

1- Lee el documento contenedores que hay en Moodle y responde razonadamente las siguientes cuestiones en este mismo documento y entrégalo por correo electrónico

- ¿Qué es un contenedor?

Un contenedor de docker es una instancia de la imagen que hemos creado o descargado que se ejecutan de forma aislada. Alojan todas las dependencias que la aplicación necesite para ser ejecutada, es decir, código, librerías, entorno de ejecución y configuración.

- ¿Qué es una imagen de docker?

Una imagen de docker es una instancia de un contenedor, es decir, una plantilla que se usa para construir contenedores Docker y ejecutarlos a partir de una imagen previamente compilada.

- ¿Qué relación/diferencia hay entre un contenedor y una imagen?

Una instancia de una imagen se llama contenedor.

Si se inicia una imagen de docker, se tiene un contenedor en ejecución de dicha imagen.

Puede haber muchos contenedores en ejecución de la misma imagen.

- ¿Qué es una máquina virtual? ¿Que diferencias hay entre una máquina virtual y un contenedor? ¿Qué ventajas / inconvenientes presentan ambas soluciones?

Una máquina virtual (VM) es un sistema operativo completo funcionando de manera aislada sobre otro sistema operativo completo.

Las diferencias entre una VM y un contenedor son:

- Por lo general, los contenedores se miden en megabytes. El elemento más grande que empaquetan es una aplicación y todos los archivos necesarios para su ejecución. También suelen utilizarse para empaquetar funciones individuales que realizan tareas específicas (conocidas como microservicios). La naturaleza ligera de los contenedores y su sistema operativo (SO) compartido permiten trasladarlos entre los distintos entornos con mucha facilidad.
- Por lo general, las máquinas virtuales se miden en gigabytes. Suelen tener su propio sistema operativo, lo cual les permite realizar varias funciones con uso intensivo de los recursos al mismo tiempo. Las máquinas virtuales cuentan con una mayor cantidad de recursos disponibles, lo cual les permite extraer, dividir, duplicar y simular sistemas operativos, escritorios, bases de datos, conexiones de red y servidores completos.

Por tanto, para múltiples copias de la misma aplicación usaríamos un contenedor, mientras que para más flexibilidad o para ejecutar múltiples aplicaciones, usaríamos la máquina virtual.

- ¿Busca información e indica qué es docker compose?

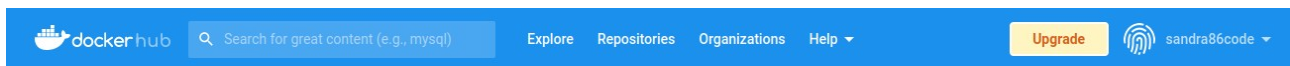
Docker Compose es una herramienta dedicada a la orquestación local de dockers, es decir, se usa para definir y ejecutar aplicaciones Docker de varios contenedores de forma fácil y rápida.

Esta definición y orquestación se lleva a cabo de forma local al interior de los containers, quienes, además, se encontrarán unidos a través de una red de Docker.





2- Comprueba si tienes instalado docker (haz captura de todo lo que hagas y lo envías por e-mail) y en caso de no tenerlo, instálalo.

```
sandra@sandra-UX410UAK: ~  
sandra@sandra-UX410UAK:~$ sudo docker --version  
[sudo] password for sandra:  
Docker version 20.10.14, build a224086349  
sandra@sandra-UX410UAK:~$
```

3- Créate una cuenta de usuario en la web oficial de docker.



4- Busca en el repositorio los 4 primeros contenedores que aparezcan en el listado y anótalos aquí.

| | |
|---|--|
|  | ubuntu DOCKER OFFICIAL IMAGE Updated 21 days ago Ubuntu is a Debian-based Linux operating system based on free software. Linux ARM ARM 64 PowerPC 64 LE riscv64 IBM Z 386 x86-64 1B+ Downloads 10K+ Stars |
|  | alpine DOCKER OFFICIAL IMAGE Updated 2 months ago A minimal Docker image based on Alpine Linux with a complete package index and only 5 MB in size! Linux riscv64 x86-64 ARM ARM 64 386 PowerPC 64 LE IBM Z 1B+ Downloads 9.3K Stars |
|  | redis DOCKER OFFICIAL IMAGE Updated 5 days ago Redis is an open source key-value store that functions as a data structure server. Linux Windows IBM Z x86-64 ARM ARM 64 386 mips64le PowerPC 64 LE 1B+ Downloads 10K+ Stars |
|  | busybox DOCKER OFFICIAL IMAGE Updated 12 days ago Busybox base image. Linux 386 mips64le PowerPC 64 LE riscv64 IBM Z x86-64 ARM ARM 64 1B+ Downloads 2.8K Stars |

5- Descarga los dos primeros contenedores, y carga el segundo. Anota la ID de este.

Descarga:

```
sandra@sandra-UX410UAK:~$ sudo docker pull ubuntu
Using default tag: latest
latest: Pulling from library/ubuntu
2b55860d4c66: Pull complete
Digest: sha256:20fa2d7bb4de7723f542be5923b06c4d704370f0390e4ae9e1c833c8785644c1
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
sandra@sandra-UX410UAK:~$ sudo docker pull alpine
Using default tag: latest
latest: Pulling from library/alpine
213ec9aee27d: Pull complete
Digest: sha256:bc41182d7ef5ffc53a40b044e725193bc10142a1243f395ee852a8d9730fc2ad
Status: Downloaded newer image for alpine:latest
docker.io/library/alpine:latest
sandra@sandra-UX410UAK:~$
```

Carga de Alpine:

sudo docker run alpine

ID: 9e04873d6f46

6- Obtén un listado de los contenedores que hay descargados y para el que esté arrancado

```
sandra@sandra-UX410UAK:~$ sudo docker ps -a
```

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|--------------|--------------|--------------------------|--------------------|-------------------------------|-------|---------------------|
| 9e04873d6f46 | alpine | "/bin/sh" | About a minute ago | Exited (0) About a minute ago | | strange_ardinghelli |
| b2005e6d3a9d | alpine | "/bin/sh" | 2 minutes ago | Exited (0) 2 minutes ago | | angry_robinson |
| 87d658a6e221 | alpine | "/bin/sh" | 3 minutes ago | Exited (0) 3 minutes ago | | busy_shaw |
| 3e8238d7e2b0 | mysql:5.7.22 | "docker-entrypoint.s..." | About an hour ago | Exited (0) 36 minutes ago | | libreriaJacaranda |

7- Borra el primer contenedor, y arranca el 3 y 4 que pusiste en la lista de la cuestión 4.

Borrado:

```
sandra@sandra-UX410UAK:~$ sudo docker rm 9e04873d6f46
9e04873d6f46
sandra@sandra-UX410UAK:~$ sudo docker ps -a
```

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|--------------|--------------|--------------------------|-------------------|---------------------------|-------|-------------------|
| b2005e6d3a9d | alpine | "/bin/sh" | 4 minutes ago | Exited (0) 3 minutes ago | | angry_robinson |
| 87d658a6e221 | alpine | "/bin/sh" | 4 minutes ago | Exited (0) 4 minutes ago | | busy_shaw |
| 3e8238d7e2b0 | mysql:5.7.22 | "docker-entrypoint.s..." | About an hour ago | Exited (0) 38 minutes ago | | libreriaJacaranda |

```
sandra@sandra-UX410UAK:~$ sudo docker rm b2005e6d3a9d
b2005e6d3a9d
sandra@sandra-UX410UAK:~$ sudo docker rm 87d658a6e221
87d658a6e221
sandra@sandra-UX410UAK:~$ sudo docker ps -a
```

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|--------------|--------------|--------------------------|-------------------|---------------------------|-------|-------------------|
| 3e8238d7e2b0 | mysql:5.7.22 | "docker-entrypoint.s..." | About an hour ago | Exited (0) 39 minutes ago | | libreriaJacaranda |

```
sandra@sandra-UX410UAK:~$
```

Arrancado de imagen redis (no arranca por memoria):

```
sandra@sandra-UX410UAK:~$ sudo docker run redis
1:C 27 Sep 2022 08:49:15.403 # o000o000o000o Redis is starting o000o000o000o
1:C 27 Sep 2022 08:49:15.403 # Redis version=7.0.5, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 27 Sep 2022 08:49:15.403 # Warning: no config file specified, using the default config. In order to specify a config
file use redis-server /path/to/redis.conf
1:M 27 Sep 2022 08:49:15.403 * Increased maximum number of open files to 10032 (it was originally set to 1024).
1:M 27 Sep 2022 08:49:15.403 * monotonic clock: POSIX clock_gettime
1:M 27 Sep 2022 08:49:15.404 * Running mode=standalone, port=6379.
1:M 27 Sep 2022 08:49:15.404 # Server initialized
1:M 27 Sep 2022 08:49:15.404 # WARNING overcommit_memory is set to 0! Background save may fail under low memory condition
. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.ove
commit_memory=1' for this to take effect.
1:M 27 Sep 2022 08:49:15.404 * Ready to accept connections
```

```
sandra@sandra-UX410UAK:~$ sudo docker run busybox
sandra@sandra-UX410UAK:~$ sudo docker ps -a
```

| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
|--------------|--------------|--------------------------|-------------------|---------------------------|-------|-------------------|
| 0bd29c839193 | busybox | "sh" | 8 seconds ago | Exited (0) 7 seconds ago | | nifty_villani |
| ccb808a298f8 | redis | "docker-entrypoint.s..." | 4 minutes ago | Exited (0) 19 seconds ago | | jovial_pascal |
| 3e8238d7e2b0 | mysql:5.7.22 | "docker-entrypoint.s..." | About an hour ago | Exited (0) 46 minutes ago | | libreriaJacaranda |

```
sandra@sandra-UX410UAK:~$
```

8- Muestra las salidas de la imagen de alguno de los contenedores que haya activos.

9- Busca en la web la sintaxis para crear un contenedor / para arrancar un contenedor

10 – Busca en la web la sintaxis para trabajar con una imagen concreta.

Principales instrucciones con Docker

docker run imagen..... para arrancar una imagen (si no existe, la descarga también)
docker run 'imagen:versión'.....para arancar una imagen con una versión concreta.
docker pull imagen.....descarga la imagen pero no la arranca (por defecto la mas reciente)
docker images.....muestra las imágenes que tenemos descargadas
docker images | head.....muestra las primeras lineas de las imágenes descargadas.
docker psmuestra las imágenes que están arrancadas, activas
docker ps -amuestra las últimas imágenes utilizadas
control C.....frena, para una imagen arrancada
docker start ID..... si queremos reiniciar-recuperar el contenedor que hemos paralizado
docker log
docker log -f ID.....muestra las salidas de la imagen
docker exec ID.....ejecuta un comando dentro de un contenedor que está arrancado
docker exec -it ID sh.....(-i crea una sesión interactiva, -t emula una terminal... sh es una shell)
docker stop ID..... Para un contenedor
docker run -d imagen para arrancar una imagen “background”
docker rm ID.....borra un contenedor