Paso a paso Docker

Santiago Bernal López

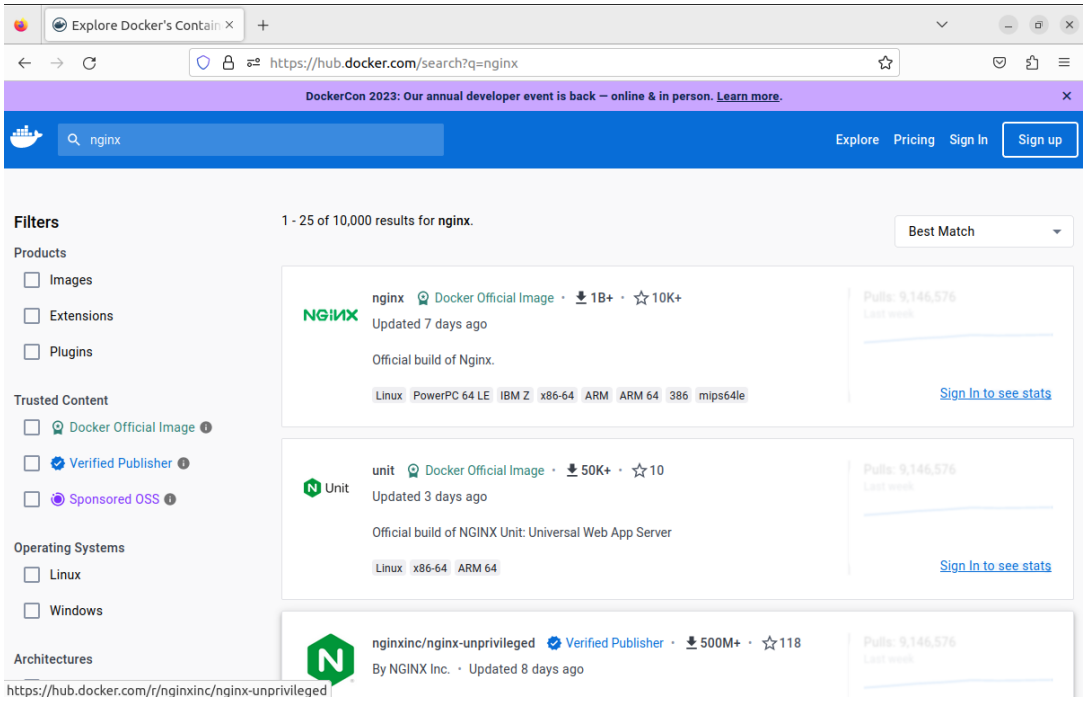
Institución Universitaria EAM

Facultad de Ingeniería

Electiva V (Computación en la nube)

2023

En primera instancia ingresamos a Docker Hub, donde debemos de buscar la imagen de nginx para descargarla.



La descargamos y montamos a docker:

```
elsanty@Santiago:~$ docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
52d2b7f179e3: Pull complete
fd9f026c6310: Pull complete
055fa98b4363: Pull complete
96576293dd29: Pull complete
a7c4092be904: Pull complete
e3b6889c8954: Pull complete
da761d9a302b: Pull complete
Digest: sha256:104c7c5c54f2685f0f46f3be607ce60da7085da3eaa5ad22d3d9f01594295e9c
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

What's Next?
  View summary of image vulnerabilities and recommendations → docker scout quick
view nginx
elsanty@Santiago:~$
```

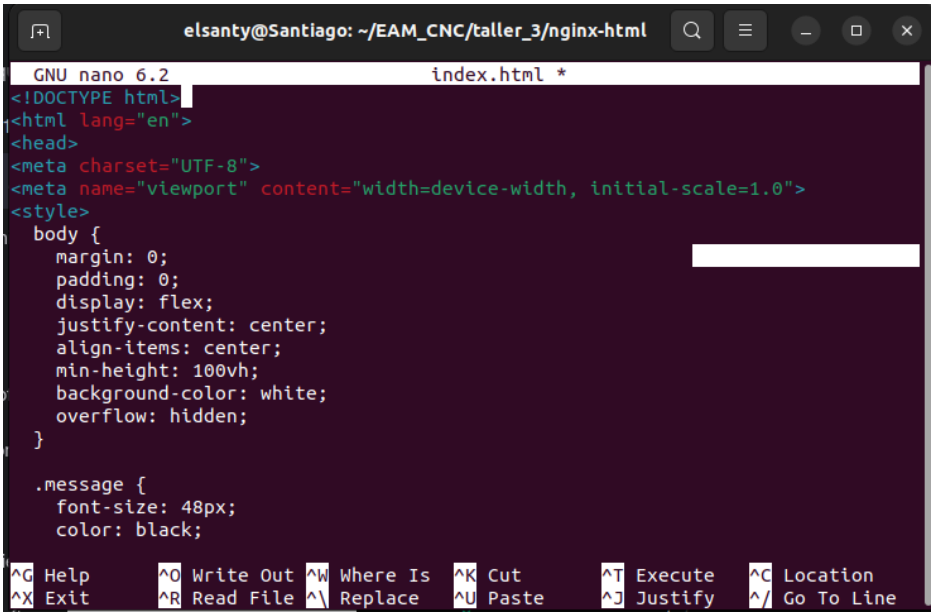
Cómo podemos ver ya quedó

```
elsanty@Santiago:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
nginx         latest    eea7b3dcba7e   2 weeks ago    187MB
hello-world   latest    9c7a54a9a43c   4 months ago   13.3kB
elsanty@Santiago:~$
```

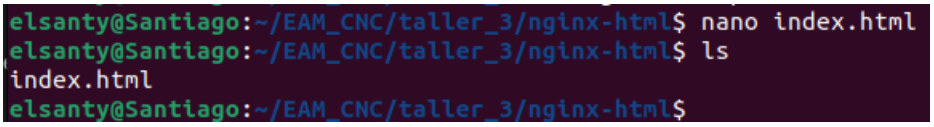
Creamos el directorio donde alojaremos el html

```
elsanty@Santiago:~$ cd EAM_CNC/
elsanty@Santiago:~/EAM_CNC$ mkdir taller_3
elsanty@Santiago:~/EAM_CNC$ cd taller_3
elsanty@Santiago:~/EAM_CNC/taller_3$ mkdir nginx-html
elsanty@Santiago:~/EAM_CNC/taller_3$ cd nginx-html/
```

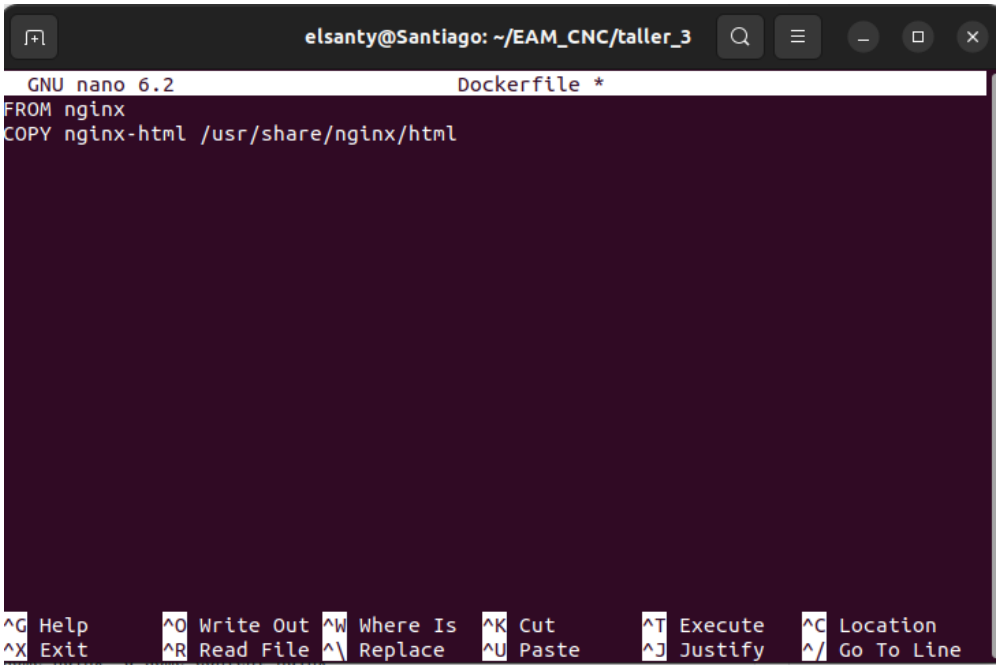
Creamos el html



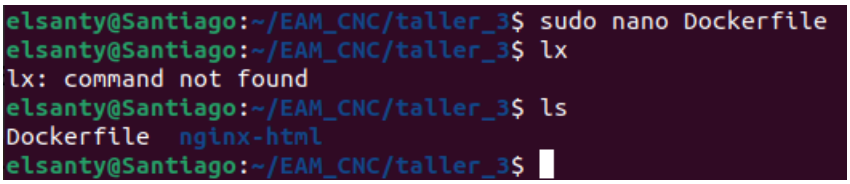
Verificamos que se haya creado



Creamos un archivo Dockerfile para poder ejecutarlo.



Verificamos que se haya creado.



Montamos a Docker con el nombre que queramos, en este caso “web”.

```
elsanty@Santiago:~/EAM_CNC/taller_3$ docker build -t web .
[+] Building 0.2s (7/7) FINISHED                                docker:default
 => [internal] load .dockerignore                                         0.0s
 => => transferring context: 2B                                           0.0s
 => [internal] load build definition from Dockerfile                      0.0s
 => => transferring dockerfile: 86B                                       0.0s
 => [internal] load metadata for docker.io/library/nginx:latest           0.0s
 => [internal] load build context                                         0.0s
 => => transferring context: 757B                                         0.0s
 => [1/2] FROM docker.io/library/nginx                                    0.0s
 => [2/2] COPY nginx-html /usr/share/nginx/html                           0.0s
 => exporting to image                                                    0.0s
 => => exporting layers                                                   0.0s
 => => writing image sha256:0a0c5ff815da5e3c3ffcf2bb092f9e2a1383f2f5cc4db  0.0s
 => => naming to docker.io/library/web                                    0.0s

What's Next?
  View summary of image vulnerabilities and recommendations → docker scout quick
view
elsanty@Santiago:~/EAM_CNC/taller_3$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED          SIZE
web           latest    0a0c5ff815da   16 seconds ago   187MB
nginx         latest    eea7b3dcba7e   2 weeks ago      187MB
hello-world   latest    9c7a54a9a43c   4 months ago     13.3kB
elsanty@Santiago:~/EAM_CNC/taller_3$
```

Ejecutamos “web” en el puerto 8080 y verificamos.

```
elsanty@Santiago:~/EAM_CNC/taller_3$ docker run -d -p 8080:80 web
d1902bf63b6de6932374a8ec79a9825ed0fce474ce7e82adc69219410a8242a7
elsanty@Santiago:~/EAM_CNC/taller_3$ docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED         STATUS
 PORTS                                  NAMES
d1902bf63b6d   web       "/docker-entrypoint.…"   9 seconds ago   Up 7 seconds
 0.0.0.0:8080->80/tcp, :::8080->80/tcp   busy_antonelli
elsanty@Santiago:~/EAM_CNC/taller_3$
```

Accedemos a “localhost:8080” y podemos ver el Html en la web.