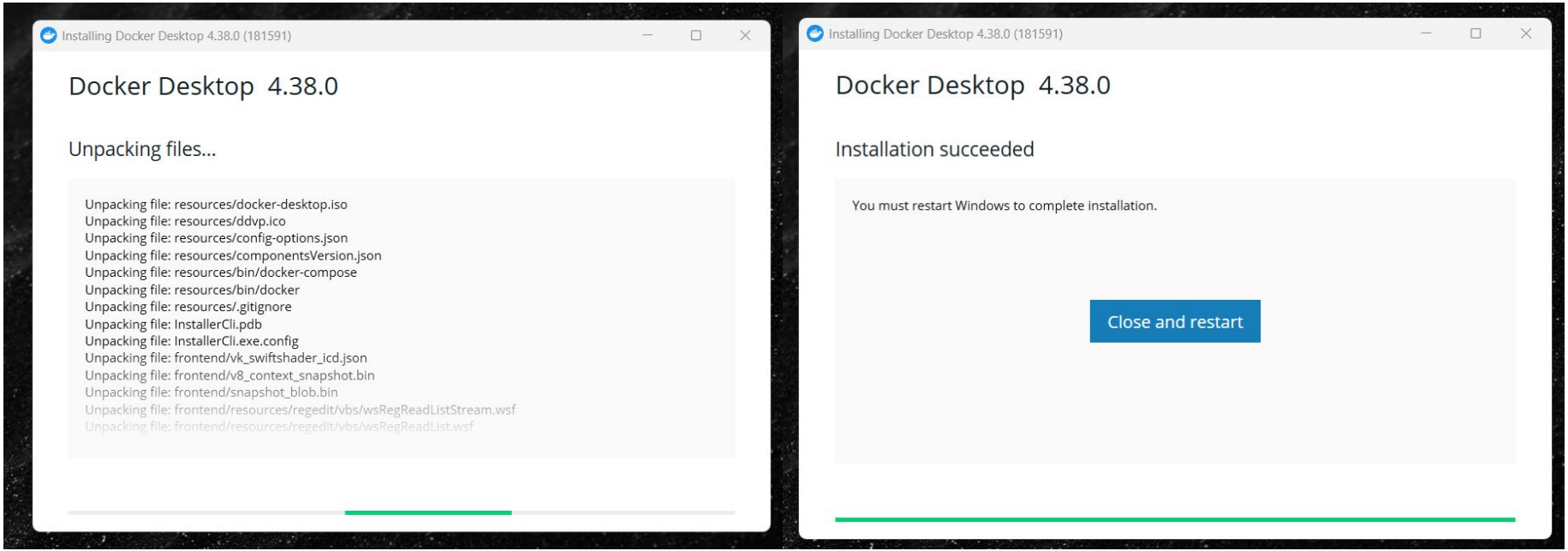
Lab 3: Getting started with Dockers

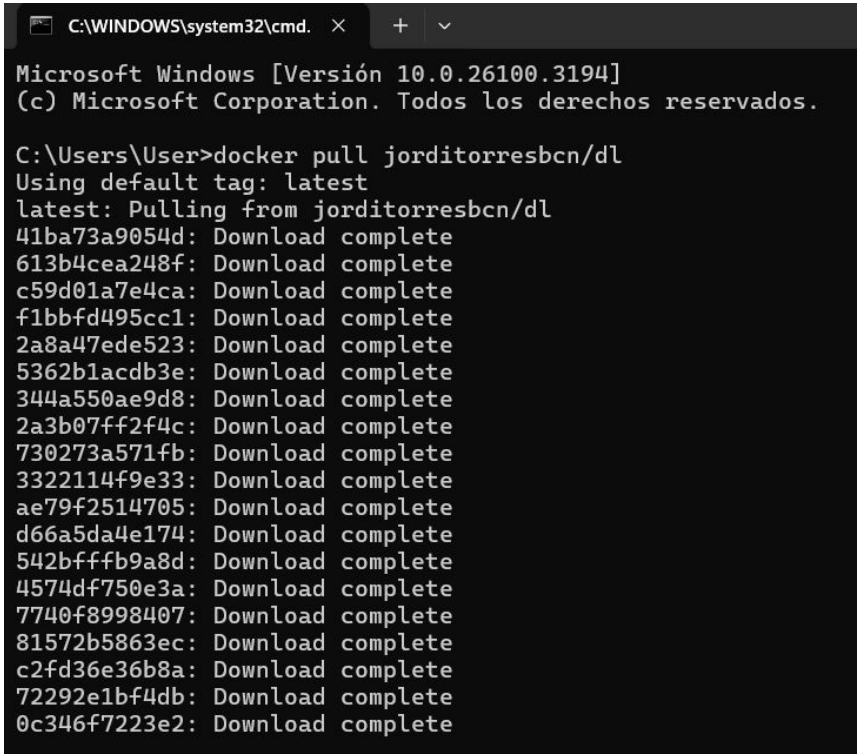
Mario Ventura

Descargar Docker for Windows



Descargar Docker image

docker pull jorditorresbcn/dl



Comprobar librerías

Comprobar librerías python

```
root@50471d4b7423:/app# pip list
DEPRECATION: The default format will switch to columns in the future. You can use --format=(legacy|columns) (or define a format=(legacy|columns) in your pip
.conf under the [list] section) to disable this warning.
bleach (1.5.0)
certifi (2018.1.18)
chardet (3.0.4)
click (6.7)
cycler (0.10.0)
decorator (4.2.1)
entrypoints (0.2.3)
enum34 (1.1.6)
Flask (0.12.2)
Flask-Cors (3.0.3)
google-api-python-client (1.6.5)
h5py (2.7.1)
html5lib (0.9999999)
httplib2 (0.10.3)
idna (2.6)
ipykernel (4.8.0)
ipython (6.2.1)
ipython-genutils (0.2.0)
ipywidgets (7.1.0)
itsdangerous (0.24)
jedi (0.11.1)
Jinja2 (2.10)
jsonschema (2.6.0)
jupyter (1.0.0)
jupyter-client (5.2.1)
jupyter-console (5.2.0)
six (1.11.0)
tensorflow (1.4.1)
tensorflow-tensorboard (0.4.0)
terminado (0.8.1)
testpath (0.3.1)
tornado (4.5.3)
traitlets (4.3.2)
tweepy (3.5.0)
uritemplate (3.0.0)
urllib3 (1.22)
wcwidth (0.1.7)
webencodings (0.5.1)
Werkzeug (0.14.1)
wheel (0.30.0)
widgetsnbextension (3.1.0)
You are using pip version 9.0.1, however version 25.0.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
root@50471d4b7423:/app#
```

Comprobar librerías

Comprobar librerías Linux

```
C:\Users\User>
C:\Users\User>docker run -it jorditorresbcn/dl /bin/bash
root@50471d4b7423:/app#
root@50471d4b7423:/app#
root@50471d4b7423:/app# cat /etc/os-release
NAME="Ubuntu"
VERSION="16.04.3 LTS (Xenial Xerus)"
ID=ubuntu
ID_LIKE=debian
PRETTY_NAME="Ubuntu 16.04.3 LTS"
VERSION_ID="16.04"
HOME_URL="http://www.ubuntu.com/"
SUPPORT_URL="http://help.ubuntu.com/"
BUG_REPORT_URL="http://bugs.launchpad.net/ubuntu/"
VERSION_CODENAME=xenial
UBUNTU_CODENAME=xenial
root@50471d4b7423:/app#
root@50471d4b7423:/app#
```

Cerrar el Docker

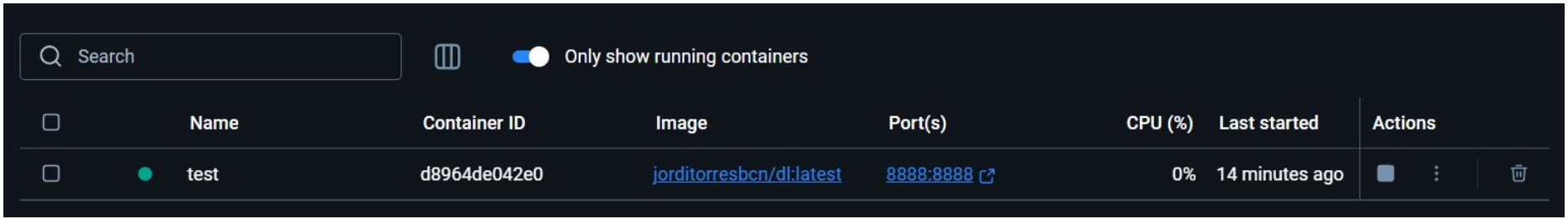
Cerrar Docker con *docker stop*

```
C:\Users\User>docker ps
CONTAINER ID   IMAGE              COMMAND       CREATED         STATUS         PORTS                NAMES
50471d4b7423   jorditorresbcn/dl   "/bin/bash"   8 minutes ago   Up 8 minutes   8888/tcp, 8954/tcp   pensive_elgamal

C:\Users\User>docker stop 50471d4b7423
50471d4b7423

C:\Users\User>docker ps
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS     NAMES

C:\Users\User>
```

Mapear puerto 8888

docker run -it -p 8888:8888 --name test jorditorresbcn/dl :latest

```
C:\Users\User>docker run -it -p 8888:8888 --name test jorditorresbcn/dl:latest
root@d8964de042e0:/app#
root@d8964de042e0:/app#
```



Iniciar Jupyter Notebook

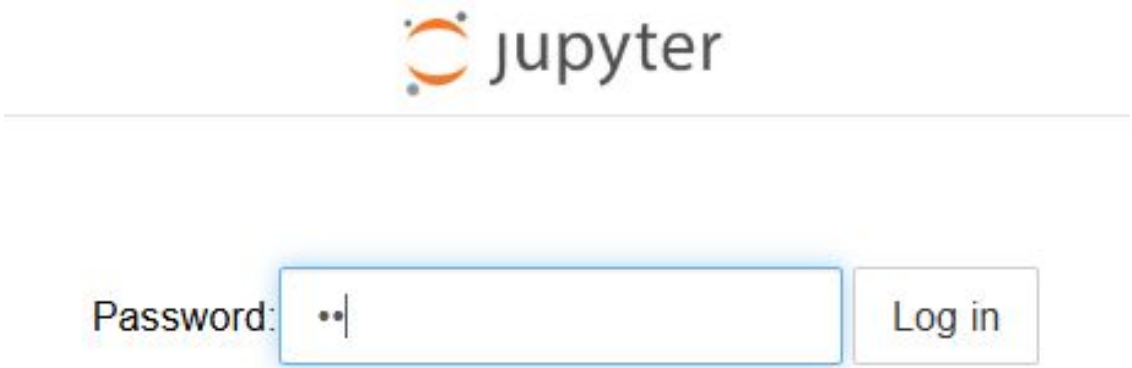
Se inicia el servidor de Jupyter Notebook dentro del docker

```
root@d8884ff3ef00:/app# jupyter notebook --ip=0.0.0.0 --allow-root
[I 11:14:09.433 NotebookApp] Serving notebooks from local directory: /app
[I 11:14:09.434 NotebookApp] 0 active kernels
[I 11:14:09.434 NotebookApp] The Jupyter Notebook is running at:
[I 11:14:09.434 NotebookApp] http://0.0.0.0:8888/
[I 11:14:09.434 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[W 11:14:09.435 NotebookApp] No web browser found: could not locate runnable browser.
```

jupyter notebook --ip=0.0.0.0 --allow-root

Entrar en Jupyter y entrenar modelo

Password: dl



Entrar en Jupyter y entrenar modelo

Notebook creado

Lab 3 - Cloud Computing

TASK 7

Printing a basic Hello World

In [12]:
```
print("Hello World")
```

```
Hello World
```

TASK 8

Training a model

In [3]:
```
import tensorflow as tf
from tensorflow import keras
import numpy as np
import matplotlib.pyplot as plt
print(tf.__version__)
```

```
1.4.1
```

In [6]:
```
x_train = x_train.astype('float32')
x_test = x_test.astype('float32')
x_train /= 255
x_test /= 255
x_train = x_train.reshape(60000, 784)
x_test = x_test.reshape(10000, 784)
```

In [7]:
```
from keras.utils import to_categorical
```

```
Using TensorFlow backend.
```

In [8]:
```
y_train = to_categorical(y_train, num_classes=10)
y_test = to_categorical(y_test, num_classes=10)
```

In [9]:
```
from keras import Sequential
from keras.layers import Dense
model = Sequential()
model.add(Dense(10, activation='sigmoid', input_shape=(784,)))
model.add(Dense(10, activation='softmax'))
model.summary()
```

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_1 (Dense)              (None, 10)                7850
_________________________________________________________________
dense_2 (Dense)              (None, 10)                110
=================================================================
Total params: 7,960
Trainable params: 7,960
Non-trainable params: 0
_________________________________________________________________
```

Crear repositorio en Docker Hub

- Ir a la página de Docker Hub.
- Crear un perfil.
- Seguir los pasos del “Step 1” para hacer pull

Step 1: Find an image in Docker Hub's library

You can search for content in Docker Hub itself, in the Docker Desktop Dashboard, or by using the CLI.

To search or browse for content on Docker Hub:

Docker Hub | Docker Desktop | CLI

1. Navigate to the Docker Hub Explore page.
   On the **Explore** page, you can browse by catalog or category, or use the search to quickly find content.
2. Under **Categories**, select **Web servers**.
   After the results are displayed, you can further filter the results using the filters on the left side of the page.
3. In the filters, select **Docker Official Image**.
   Filtering by Trusted Content ensures that you see only high-quality, secure images curated by Docker and verified publishing partners.
4. In the results, select the **nginx** image.
   Selecting the image opens the image's page where you can learn more about how to use the image. On the page, you'll also find the `docker pull` command to pull the image.

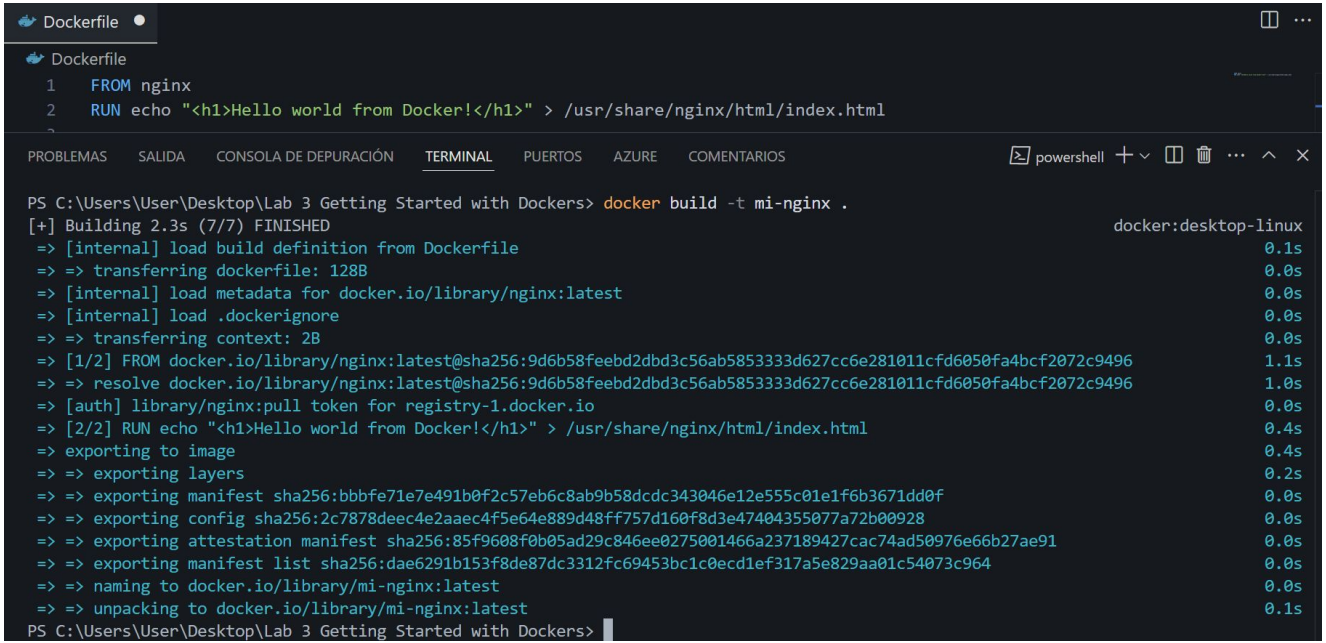
Crear repositorio en Docker Hub

Una vez hecho el ***pull***, se hace ***docker run***


```
PS C:\Users\User\Desktop\Lab 3 Getting Started with Dockers> docker run -d -p 8080:80 mi-nginx
a09ecc5f33734b33d7ae7248e8c1647bf3352525a9c296c81bafb36030bda1d2
PS C:\Users\User\Desktop\Lab 3 Getting Started with Dockers>
```

Crear repositorio en Docker Hub

Se crea el ***Dockerfile*** (ej: VS Code)

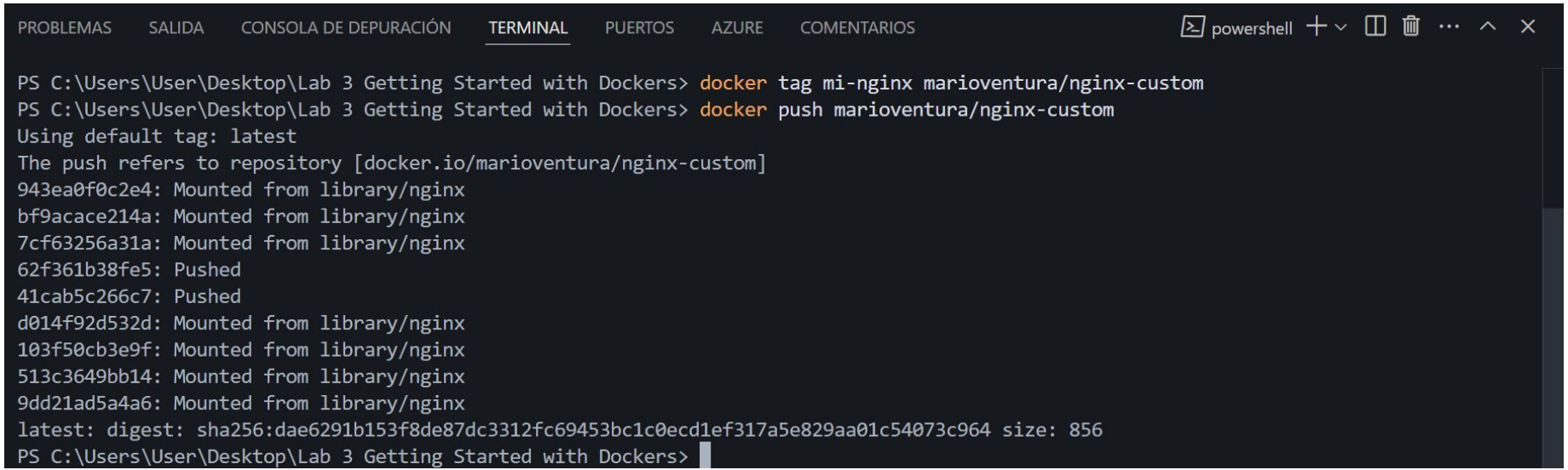


Crear repositorio en Docker Hub

Se crea el ***Dockerfile*** (ej: VS Code)

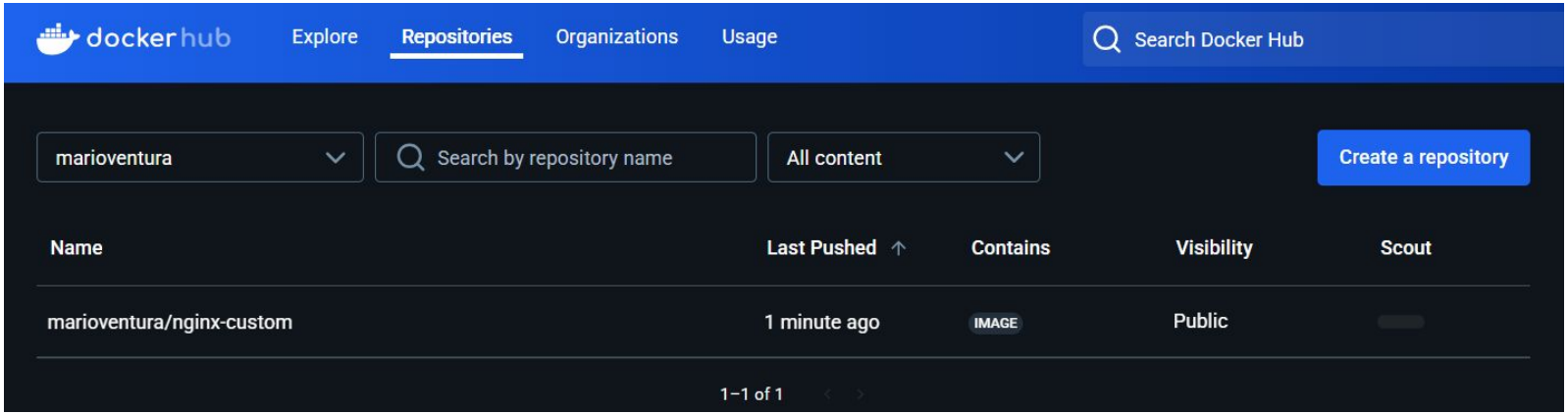
Crear repositorio en Docker Hub

Se crea un **tag** y se hace **push** del repositorio hacia Docker Hub



Crear repositorio en Docker Hub

Adicionalmente, comprobar que el repositorio existe en Docker Hub



Gracias