



docker

Profesor: Víctor Villegas Borge

Ejercicios

Ejercicio 1. Ejecutar la aplicación hello-world

Ahora que hemos visto qué es y cómo funciona Docker, vamos a probar a usar este programa.

Ejecuta el siguiente comando:

```
$ docker run hello-world
```

¿Qué aparece? ¿Qué ha sucedido? Explícalo con tus propias palabras.

Vuelve a ejecutar el comando `$ docker run hello world`. ¿Qué sucede? ¿Qué diferencia aprecias comparado con la primera vez que lo ejecutaste?

Ejercicio 2. Crear imagen de Ubuntu

En este segundo ejercicio vamos a realizar algo más ambicioso que simplemente lanzar una aplicación hello-world.

Para ello, vamos a descargar y lanzar una imagen de Ubuntu.

Siguiendo la filosofía del ejercicio anterior: ¿Cuál crees que deberá ser el comando que tendremos que ejecutar? Escríbelo a continuación:

```
$
```

¿Y si quiero abrir dentro del contenedor la bash del sistema?

```
$
```

Ejercicio 3. Ejecutar imagen de Ubuntu y usar la bash

En este ejercicio vamos a iniciar una máquina virtual de Ubuntu (a partir del comando del ejercicio anterior) y vamos a interactuar con ella. El comando que deberíamos haber obtenido en el ejercicio anterior era:

```
$ docker run -it ubuntu bash
```

Este comando, nos permite lanzar una imagen de Ubuntu e interactuar con la bash (<https://es.wikipedia.org/wiki/Bash>). En esencia, nos permite “jugar” con la máquina que hemos montado de Ubuntu.

Si ejecutas el comando `$ ls`, verás que puedes ver el típico sistema de archivos de Linux que ya hemos visto muchas veces con anterioridad.

Así, se pide que en este ejercicio realices lo necesario para poder (dentro de tu contenedor) crear un documento en el directorio `/home/carpeta` a través del comando:

```
$ nano documento.txt
```

Ejercicio 4. Crear una imagen a partir de las modificaciones de un contenedor.

Una vez hayas completado el ejercicio anterior, sal de la bash del contenedor de Ubuntu a través del comando: `$ exit`.

Vuelve a ejecutar el comando: `$ docker run -it ubuntu bash`

Ahora ejecuta el comando `nano`. ¿Qué sucede? ¿Por qué se debe? Demuestra tu respuesta con las evidencias necesarias.

Una vez hayas acabado con la demostración, lanza un nuevo contenedor en el que crees un documento con el comando `nano` en el directorio `/home/carpeta/subcarpeta` que se llame `index.html` y que esté vacío.

Una vez hecho, crea una nueva imagen de un Ubuntu con `nano` instalado y el documento creado.

Ejercicio 5. Ver imágenes creadas, contenedores creados y borrar.

En ejercicios anteriores ya hemos creado una cierta cantidad de imágenes. En el anterior ejercicio creamos una imagen llamada hello-world. Para ello, ejecutamos el comando `$ docker run hello-world`, y dado que en nuestro sistema no existía dicha imagen, nuestro cliente se puso en contacto con el servidor para descargarla y ejecutar dicho comando.

Se pide que en este ejercicio veas qué imágenes has creado hasta el momento, qué contenedores han sido creados y determines el estado actual de los contenedores.

Finalmente, elimina todos los contenedores y las imágenes del sistema.

Ejercicio 6. Crear un Dockerfile y crear una imagen a partir de ello.

En este ejercicio se pide que realices un documento Dockerfile que cree una imagen de Ubuntu en la que se actualice el S.O.

Como primera aproximación, puedes basarte en el código que hemos visto en clase:

```
FROM ubuntu
ADD . /app
RUN apt-get update -y && apt-get upgrade -y
```

Una vez creado el Dockerfile, se pide que crees la imagen a partir de este documento. Finalmente, deberás lanzar un contenedor a partir de la imagen creada.

Cosas que has de tener en cuenta:

- Tanto el Dockerfile como cualquier cosa que dependa de él, deberá incluirse dentro de la misma carpeta.
- El comando `"docker build -t [nombre] ."` permite crear una imagen a partir del contenido de un documento Dockerfile.

Ejercicio 7. Crear un Dockerfile (2).

En este ejercicio se pide que realices un documento Dockerfile que te permita crear una imagen con un Sistema Operativo Ubuntu, actualizar dicho S.O., instalar el programa nano. Además, la propia imagen deberá crear una carpeta en el directorio `/app` con el nombre: "Canciones", en la que se copiará un documento txt que haya sido creado desde tu máquina real en Windows que contenga el título de 3 canciones de Elvis.

Una vez haya sido creada la imagen, se pide que lances un contenedor de la misma que te permita acceder a la bash de Ubuntu y compruebes que las labores se han realizado satisfactoriamente.

Ejercicio 8. Tetris

En este ejercicio vamos a lanzar una aplicación que nos permita ejecutar tetris en nuestro navegador web. Explica qué has hecho para poder ejecutar tetris en tu navegador.

Ejercicio 9. Portainer

En este ejercicio, vas a aprender a usar docker compose, lanzando la herramienta portainer.io. Para ello, sigue los pasos de la siguiente página web: <https://jacar.es/como-instalar-portainer-con-docker-compose/>.

Una vez hecho, intenta entender el código del docker-compose que has utilizado. Finalmente, prueba a gestionar las diversas imágenes y contenedores que tienes en tu sistema. Mira cuántos recursos están consumiendo, el tamaño que ocupan, y elimina los que no necesites (si no tienes imágenes o contenedores, crea un par de contenedores con las imágenes hello-world y ubuntu, por ejemplo).