

Tarea de docker 2



Índice:

Descargar imagen de Ubuntu	2
Ejecutar un contenedor de la imagen de Ubuntu	2
Contenedores	3

Descargar imagen de Ubuntu

```
joseito@kyogre:~$ docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
aece8493d397: Pull complete
Digest: sha256:2b7412e6465c3c7fc5bb21d3e6f1917c167358449fecac8176c6e496e5c1f05f
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest

What's Next?
  1. Sign in to your Docker account -> docker login
  2. View a summary of image vulnerabilities and recommendations -> docker scout quickview ubuntu
joseito@kyogre:~$ docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
mongo         latest    ee3b4d1239f1   5 weeks ago    748MB
ubuntu        latest    e4c58958181a   6 weeks ago    77.8MB
joseito@kyogre:~$
```

Comando:

```
docker pull ubuntu
```

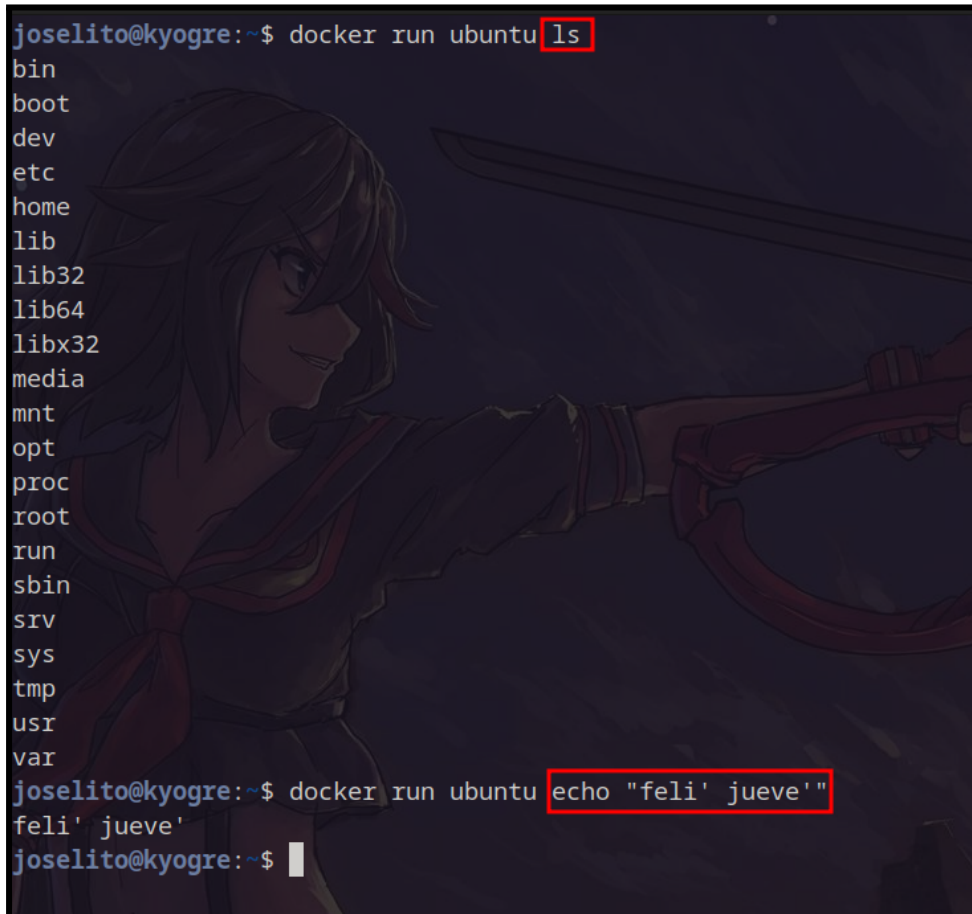
Ejecutar un contenedor de la imagen de Ubuntu

La función de Ubuntu en docker es ejecutar comandos de bash.

Así que ejecutaremos el comando:

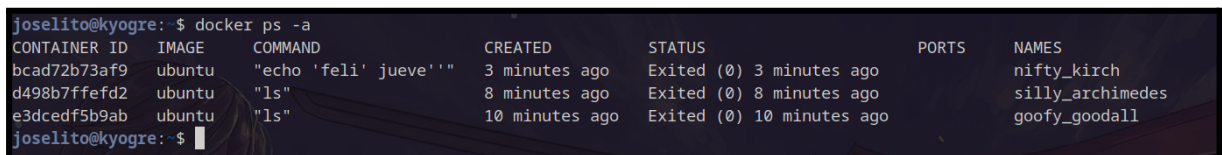
`docker run ubuntu [COMANDO DE BASH]`

ejemplo:



```
joselito@kyogre:~$ docker run ubuntu ls
bin
boot
dev
etc
home
lib
lib32
lib64
libx32
media
mnt
opt
proc
root
run
sbin
srv
sys
tmp
usr
var
joselito@kyogre:~$ docker run ubuntu echo "feli' jueve'"
feli' jueve'
joselito@kyogre:~$
```

Esto luego se ve reflejado en docker como contenedores ejecutados, procesos como tal:



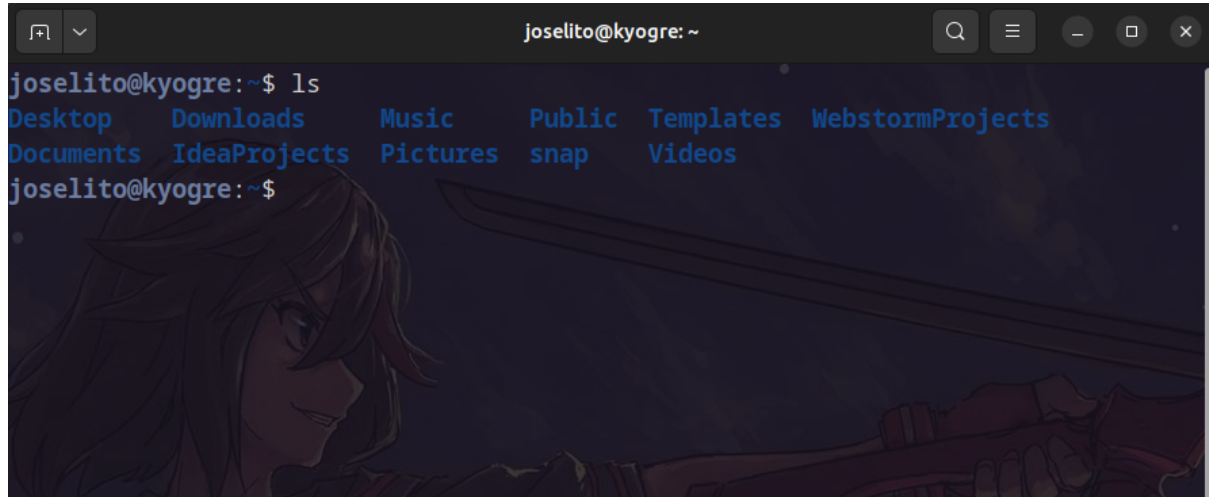
```
joselito@kyogre:~$ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
bcad72b73af9	ubuntu	"echo 'feli' jueve'"	3 minutes ago	Exited (0) 3 minutes ago		nifty_kirch
d498b7ffefd2	ubuntu	"ls"	8 minutes ago	Exited (0) 8 minutes ago		silly_archimedes
e3dcedf5b9ab	ubuntu	"ls"	10 minutes ago	Exited (0) 10 minutes ago		goofy_goodall

```
joselito@kyogre:~$
```

También un pequeño detalle a la hora de ejecutar “ls”, no me mostro como tal mi carpeta home local de mi PC, si no que mostró la de viene derivada la imagen de ubuntu que descargamos:

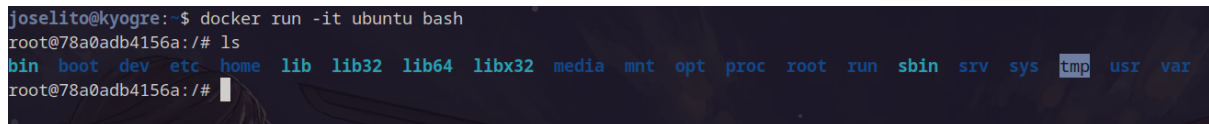
Mi carpeta home local:



```

joselito@kyogre:~$ ls
Desktop  Downloads  Music      Public  Templates  WebstormProjects
Documents  IdeaProjects  Pictures  snap    Videos
joselito@kyogre:~$
  
```

Contenido derivado de la imagen de ubuntu:



```

joselito@kyogre:~$ docker run -it ubuntu bash
root@78a0adb4156a:/# ls
bin  boot  dev  etc  home  lib  lib32  lib64  libx32  media  mnt  opt  proc  root  run  sbin  srv  sys  tmp  usr  var
root@78a0adb4156a:/#
  
```

Para verlo, como se ve en la imagen ejecute un comando para poder interactuar con el contenido de la imagen de ubuntu.

(Como tal, cree un nuevo proceso para visualizarlo a través de *docker run*)

Contenedores

En docker los procesos los conocemos como contenedores, que es todo lo que hemos ejecutado a través de una imagen que vemos con el comando *docker ps -a*.

A través de un comando llamado “*docker exec*”, podemos ejecutar estos contenedores.

Ejemplo:

Anteriormente creamos un contenedor para ver el contenido de la imagen de docker por dentro, pues ahora me salgo, pero quiero volver a entrar:

```
jose1ito@kyogre:~$ docker exec -it 78a0adb4156a bash
root@78a0adb4156a:/# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@78a0adb4156a:/# exit
exit
jose1ito@kyogre:~$
```

Pues ahora ejecuto el comando:

`docker exec -it [ID DEL CONTENEDOR] bash`

Y se volverá a ejecutar

```
jose1ito@kyogre:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
78a0adb4156a   ubuntu   "bash"                  16 minutes ago Up 9 minutes           friendly_mahavira
bcad72b73af9   ubuntu   "echo 'feli' jueve'"    22 minutes ago Exited (0) 8 minutes ago nifty_kirch
d498b7ffefd2   ubuntu   "ls"                    28 minutes ago Exited (0) 9 minutes ago silly_archimedes
e3dcedf5b9ab   ubuntu   "ls"                    30 minutes ago Exited (0) 30 minutes ago goofy_goodall
jose1ito@kyogre:~$ docker exec -it 78a0adb4156a bash
root@78a0adb4156a:/# ls
bin boot dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@78a0adb4156a:/#
```

Ahora bien, no os “funcionará” siempre, por ejemplo, el *ls* o el *echo* que hice anteriormente, no os darán el mismo resultado.

Si intentamos hacer un *docker exec* por ejemplo de el echo, nos dará un error

```
jose1ito@kyogre:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
78a0adb4156a   ubuntu   "bash"                  20 minutes ago Up 14 minutes           friendly_mahavira
bcad72b73af9   ubuntu   "echo 'feli' jueve'"    27 minutes ago Exited (0) 12 minutes ago nifty_kirch
d498b7ffefd2   ubuntu   "ls"                    33 minutes ago Exited (0) 14 minutes ago silly_archimedes
e3dcedf5b9ab   ubuntu   "ls"                    35 minutes ago Exited (0) 35 minutes ago goofy_goodall
jose1ito@kyogre:~$ docker exec -it bcad72b73af9 bash
Error response from daemon: Container bcad72b73af9e343f2d18df95ce69a0462fc2fcae79ef7bff80dc3a316b55914 is not running
jose1ito@kyogre:~$
```



Fecha: 18 / 11 / 2023

Esto es más que nada porque los comandos *ls* o *echo*, son comandos que se ejecutan y se finalizan al proceso, entonces al intentar ejecutarlos, dará error, porque su ejecución finalizado en el momento en el que los ejecute.

En cambio, un *bash* es proceso que se queda abierto hasta que tú te deshaces de él.

Eso si, si ejecuto un *docker start* de alguno de los procesos de *ls* o *echo* si funcionará, pero nosotros no veremos el *ls* o el *echo* porque se ejecutarán dentro de lo que es, el contenedor:

```
joselito@kyogre:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS   NAMES
78a0adb4156a   ubuntu   "bash"                  26 minutes ago Up 19 minutes                friendly_mahavira
bcad72b73af9   ubuntu   "echo 'feli' jueve'"    32 minutes ago Exited (0) 18 minutes ago    nifty_kirch
d498b7ffefd2   ubuntu   "ls"                   38 minutes ago Exited (0) 20 minutes ago    silly_archimedes
e3dcedf5b9ab   ubuntu   "ls"                   40 minutes ago Exited (0) 40 minutes ago    goofy_goodall
joselito@kyogre:~$ docker start bcad72b73af9
bcad72b73af9
joselito@kyogre:~$
```