FILL IN : Lab 2

*Lovepreet Singh*

Questions from the lab

*In the lab assignment, you’ll see several questions in red boxes. Paste those questions and their respective answers below. Make sure your answer is concise and well-formatted. You may submit this as e.g. a screenshot of a filled-out cell in a copy of the Notion document (e.g. with code, so that code formatting is maintained).*

**Q1 :** Add your own comments to explain in your own words what this Dockerfile does.

**A1 :** It installs Flask, uWSGI and sets everything up.

**Q2 :** To obtain an image, we have to create a so-called build. The basis for this is a Dockerfile. With which command can you do this execute this as well.

**A2 :** docker build -t my-flask-app web

**Q3 :** Reflect on whether it is feasible to minimise the number of layers. Once you have made your assessment, take the necessary steps to reduce them.

You need to reduce the Docker Layers in your Dockerfile by removing or combining certain layers.

A3 : Uses --no-cache-dir with pip to reduce image size. Keeps the order of instructions optimal for layer caching.

**Q4 :** Can you figure out a way to replace the previous commands (docker build and docker history) with an image name instead of a container\_id ?

Try to tag your image as following: your-name/frontend. You do not have to specify any version (by default that will be latest). The advantage of Tagging would be that your image now has a specific tag it can use.

**A4 :** docker history lovepreet-singh/frontend

docker run Lovepreet-singh/frontend

**Q5 :** What does the -p and -it (can also be written as -i -t) flags do? You should search for this in the official Docker documentation.

**A5 :** -p maps port 5000 on your computer to port 5000 in the container, so you can access the app from your browser.

-it allows you to interact with the container’s terminal (it combines --interactive and --tty).

**Q6 :** In your opinion, is the build process fast? Remember you just added a small extra endpoint to the application …

**A6 :** It could be faster

**Q7 :** What did you change to the Dockerfile to make it build faster?

**A7 :** look into the Dockerfile

**Q8 :** Take a look at prestart.sh and explain what's going on in this script.

**A8 :**

**Q9 :** What is the advantage of an official image? What other types are there?

**A9 :** The advantage of using an official image is that it is maintained, tested, and regularly updated by Docker or the upstream project’s maintainers.

**Q10 :** Will this work? What's going wrong?

**A11 :**

**Q11 :** Why can't you reach the API from the front-end? How could you solve this? You don't have to implement your solution.

**A11 :**

Questions to answer for every lab

**What did you learn?**

*Fill in your three take aways that you learned during this lesson.*

* fdgh

**Givethree interesting exam questions about the contents of the lab and/or the theory**.

*Thinking about this will make sure you remember the key take-aways and important details better and longer.*

* Can
* sdf

**Check the following:**

* I have made the entire lab assignment (be careful, some labs consist out of two or more Notion documents!).
* I have answered all the questions from the lab assignment.
* I have submitted my code as a zip file and/or as a link to a *public* Git repository.
* <For labs on Azure> I have shut down any resources that are in use, in order to avoid unexpected costs.