Use Docker compose to automatically build images and start container

Contents

Introduction

Docker compose is normally used to build, start and stop multiple containers in a coordinated manner. But compose can also be used to start and stop a single container. In fact, doing so has the nice effect that you don't have to manually build the image you want to run in a container, compose will automatically build your images if needed (as long as you created the necessary Dockerfile(s)).

Your assignment for this exercise is to add a Docker compose file (compose.yaml) to the small Java Hello World project from exercise 2-2.

Step 1- Create a Docker compose file (compose.yaml)

Create a new file called compose.yaml in the top of the project. Put the necessary stuff in the file. Here are some hints:

- The compose file only need to define a single service, that uses the image you created in exercise 2-2 (i.e. the service just needs to know where to find the Dockerfile you created in exercise 2-2).
- The compose don't have to configure any network stuff (ports and things) because our application (Hello World) doesn't use any network things.
- Look at these examples/tutorials:
  https://github.com/DaFessor/devops-ci-demo/tree/main (code example)
  https://docs.docker.com/compose/compose-file/05-services/#image (documentation)

Step 2 – Start your container with compose

Once your compose.yaml is ready, you should be able to start your HelloWorld image by:

1. Start a Linux in Windows Terminal
2. Do the `cd /mnt/c.....` thing to change into your project
3. Type the command

```
docker compose up
```

The container runs and then terminates after having written the “Hello world ....” message. That’s okay, that’s because the application (the Java main() method) just writes the message and then terminates.

Step 3 – Delete all HelloWorld containers and images

Once your compose.yaml file works, and you can start the container with docker compose, try to delete all the HelloWorld images and containers in Docker Desktop.

When you then do the `docker compose up` thing, you will see that docker compose automatically builds your image if it does not exist, before it runs the container.