

Containerize a .NET application

Prerequisites

- You have installed the latest version of [Docker Desktop](#).
- You have a [git client](#) . The examples in this section use a command-line based git client, but you can use any client.

Overview

This section walks you through containerizing and running a .NET application.

Give feedback

Get the sample applications

In this guide, you will use a pre-built .NET application. The application is similar to the application built in the Docker Blog article, [Building a Multi-Container .NET App Using Docker Desktop](#) .

Open a terminal, change directory to a directory that you want to work in, and run the following command to clone the repository.

```
$ git clone https://github.com/docker/docker-dotnet-sample
```

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

[Cookies Settings](#)

[Reject All](#)

[Accept All Cookies](#)

you'll need to answer a few questions about your application. Refer to the following example to answer the prompts from `docker init` and use the same answers for your prompts.

```
$ docker init
Welcome to the Docker Init CLI!

This utility will walk you through creating the following files with sensible defaults:
- .dockerignore
- Dockerfile
- compose.yaml
- README.Docker.md

Let's get started!

? What application platform does your project use? ASP.NET Core
? What's the name of your solution's main project? myWebApp
? What version of .NET do you want to use? 6.0
? What local port do you want to use to access your server? 8080
```

You should now have the following contents in your `docker-dotnet-sample` directory.

```
├─ docker-dotnet-sample/
│  ├─ .git/
│  ├─ src/
│  ├─ .dockerignore
│  ├─ compose.yaml
│  ├─ Dockerfile
│  ├─ README.Docker.md
│  └─ README.md
```

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

Run the application

Inside the `docker-dotnet-sample` directory, run the following command in a terminal.

```
$ docker compose up --build
```

Open a browser and view the application at <http://localhost:8080> . You should see a simple web application.

In the terminal, press `ctrl+c` to stop the application.

Run the application in the background

You can run the application detached from the terminal by adding the `-d` option. Inside the `docker-dotnet-sample` directory, run the following command in a terminal.

Open a browser and view the application at <http://localhost:8080> . You should see a simple web application.

In the terminal, run the following command to stop the application.

```
$ docker compose down
```

For more information about Compose commands, see the [Compose CLI reference](#).

Give feedback

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.

- [.dockerignore file reference](#)
- [Docker Compose overview](#)

Next steps

In the next section, you'll learn how you can develop your application using Docker containers.

Develop your application

Product offerings

Pricing

About us

Support

Contribute

Give feedback

Copyright © 2013-2024 Docker Inc. All rights reserved.

By clicking "Accept All Cookies", you agree to the storing of cookies on your device to enhance site navigation, analyze site usage, and assist in our marketing efforts.