Challenges - Docker

Docker

How to solve these challenges:

- To be consistent with the filenames and paths run the commands on Ubuntu. Install it locally in a VM or at a cloud provider. Use my referral link below to create your account on DigitalOcean and you'll get $100 in credit over 60 days: https://m.do.co/c/f1d4debf213f
- Write your solution **in a terminal or in a shell script** and **test it.**
- If your solution is not correct, then try to understand the error messages, watch the video again, rewrite the solution, and test it again. Repeat this step until you get the correct solution.
- **Save** the solution in a file for future reference or recap.

Challenge #1

- Using the docker client, display the help of the docker management command **container** and the help of the **run** subcommand.

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #2

- Using the Docker CLI search on Docker Hub for an image called **mariadb**

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #3

- Using the docker client pull the image called **alpine** with the **edge** tag.
- List all images downloaded locally.

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #4

- Launch a container from the **alpine:edge** image. Get shell access to the container.
- Create a new file called **a.txt** in the **/root** directory of the container
- Exit the container without stopping it
- Check that the container is still running
- Attach to the container by running the **sh** command in the container
- Check that the file (a.txt) still exists
- Exit the container by running **exit**

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #5

- Run the Nginx web server in Docker container, in the background, and publish port 80
- Connect to the server from another machine using a browser
- Check the web server logs
- Attach to the container in which nginx is running

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #6

- Run two Nginx and two Apache containers that publish random ports
- Connect to the web servers from another machine using a browser
- Stop the containers
- Remove all stopped containers

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #7

- Run **ubuntu:latest** in a container and attach to it (get shell access)
- Install the OpenSSH server in the container (apt update && apt install ssh), add a user, and set its password
- Exit the container without stopping it (Ctrl P + Q)
- Check the IP address of the container and connect to it using SSH and the user you've just created

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #8

- Consider the previous challenge when you've installed the OpenSSH server in a Ubuntu docker container
- Commit the changes to a new image named **myubuntu** with the tag **custom**
- Start a container from this new image and check that the ssh daemon is installed and is running
- Push the image to Docker Hub

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #9

- Go to Docker Hub and search for Apache
- Go to the Dockerfile of the image with the **latest** tag
- Copy the Dockerfile locally and build a custom image from that file

Note: You'll need a file called **httpd-foreground** as well. The file is available in the GitHub repository, where the Dockerfile resides.

Download the httpd-foreground file as well and set the execution permission before building the image.

- Start a container from the image and test that it works as expected

Are you stuck? Do you want to see the solution to this challenge? Click here.

Challenge #10

- Create a volume named **webapp1**
- Inspect the volume
- Copy a few files in the volume's directory
- Start the Apache web server (httpd image) in a docker container with the volume mounted in its DirectoryRoot (/usr/local/apache2/htdocs)
- Access the files in the volume with a browser

Are you stuck? Do you want to see the solution to this challenge? Click here.