

1. ¿Qué es Docker Compose? ¿Qué utilidades tiene? Instálalo y comprueba que está instalado.

Es una herramienta para desplegar grupos de contenedores que forman parte de una misma aplicación o un mismo entorno.

Docker Compose nos permite poner en funcionamiento todos los contenedores necesarios, en el orden adecuado, de una sola vez y asegurando la comunicación entre ellos.

Para instalarlo usamos el siguiente comando:

```
sudo curl -L
```

```
"https://github.com/docker/compose/releases/download/v2.11.2/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

Concedemos los permisos necesarios:

```
sudo chmod +x /usr/local/bin/docker-compose
```

Comprobamos que está instalado:

```
docker-compose --version
```

2. Haz un glosario de comandos de Docker Compose.

docker-compose --version	obtener la versión de docker compose
docker-compose up - -d	crear los contenedores descritos en el docker-compose.yml - crear en modo detach
docker-compose stop	detener contenedores previamente lanzados
docker-compose run	iniciar los contenedores descritos que estén parados
docker-compose pause	pausar contenedores lanzados
docker-compose unpause	reanudar contenedores pausados
docker-compose restart	reiniciar contenedores lanzados
docker-compose down - -v	parar y borrar (redes incluidas) contenedores creados en docker-compose up - borra también los volúmenes
docker-compose logs x	muestra los logs del servicio x
docker-compose exec x y	ejecuta orden y en el contenedor x
docker-compose build	ejecuta el proceso de construcción de una imagen que va a ser usado en el docker-compose.yml a partir de los ficheros Dockerfile que se indiquen
docker-compose top	muestra los procesos que están ejecutándose en cada uno de los contenedores de los servicios

3. ¿Qué es un fichero YAML? ¿Qué estructura tiene? Investiga y detállalo. Haz también un glosario de YAML.

Es un lenguaje de declaración de datos que se encarga de almacenar archivos de configuración.

Los ficheros YAML empiezan con --- y la sintaxis de mapeo es clave: valor (el espacio es importante). Admite diferentes tipos de datos como caracteres, cadenas, números, colecciones, listas... en este lenguaje es muy importante el uso de espacios y distingue entre mayúsculas y minúsculas.

Version:	muestra la versión
services:	lista los diferentes servicios
service:	configuración de un servicio
image:	imagen usada
build:	ruta del directorio dockerfile
entrypoint:	sobreescribe entrypoint por defecto
container_name:	nombre del contenedor
ports:	puertos públicos
volumes:	volúmenes que usa
networks:	redes que usa

4. Crea un archivo de configuración para una aplicación que contiene un único servicio:

- Imagen:** httpd:2.4 (del servidor web Apache)
- Puerto:** el host de Docker publicará el puerto 80 y hará una redirección con el puerto 80 del contenedor.
- Bind mount:** lo creamos entre el directorio actual del host de Docker y el directorio `usr/local/apache2/htdocs/` del contenedor (que es el directorio que utiliza el servidor web para servir el contenido que encuentre en su interior).

Version: '1.8'

services:

httpd:

image: httpd:2.4

ports:

- 80:80

volumes:

- ./usr/local/apache2/htdocs/

5. Consulta la lista de contenedores que están en ejecución y explica la salida.**¿Qué diferencia hay entre “docker ps y docker compose ps”?**

Con docker ps obtenemos desde cualquier directorio una lista de todos los contenedores que tiene Docker activos, mientras que con docker compose ps solo obtenemos los contenedores en ejecución del archivo docker-compose.yml, si hemos ejecutado el comando del directorio que contiene el archivo docker-compose.yml.

6. Muestra la salida de docker-compose.yml.

docker-compose logs

7. Visiona el video relativo a docker compose del módulo 7.4.

[Enlace al vídeo](#)