CS-548: Cloud-native Software Architectures

Computer Science Department • University of Crete
Instructor: Antony Chazapis

Website: https://www.csd.uoc.gr/~hy548/

Version: 202301

Assignment 1: Docker

1. Nginx is one of the most popular web servers. Container images are readily available on Docker Hub. Write down the commands needed to:
    a. Download the images tagged `1.23.3` and `1.23.3-alpine` locally.
    b. Compare the sizes of the two images.
    c. Start one of the two images in the background, with the appropriate network settings to forward port 80 locally and use a browser (or `curl` or `wget`) to see that calls are answered. What is the answer?
    d. Confirm that the container is running in Docker.
    e. Get the logs of the running container.
    f. Stop the running container.
    g. Start the stopped container.
    h. Stop the container and remove it from Docker.

2. Following the previous exercise, start an Nginx container again and issue the commands needed to:
    a. Open a shell session inside the running container and change the first sentence of the default page to "Welcome to MY nginx!". Close the session.
    b. From your computer's terminal (outside the container) download the default page locally and upload another one in its place.

    Close the container, delete it and start another instance.
    c. Do you see the changes? Why;

3. The code that produces the course's website is available on GitHub (https://github.com/chazapis/hy548). Write down the commands needed to download the repository (and submodules) and hugo (the tool that builds the website), build the website locally, and start an Nginx container to serve the CS-548 website instead of the default page.

4. Following the previous exercise, create your own container image, based on Nginx, that will contain the CS-548 website instead of the default page. Downloading the CS-548 repository (and submodules), hugo and building the site should be done in the Dockerfile. Create a Docker Hub account and upload the image. Provide:
    a. The Dockerfile.
    b. The command needed to upload the image to Docker Hub.

    Explain:
    c. How much bigger is your own image than the image you were based on. Why;

    d. What have you done in the Dockerfile to keep the image as small as possible?

5. Upload the Dockerfile from the previous exercise to your GitHub repository. Create a GitHub Action that will automatically build and push the image to your Docker Hub account (the workflow should be initiated by the user). Provide the YAML of the workflow you made.

Notes:

- The assignment is personal.
- All exercises contribute equally to the overall grade (unless individual percentages are defined).
- A day/time will be set for answering questions and giving clarifications.
- Write down your answers in a Markdown-formatted text file in either Greek or English and commit it (along with any other files) in a private GitHub repository before the exercise's deadline. Share the repository with the instructor (username "chazapis").