

Vicente Villarreal Hinojo

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?

Empaquetado portable normalizado para sus aplicaciones.

Del mismo modo que en el sector del transporte se usan contenedores físicos para aislar diferentes cargas (por ejemplo, para el transporte en buques y en trenes), las tecnologías de desarrollo de software usan cada vez más un método denominado contenerización.

Un paquete de software estándar (conocido como “contenedor”) agrupa el código de una aplicación con las bibliotecas y los archivos de configuración asociados, junto con las dependencias necesarias para que la aplicación se ejecute. Esto permite a los desarrolladores y profesionales de TI implementar aplicaciones sin problemas en todos los entornos.

- ¿Qué es una imagen de docker?

Una imagen en Docker es un archivo o file que se encuentra compuesto de diversas capas y que se utiliza con el objetivo de ejecutar un código dentro de un contenedor de Docker. Estas imágenes contienen todo el sistema de ficheros inicial en los que se va a basar el container para su funcionamiento, así como su punto de entrada o **entrypoint**.

Este punto de entrada se refiere a la aplicación o comando que deberá ejecutarse una vez que el usuario lance un contenedor que esté asociado a esa imagen en Docker.

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

Las imágenes de Docker se utilizan para empaquetar aplicaciones y entornos de servidor preconfigurados. Los contenedores utilizan la información del servidor y el sistema de archivos proporcionado por la imagen para funcionar. Las imágenes se pueden compartir en Docker Hub.

- ¿Qué es una máquina virtual? ¿Que diferencias hay entre una máquina virtual y un contenedor?

Una máquina virtual es un software que permite emular el funcionamiento de un ordenador dentro de otro ordenador gracias a un proceso de encapsulamiento que aísla a ambos.

Una **máquina virtual** requiere combinar su propio sistema operativo y el hipervisor con el sistema operativo del host. En cambio, un **contenedor** no necesita un sistema operativo para funcionar y aprovechan el kernel del sistema operativo del host para ejecutarse.

¿Que ventajas / inconvenientes presentan ambas soluciones?

Ventajas de las máquinas virtuales:

- **No requiere inversiones en hardware:** a diferencia de una máquina física, la configuración de las máquinas virtuales no necesita componentes físicos y, por tanto, evita que las organizaciones gasten el dinero en el mantenimiento y la renovación de estos dispositivos.
- **Proporciona mayor accesibilidad**
- **Aporta elasticidad, flexibilidad y escalabilidad**
- **Ofrece mayor seguridad**

Inconvenientes de las máquinas virtuales:

- La máquina virtual no es tan eficiente como una real cuando se accede al hardware.
- Redundancia
- Gastos generales : las máquinas virtuales consumen más recursos que un contenedor.
- Velocidad

Ventajas de contenedores:

- Se obtiene mayor modularidad.
- Solo se tiene que programar la aplicación una sola vez.
- Se obtiene una mayor consistencia entre los entornos de prueba y los entornos de producción.
- Los contenedores son gratuitos y de código abierto.
- Los contenedores son muy, muy, muy consistentes.

Desventajas de los contenedores:

- Se requiere mínimo la versión de Kernel 3.8.

- Algunas versiones de Docker dan error debido a que se encuentran en constante desarrollo.
- Solo soporta a sistemas operativos Linux de arquitectura de 64 bits.
- Para Windows aun se encuentra en fase de desarrollo.

¿Busca información e indica qué es docker compose?

Docker Compose es una herramienta de la plataforma dedicada a la orquestación local de dockers, es decir, se utiliza con el objetivo de 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.

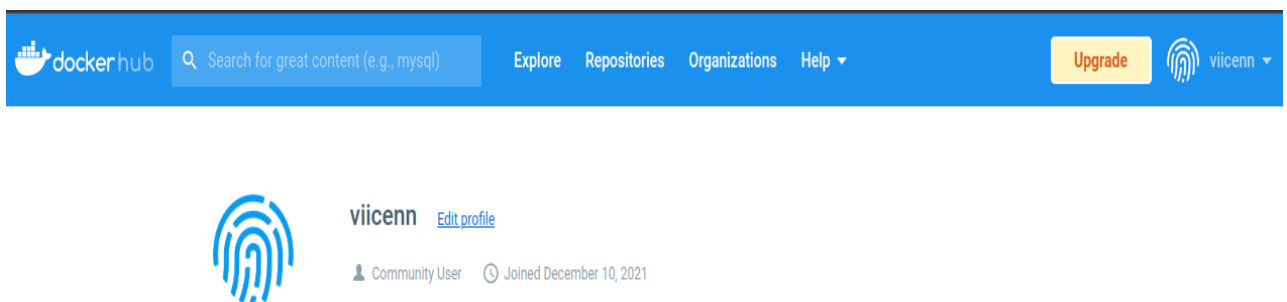
2- Comprueba si tienes instalado docker (haz captura de todo lo que hagas y lo envías por e-mail) y en caso de no tenerlo, instálalo.

```

estudiante@DAW1: ~/CARS
estudiante@DAW1:~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS
2843d55b306c   mysql:5.7.22   "docker-entrypoint.s..." 25 hours ago   Up 2 hours
0.0.0.0:3306->3306/tcp, :::3306->3306/tcp   coches
2817b09367fb   hello-world    "/hello"                 3 months ago   Exited (0)
3 months ago   wizardly_snyder
estudiante@DAW1:~/CARS$

```

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.

Alpine, Ubuntu, Python, busybox

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

```
estudiante@DAW1: ~/CARS
wizardly_snyder
estudiante@DAW1:~/CARS$ 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
estudiante@DAW1:~/CARS$ 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
estudiante@DAW1:~/CARS$ docker run alpine
estudiante@DAW1:~/CARS$ docker run ubuntu
```

```
estudiante@DAW1:~/CARS$ docker run ubuntu
estudiante@DAW1:~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS
NAME           NAMES
aa8c0afadbdc   ubuntu        "bash"                  4 seconds ago   Exited (0) 3 seconds ago
hopeful_shannon
a99ab46740dd   alpine        "/bin/sh"               9 seconds ago   Exited (0) 8 seconds ago
elastic_dhawan
2843d55b306c   mysql:5.7.22  "docker-entrypoint.s..."  5 hours ago    Up 2 hours      0.0.0.0:3306->3306/tcp, :::3306->3306/tcp
coches
2817b09367fb   hello-world   "/hello"                3 months ago    Exited (0) 3 months ago
wizardly_snyder
estudiante@DAW1:~/CARS$
```

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

```
estudiante@DAW1:~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS          NAMES
aa8c0afadbdc   ubuntu        "bash"                  2 minutes ago   Exited (0) 2 minutes ago             hope
ful_shannon
a99ab46740dd   alpine        "/bin/sh"               2 minutes ago   Exited (0) 2 minutes ago             elas
tic_dhawan
2843d55b306c   mysql:5.7.22  "docker-entrypoint.s..." 5 hours ago     Exited (0) 7 seconds ago             coch
es
2817b09367fb   hello-world   "/hello"                3 months ago    Exited (0) 3 months ago             wiza
rdly_snyder
estudiante@DAW1:~/CARS$
```

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

```
estudiante@DAW1:~/CARS$ docker rm aa8c0afadbdc
aa8c0afadbdc
estudiante@DAW1:~/CARS$ docker rm a99ab46740dd
a99ab46740dd
estudiante@DAW1:~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  NAMES          CREATED
STATUS          PORTS          NAMES          CREATED
2843d55b306c   mysql:5.7.22  "docker-entrypoint.s..." 25 hours ag
o Exited (0) 3 minutes ago             coches
2817b09367fb   hello-world   "/hello"                3 months ag
o Exited (0) 3 months ago             wizardly_snyder
estudiante@DAW1:~/CARS$
```

Aquí borramos los dos contenedores el Alpine y el Ubuntu

```

estudiante@DAW1:~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS          NAMES
faf95a5d1bf3   redis         "docker-entrypoint.s..." 43 seconds ago   Up 41 seconds   6379/tcp       infante
2843d55b306c   mysql:5.7.22  "docker-entrypoint.s..." 25 hours ago     Exited (0) 6 minutes ago                  coches
2817b09367fb   hello-world   "/hello"                 3 months ago     Exited (0) 3 months ago                  wizardly_snyder

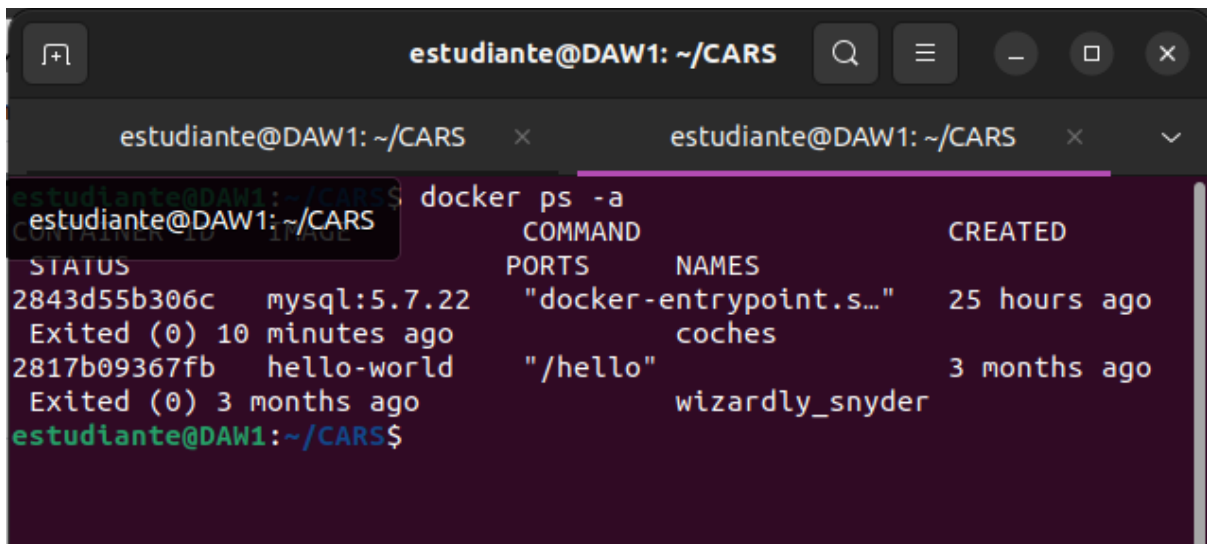
estudiante@DAW1:~/CARS$ docker run python
Unable to find image 'python:latest' locally
latest: Pulling from library/python
23858da423a6: Pulling fs layer
326f452ade5c: Downloading 404.8kB/5.163MB

23858da423a6: Downloading 540kB/55.03MB
326f452ade5c: Downloading 732.5kB/5.163MB
326f452ade5c

```

Aquí podemos ver que he instalado con el docker run el contenedor de redis y más abajo he empezado a lanzar el de python.

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



```

estudiante@DAW1: ~/CARS
estudiante@DAW1: ~/CARS$ docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS          NAMES
2843d55b306c   mysql:5.7.22  "docker-entrypoint.s..." 25 hours ago     Exited (0) 10 minutes ago                  coches
2817b09367fb   hello-world   "/hello"                 3 months ago     Exited (0) 3 months ago                  wizardly_snyder
estudiante@DAW1:~/CARS$

```

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

```

docker run -d --name nombreContenedor //
docker run -d --name nombreContenedor -e MYSQL_ROOT_PASSWORD=passwordRoot
-p 3306:3306 mysql:5.7.22

```

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

`docker exec -it nombreContenedor /bin/bash`