

Big Data con Hadoop y Spark

Módulo 02 – Desafío

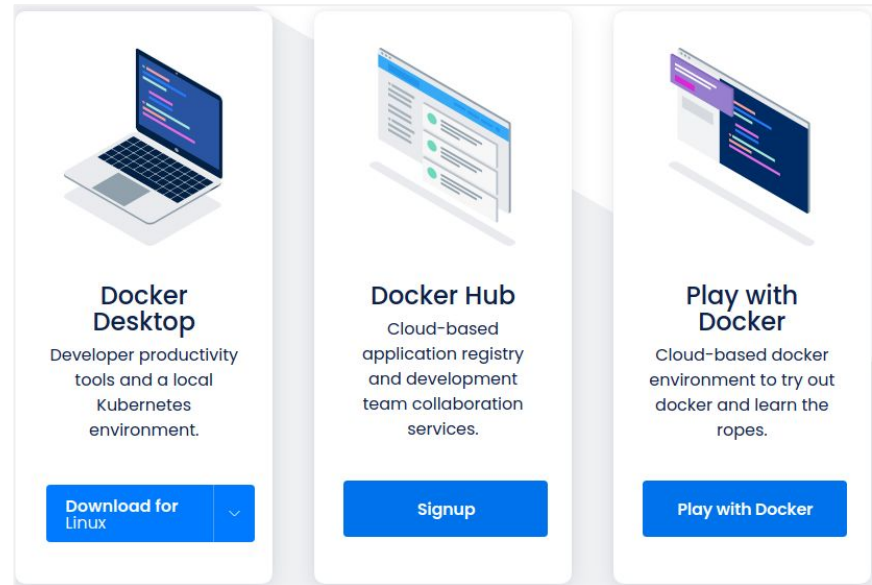
¿Qué son los desafíos?

- Son **ejercicios complementarios** a los realizados en clase.
- Permiten practicar y consolidar los conocimientos adquiridos.
- Es recomendable que los realices antes de rendir el examen de la clase.



Consideraciones

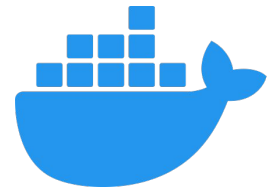
1. Es necesario tener instalado **Docker**.
2. Registrarnos en Docker Hub.
<https://hub.docker.com/>
3. Al ejecutar las instrucciones, anteponer **"sudo"**.



Ejercicio 1: Volúmenes

1. La parte del disco donde Docker crea los volúmenes es accesible sólo por Docker, por lo tanto es más seguro y ordenado.
 - a. Ver los volúmenes creados.
 - b. Crea el volumen de nombre *“dbdata”*.
 - c. Montar el volumen *“dbdata”* al contenedor en el destino **/data/db** y ejecuta la base de datos Mongo.
 - d. Ingresar al contenedor.
 - e. Conectarse a la BBDD.
 - f. Cargar y visualizar el dato cargado.
 - g. Crear un nuevo contenedor, se usa el mismo.

2. Correr un ubuntu.
3. Acceder al bash.
4. En el contendedor, se crea el directorio "test", al salir del contenedor para copiar un archivo dentro del contenedor.
5. Copiar desde el contenedor a la máquina anfitrión.



Ejercicio 2: Imágenes

Contienen distintas capas de datos (distribución, diferente software, librerías y personalización).

1. Ver las imágenes que tengo localmente.
 - a. Bajar la imagen de ubuntu con una versión específica.
2. Crear un directorio en mi máquina.
 - a. Entrar al directorio.
 - b. Crear un Dockerfile.
 - c. Abrir code en el directorio en el que estoy.



3. **##Contenido del Dockerfile##**

```
FROM ubuntu:latest
```

```
RUN touch /ust/src/hola.txt (comando  
a ejecutar en tiempo de build)
```

```
##fin##
```

4. Crear una imagen con el contexto de build
y correr el contenedor con la nueva imagen.

a. Loguearse en docker hub.

b. Cambiar el tag para poder subirla a mi
docker hub.

5. Publicar la imagen a mi docker hub.

La importancia de entender el **sistema de capas** consiste en la **optimización de la construcción del contenedor para reducir espacio** ya que cada comando en el dockerfile crea una capa extra de código en la imagen.

Con **docker commit** se crea una nueva imagen con una capa adicional que modifica la capa base.

Ejercicio 3: Imágenes

1. Crear una nueva imagen a partir de la imagen de Ubuntu.
2. Ver la información de como se construyó cada capa.
3. Ver la información de la imagen con el programa dive.



¡Terminaste el módulo!
Estás listo para rendir el examen