CENTRE ESPECÍFIC
D´EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

UT 11.
DOCKER
Activities

Computer Systems
CFGS DAW

Álvaro Maceda

a.macedaarranz@edu.gva.es

2022/2023

Version:230309.1334

License

Attribution - NonCommercial - ShareAlike (by-nc-sa): No commercial use of the original work or any derivative works is permitted, distribution of which must be under a license equal to that governing the original work.

Nomenclature

Throughout this unit different symbols will be used to distinguish important elements within the content. These symbols are:

Important

Attention

Interesting

UT 11. Docker

Activities

You will need Docker installed in your machine in order to carry out these exercises. If you don't want to install Docker in your computer, remember that you can use a virtual machine and install Docker there.

Take a look often to your containers with `docker ps -a` until you can understand easily what is happening.

4. Exercise 4

Search in Docker Hub and download a Docker's official image for Java JDK, version 11, provided by IBM (ibmjava)

Create a container and test that the command `javac` exists in the container. Don't forget to remove the container after that.

Remove the image from your computer.

5. Exercise 5

Create an image for containers that runs the `sl` command when started. You will need to install the `sl` package to have that command available. It will install the command in the `/usr/games` directory.

6. Exercise 6

Part 1

Create an image based on a `python:3` container with a file named `reverse.py` in the directory `/usr/bin.` The file will have this content:

```
import sys

param = sys.argv[1]
print(param[::-1])
```

Part 2

Modify the image so the script is launched with the string `Yo, banana boy!` when a container using that image runs (it should print `!yob ananab ,oY`):

```
docker run --rm <image name>
!yob ananab ,oY
```

Part 3

Modify the image so the script is launched using a parameter provided in the command line. For example, when run with:

```
docker run --rm <image name> Kayak
```

It should print `kayaK`.

7. Exercise 7

Part 1

Create an image tagged `ubuntu-net`, based on `ubuntu:latest`, that has the following network tool packages installed: `iproute2` (for the `ip` command), `iputils-ping` (for `ping`) and `net-tools` (for `ifconfig`).

Part 2

Launch four containers: `container_a`, `container_b`, `container_c` and `container_d`, based on the previous image:

- Containers `container_a` and `container_b` should be able to communicate between them through the network, but not with `container_c` and `container_d`
- Containers `container_c` and `container_d` should be able to communicate between them but not with `container_a` and `container_b` .

Test the connections using ping with the container addresses (ping from `container_a` to `container_b`, and from `container_a` to `container_c`) For example, if container `container_b` has the IP `5.6.7.8` run `ping 5.6.7.8` from `container_a`.

Test the connections using ping with the container's names. For example, run `ping B` from `container_a`.