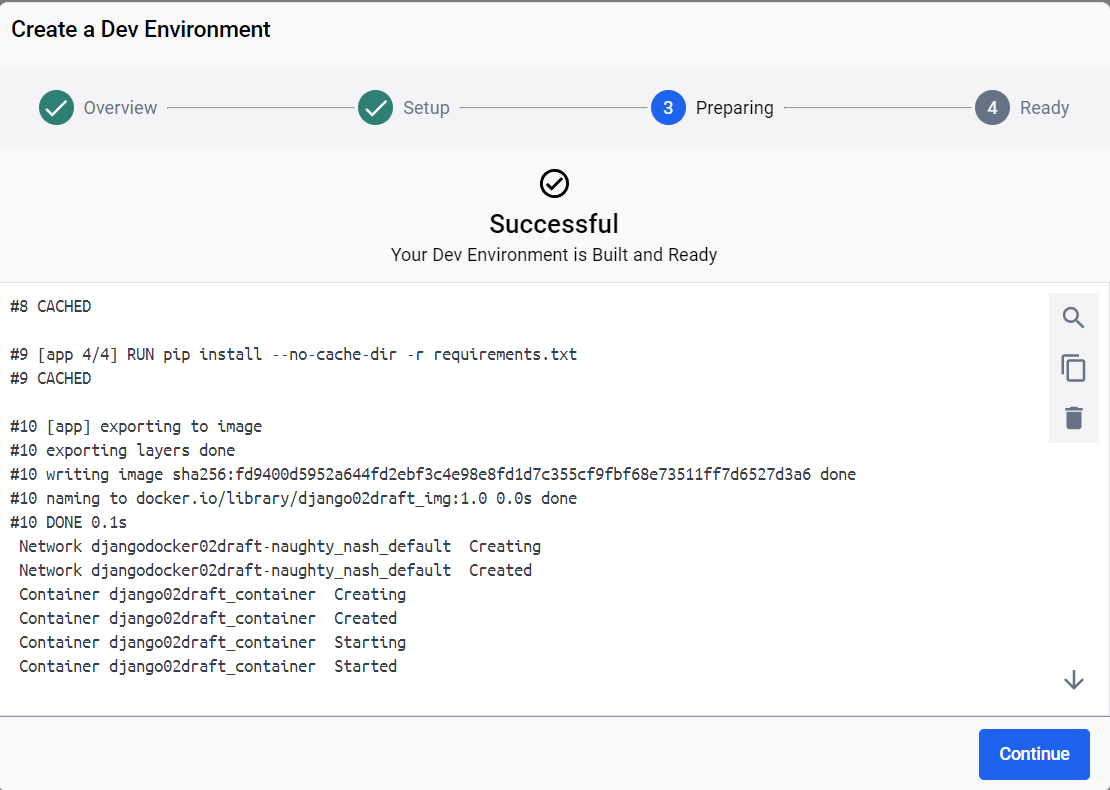
Container django02draft\_container Creating

Container django02draft\_container Created

Container django02draft\_container Starting

Container django02draft\_container Started

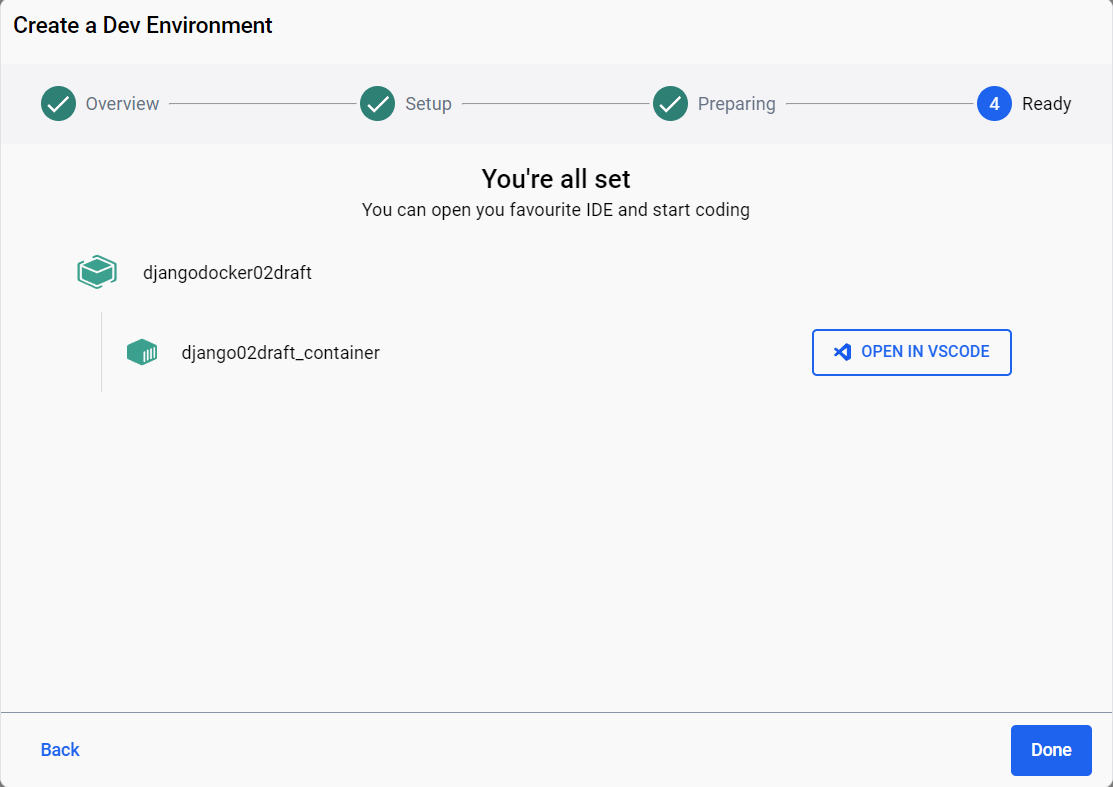
* once you've configured the container, click on the "*Continue*" button



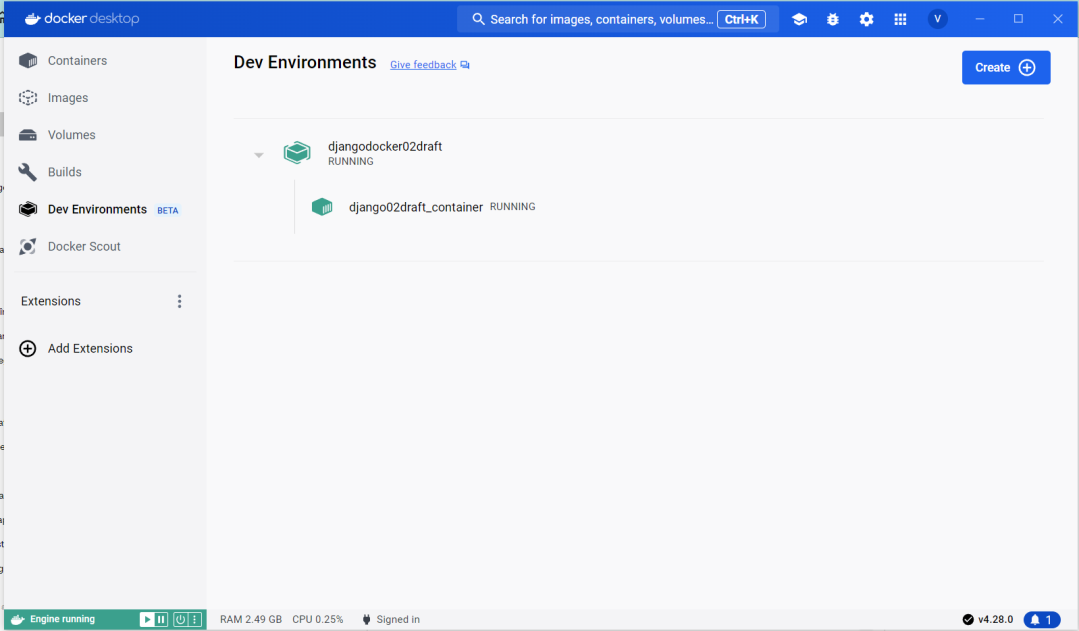


Step 4 – Ready – Create a Dev Environment

* click on the "*Done*" button



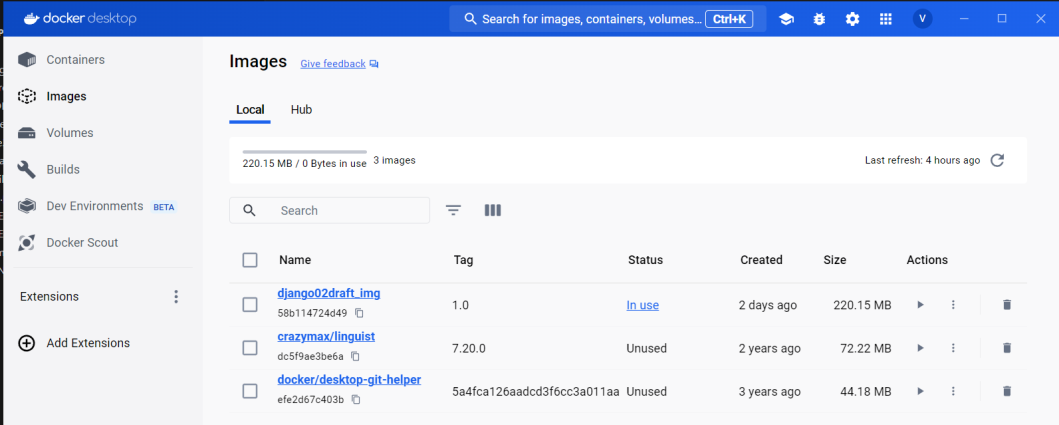




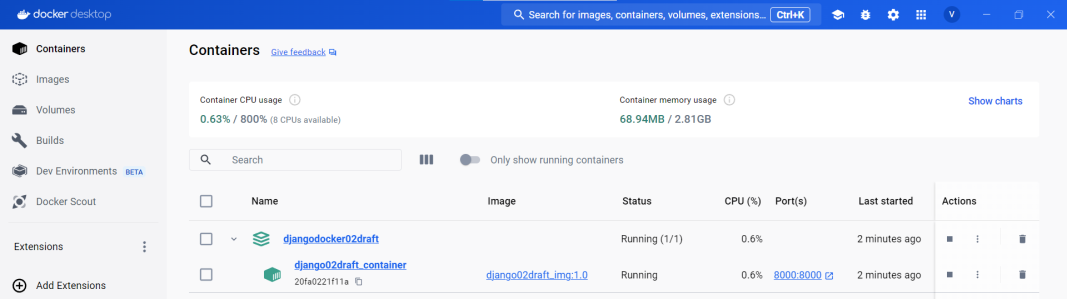
Usage

After dev environment is configured, we have:

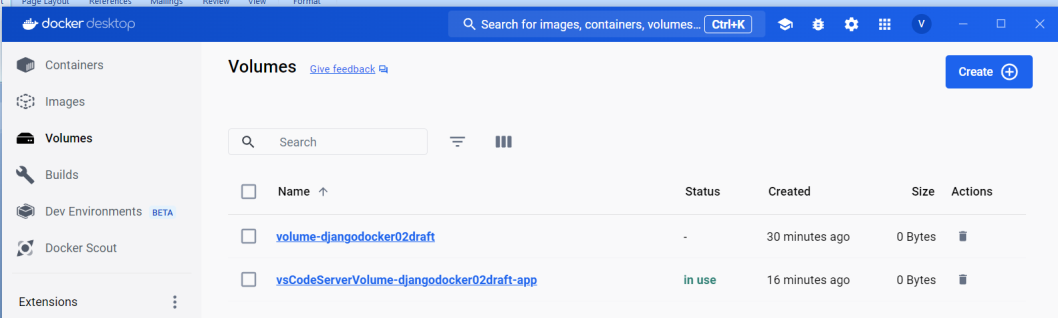
* Docker images



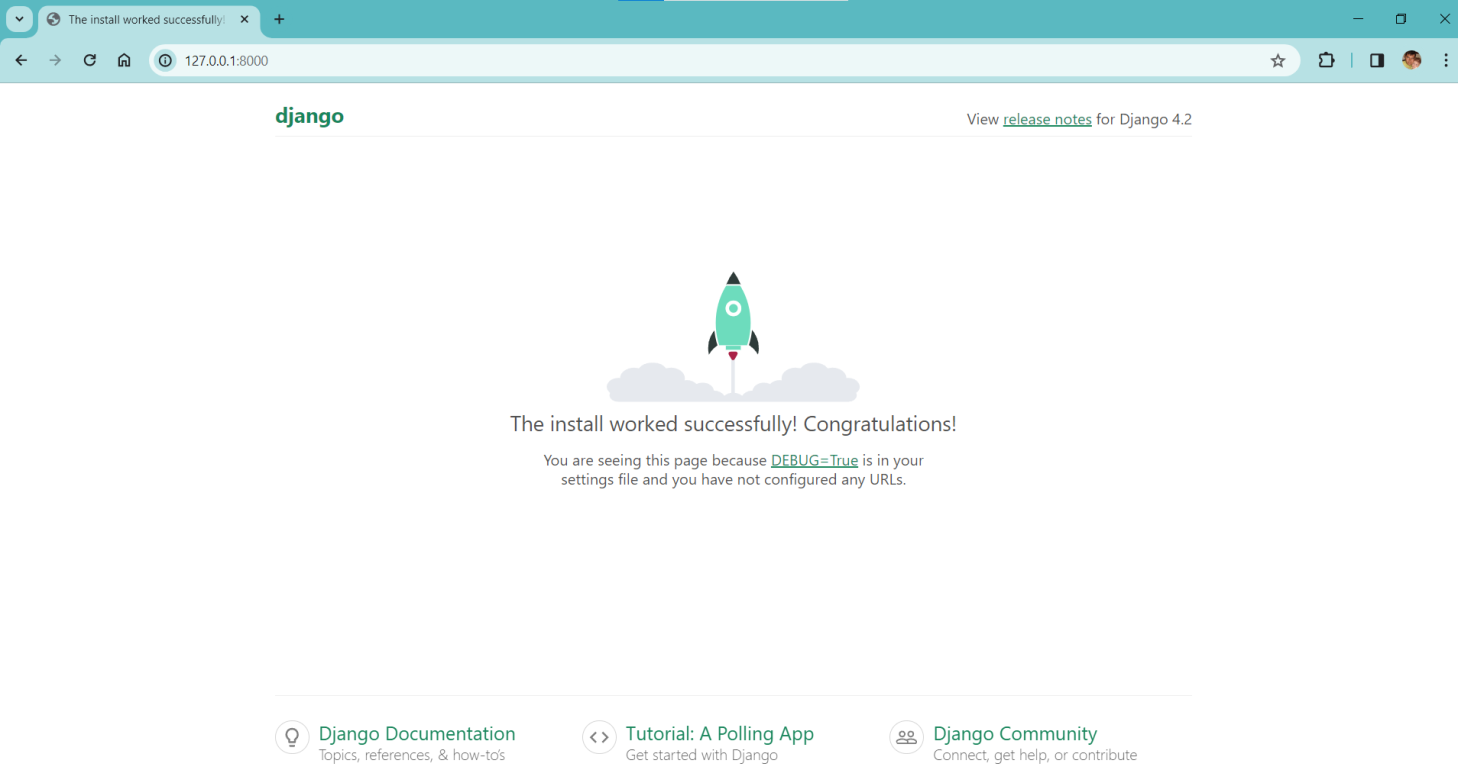
* Docker container



* Docker volumes

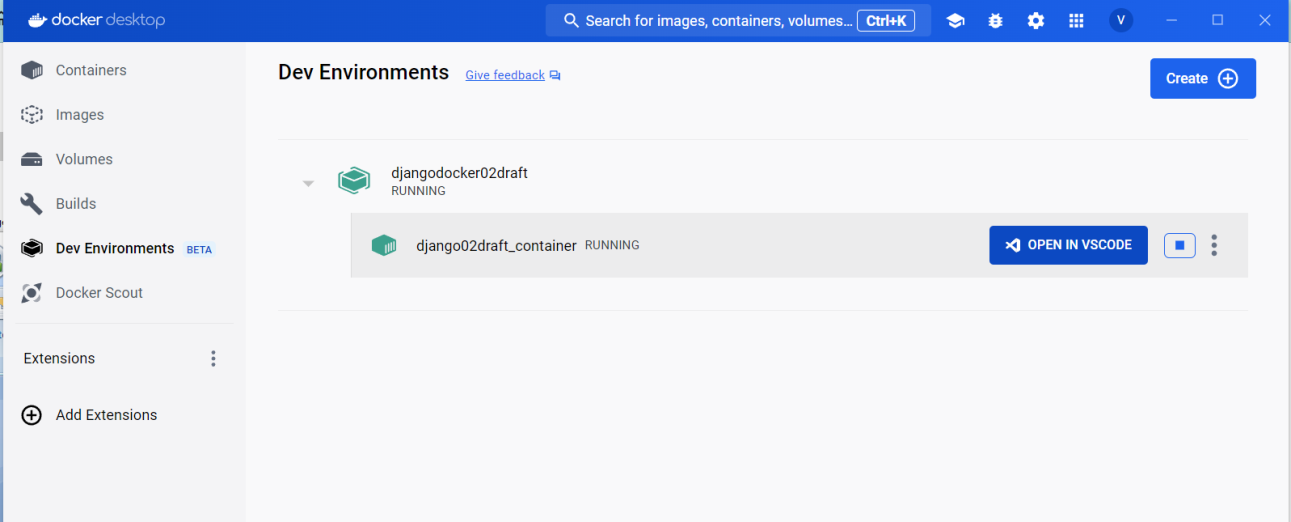


Once the container is up and running, you can access the Django server locally at <http://localhost:8000>.



Open django script from container using Visual Code Studio

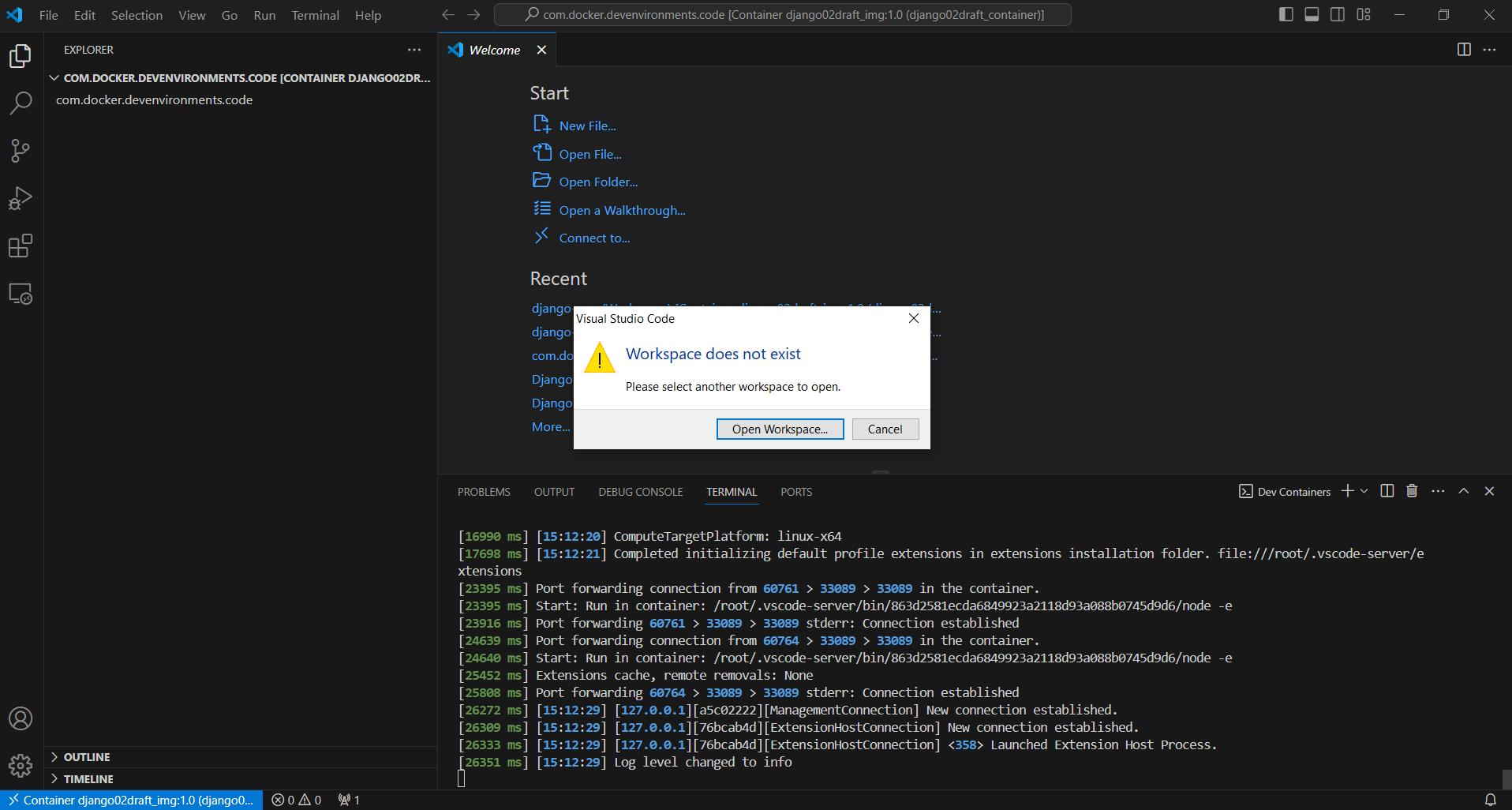
In the Docker Desktop interface, navigate to the "*Dev Environments*" section, select the appropriate container and click the "*OPEN IN VSCODE*" button.





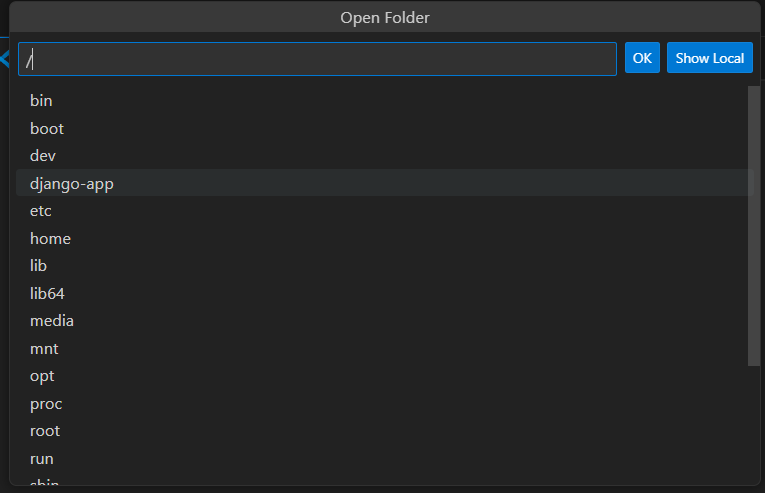
Visual Code Studio is launching and follow these steps:

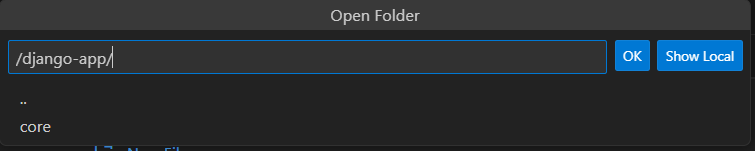
* once the windows with the message ”Workspace does not exist” is open, click the ”Open Workspace...” button



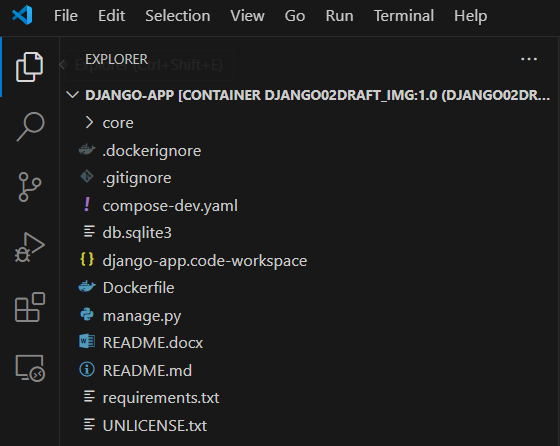


* select the project folder and click on the "*OK* button

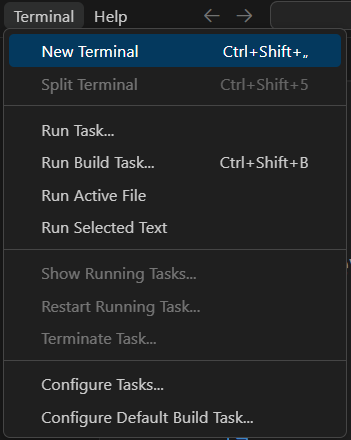




Now, Visual Code Studio is open and in Explorer section we have project files.



Open a new terminal from Visual Code Studio menu (*Terminal > New Terminal*)



* bash terminal:

