

# INSTALACIÓN DE ORACLE MEDIANTE DOCKER

Base de Datos  
CFGs DAW

Francisco Aldarias Raya  
[paco.aldarias@ceedcv.es](mailto:paco.aldarias@ceedcv.es)

2019/2020

Fecha 15/11/19

Versión:191115.1153


## Licencia



**Reconocimiento - NoComercial - CompartirIgual (by-nc-sa):** No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

## Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

 Importante

 Atención

 Interesante

## Revisión:

11-11-2019. Apartado 3. Ubuntu.

# ÍNDICE DE CONTENIDO

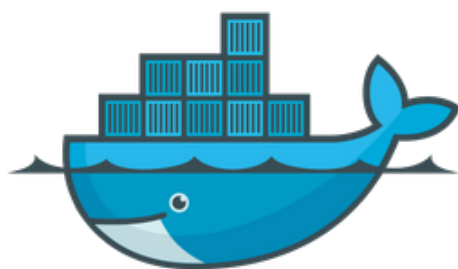
<b>1.INTRODUCCIÓN.....</b>	<b>3</b>
1.1 Imágenes de contenedor.....	3
<b>2.WINDOWS/MAC.....</b>	<b>4</b>
<b>3.UBUNTU.....</b>	<b>9</b>
3.1 Instalación de docker.....	9
3.2 Instalando contenedor.....	11
<b>4.PRIMERA TOMA DE CONTACTO.....</b>	<b>14</b>
<b>5.COMENZANDO CON ORACLE.....</b>	<b>15</b>
5.1 Aspectos generales.....	15
5.2 Aspectos a tener en cuenta.....	16
5.3 Creación de un TABLESPACE.....	16
5.4 Creación de un usuario.....	17
5.5 Ejecución de Scripts.....	18
<b>6.BIBLIOGRAFÍA.....</b>	<b>21</b>

## UD04. INSTALACIÓN DE ORACLE MEDIANTE DOCKER

### 1. INTRODUCCIÓN

Docker es un proyecto de código abierto que automatiza el despliegue de aplicaciones dentro de contenedores de software, proporcionando una capa adicional de abstracción y automatización de virtualización de aplicaciones en múltiples sistemas operativos.

Un **contenedor** con docker es más ligero, ya que mientras que a una máquina virtual con virtualbox necesitas instalarle un sistema operativo para funcionar, un **contenedor** de **Docker** funciona utilizando el sistema operativo que tiene la máquina en la que se ejecuta el **contenedor**.

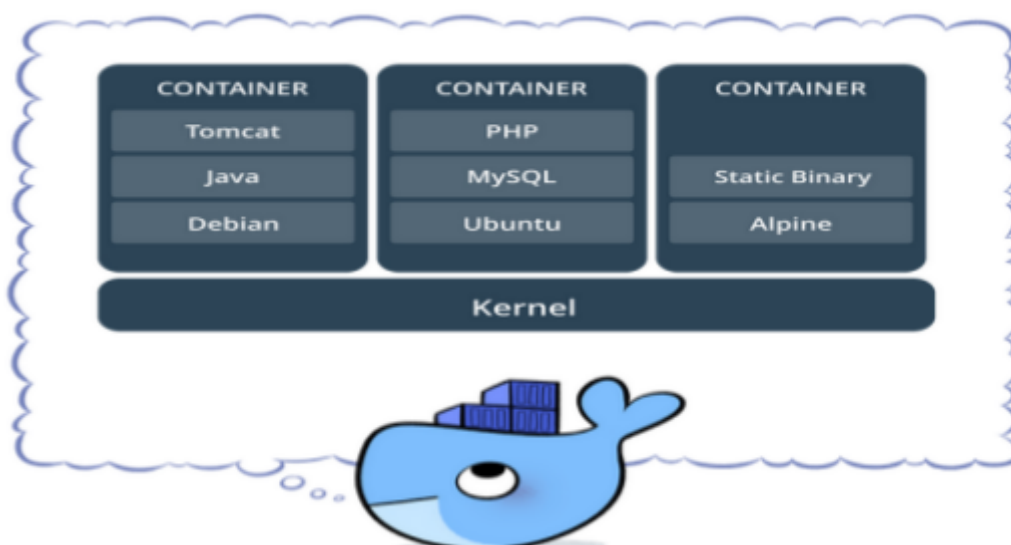


Dado que la instalación de Oracle 11g xe sobre Linux, Windows o Mac, puede generar múltiples errores, algunas veces difíciles de solucionar, o como mínimo que requieren de mucho tiempo y dedicación para ello, se recomienda utilizar un contenedor Docker, que ya dispone de la versión de Oracle que utilizaremos en el curso y que podréis instalarla sobre cualquier sistema operativo.

#### 1.1 Imágenes de contenedor

Una imagen de contenedor es un paquete ligero, independiente y ejecutable de una pieza de software que incluye todo lo necesario para ser ejecutado: código, herramientas y bibliotecas del sistema, configuraciones, etc.

Está disponible para aplicaciones basadas en Linux y Windows. El software en contenedores siempre funcionará igual, independientemente del entorno. Los contenedores aíslan el software de su entorno y ayudan a reducir los conflictos entre los equipos que ejecutan software diferente sobre la misma infraestructura.



Para más información sobre qué es un contenedor pincha aquí. <https://www.docker.com/what-container>

## 2. WINDOWS/MAC

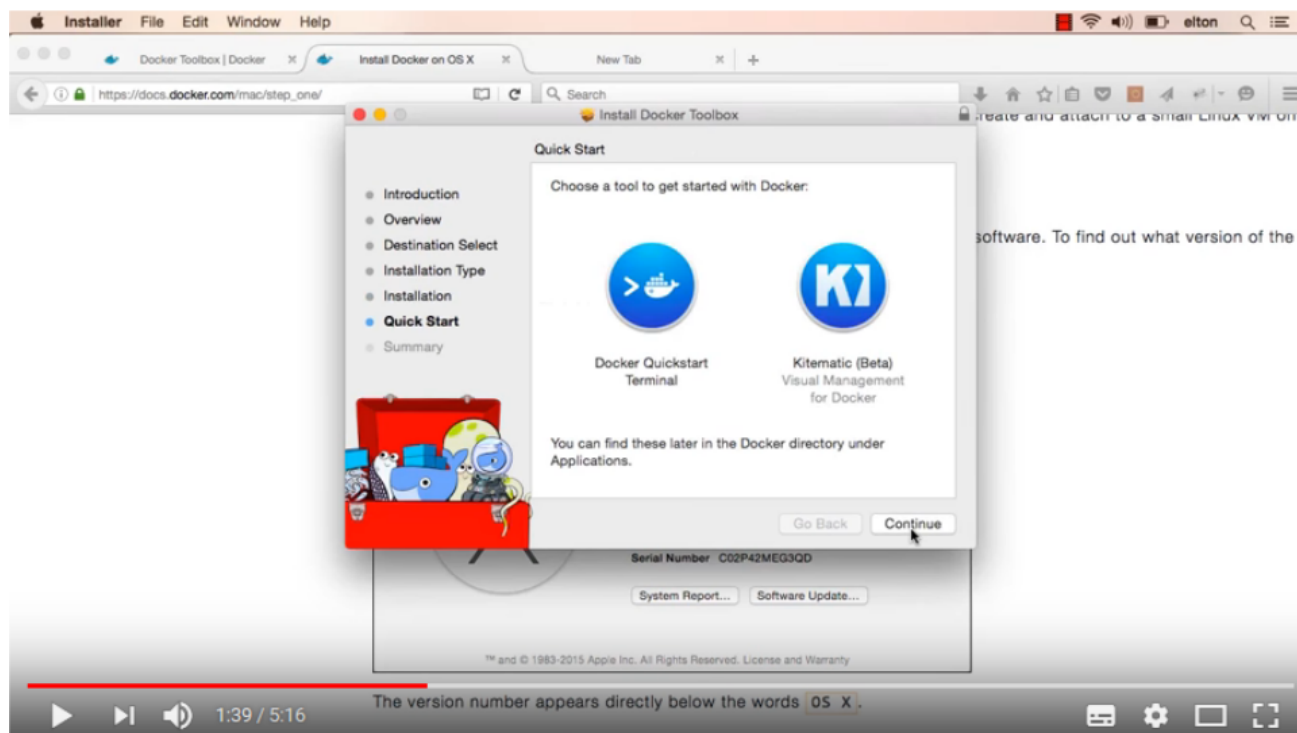
En el siguiente apartado veremos como instalar docker y oracle en máquinas con sistema operativo windows y mac.

Para Mac y Windows desde el enlace se accede a un paquete Toolbox que ya incluye Docker y Kitematic. En el caso de Ubuntu habrá que instalarlo por separado.

Podemos instalarnos Docker y su interfaz gráfica Kitematic desde los siguientes enlaces, según el SO:

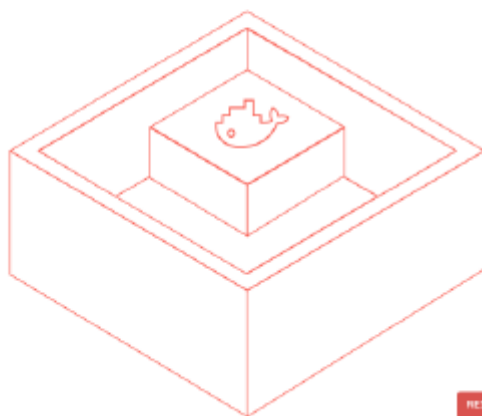
- MAC
  - web: [https://docs.Docker.com/toolbox/toolbox\\_install\\_mac/](https://docs.Docker.com/toolbox/toolbox_install_mac/)
  - vídeo <https://www.youtube.com/watch?v=INkVxDSRo7M>
- Windows
  - web: [https://docs.docker.com/toolbox/toolbox\\_install\\_windows/](https://docs.docker.com/toolbox/toolbox_install_windows/)
  - vídeo [https://www.youtube.com/watch?v=3h\\_KBqXY69w](https://www.youtube.com/watch?v=3h_KBqXY69w)

Seguiremos los pasos indicados en el vídeo correspondiente para descargar e instalar el software de Docker con windows y mac. Una vez lleguemos a la siguiente pantalla:



En lugar de continuar con el terminal, tal y como indica el vídeo, haremos clic en el icono de la herramienta gráfica de Docker (**Kitematic**). Seguiremos con la interfaz gráfica para que os sea más sencillo. Por supuesto, quien quiera aprender más sobre Docker puede continuar con la consola.

Si al hacer clic sobre Kitematic os aparece esta imagen:



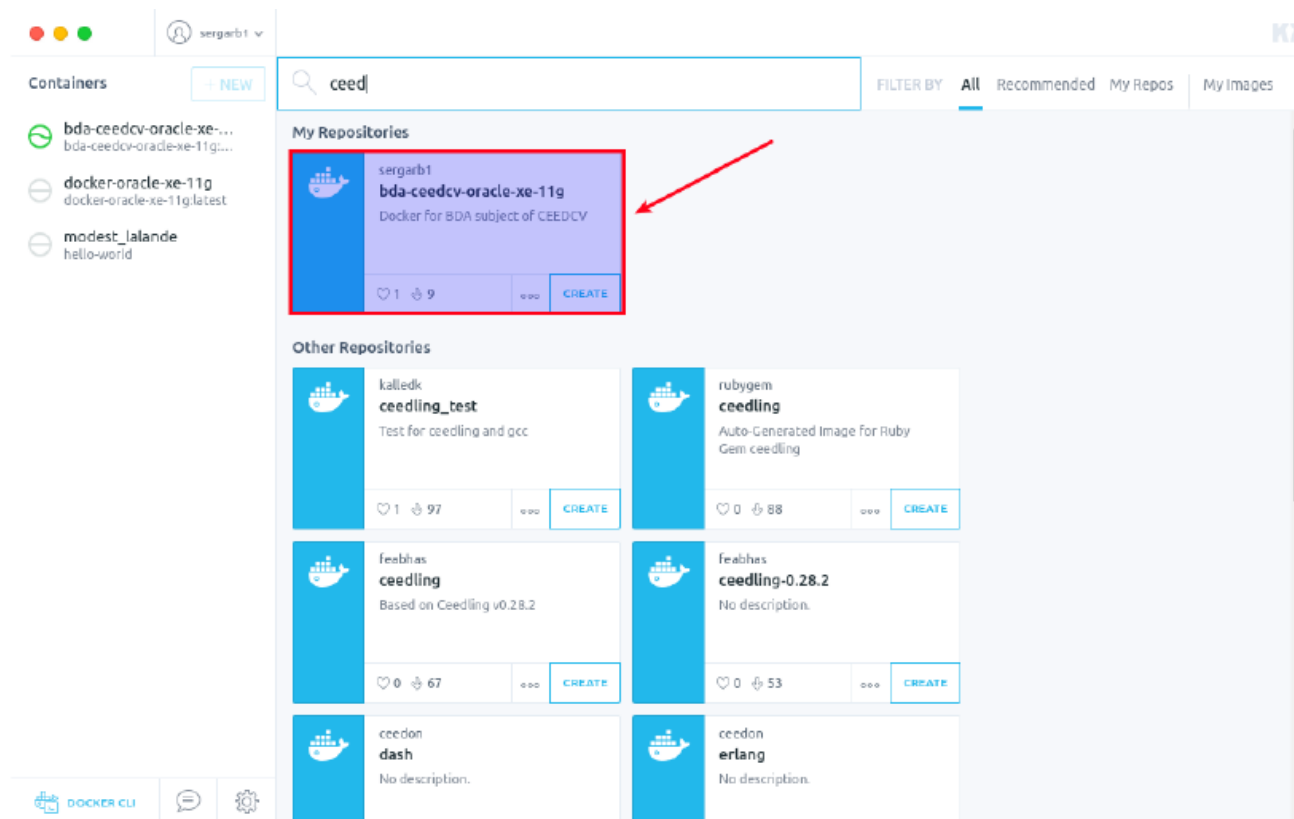
RESET

## Setup Initialization

We couldn't find a native setup - Click the Retry button to check again.

RETRY SETUP

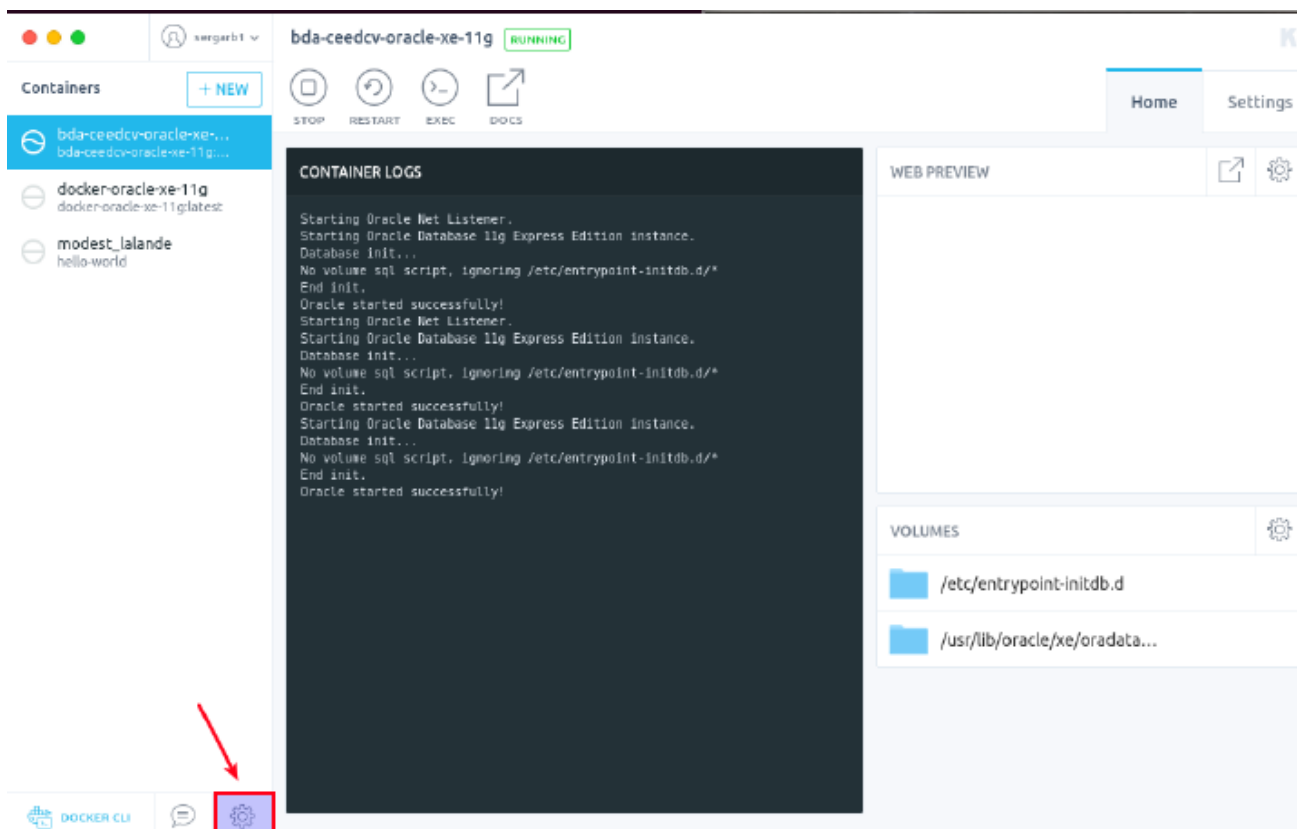
### Ejecutando Kinetic (con windows y mac)



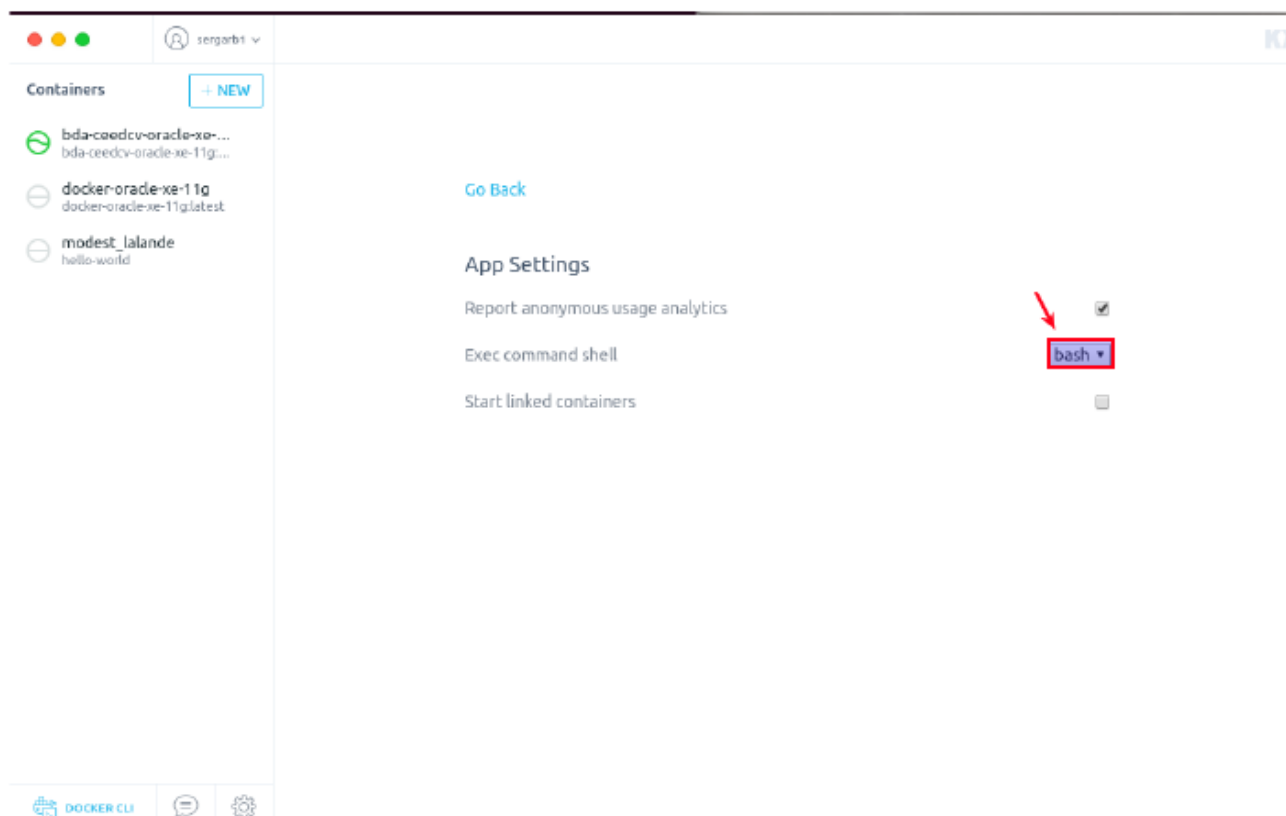
Como veis en la imagen anterior, la interfaz gráfica consta de dos partes. En la parte de la izquierda se encuentra un listado con los contenedores que tenemos ya instalados. En mi caso,

tengo tres contenedores, pero vosotros, como es la primera vez que entráis, lo tendréis vacío. Y en la parte derecha aparecen los resultados de la búsqueda.

Antes de continuar, debemos cambiar el shell que Kitematic utiliza por defecto, de so a baso. Para ello, hacemos clic en la rueda dentada de la imagen:



Y en el combo elegir baso, en lugar de so que es lo viene por defecto.



Ahora ya podemos continuar con la instalación, buscaremos la imagen de Oracle que se ha preparado para nuestro módulo: `sergarb1/bda-ceedcv-oracle-xe-11g` y haremos clic en el botón CREATE.

Que nos abrirá el terminal de la máquina. Allí ejecutaremos el comando `sqlplus`, que nos pedirá el usuario y contraseña de Oracle. Introduciremos los siguientes valores:•

user-name: system

password: oracle



```
root@56beaf355131: /  
File Edit View Search Terminal Help  
root@56beaf355131:/# sqlplus  
  
SQL*Plus: Release 11.2.0.2.0 Production on Fri Feb 16 11:31:25 2018  
  
Copyright (c) 1982, 2011, Oracle. All rights reserved.  
  
Enter user-name: system  
Enter password:  
ERROR:  
ORA-28002: the password will expire within 4 days  
  
Connected to:  
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production  
SQL> █
```

### 3. UBUNTU

La instalación con linux de oracle con docker se hace desde la terminal. Los pasos a seguir pasan por instalar docker, seguidamente bajamos el fichero de instalación oracle, posteriormente instalamos la imagen de oracle con docker.

#### 3.1 Instalación de docker

Seguidamente vamos a ver el proceso de instalación de docker desde terminal para ubuntu:

1. Comprobamos la versión de ubuntu:

```
cat /etc/lsb-release
```

```
paco@pacoceed:~$ cat /etc/lsb-release  
DISTRIB_ID=Ubuntu  
DISTRIB_RELEASE=16.04  
DISTRIB_CODENAME=xenial  
DISTRIB_DESCRIPTION="Ubuntu 16.04.6 LTS"
```

2. Actualizamos ubuntu:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

3. Instalamos docker:

```
sudo apt install docker.io
```

4. Para añadir docker en el arranque:

```
sudo systemctl start docker  
sudo systemctl enable docker
```

5. Para ver la versión:

```
docker version
```

```
paco@pacocceed:~$ docker --version  
Docker version 18.09.7, build 2d0083d  
paco@pacocceed:~$
```

6. Ejecutar docker sin ser root.

Vamos a ejecutar docker por parte del usuario paco.

```
sudo usermod -aG docker paco
```

Reiniciamos el ordenador para que coja los cambios

Ahora nos logeamos como paco y probamos a ejecutar el comando docker run

```
docker run hello-world
```

```
paco@pacocceed:~$ docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
1b930d010525: Pull complete  
Digest: sha256:c3b4ada4687bbaa170745b3e4dd8ac3f194ca95b2d0518b417fb47e5879d9b5f  
Status: Downloaded newer image for hello-world:latest  
  
Hello from Docker!  
This message shows that your installation appears to be working correctly.
```

Esto nos informa que tenemos docker instalado.

Una vez instalado el contenedor se puede ver con:

```
docker images
```

```
paco@pacoceed:~$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
hello-world         latest             fce289e99eb9       10 months ago      1.84kB
paco@pacoceed:~$
```

Para más información:

- instalación: <https://docs.docker.com/install/linux/docker-ce/ubuntu/>
- tutorial: <https://www.howtoforge.com/tutorial/ubuntu-docker/>
- web: <https://docs.docker.com/>
- video: <https://www.youtube.com/watch?v=eDspGMilhxl>

### 3.2 Instalando contenedor

Busqueda de contenedores:

```
docker search oracle
```

```
NAME                DESCRIPTION                STARS     OFFICIAL   AUTOMATED
oraclelinux         Official Docker builds of Oracle Linux.    616      [OK]
jaspeen/oracle-11g  Docker image for Oracle 11g database       130
```

También se puede ver en:

<https://hub.docker.com/search?q=oracle&type=image>

Clonamos el repositorio:

```
cd $HOME
```

```
git clone https://github.com/oracle/docker-images.git
```

Nos situamos en la carpeta dockerfiles

```
cd $HOME/docker-images/OracleDatabase/SingleInstance/dockerfiles/
```

Siendo {ORACLE\_DOCKER\_DIR} la carpeta

```
$HOME/docker-images/OracleDatabase/SingleInstance/dockerfiles/
```

Para poder crear la imagen, necesitamos descargar el instalador oficial de la base de datos (Express Edition Linux x86-64), el cual se puede encontrar en [esta página](#). Almacenarlo en {ORACLE\_DOCKER\_DIR}/11.2.0.2

La estructura final debería ser parecida a la que se muestra a continuación:

```
{ORACLE_DOCKER_DIR}
|_ buildDockerImage.sh
|_ 11.2.0.2
|_ Checksum.xe
|_ Dockerfile.xe
|_ checkDBStatus.sh
|_ runOracle.sh
|_ setPassword.sh
|_ xe.rsp
|_ oracle-xe-11.2.0-1.0.x86_64.rpm.zip
```

Ejecuta el archivo que construye la imagen y que se encuentra almacenado en {ORACLE\_DOCKER\_DIR}. De acuerdo a tu instalación de docker es posible que requieras permisos de superusuario:

```
./buildDockerImage.sh -v 11.2.0.2 -x
```

```
paco@pacocceed:~/docker-images/OracleDatabase/SingleInstance/dockerfiles$ ./buildDockerImage.sh -v 11.2.0.2 -x
Checking if required packages are present and valid...
oracle-xe-11.2.0-1.0.x86_64.rpm.zip: La suma coincide
=====
DOCKER info:
Containers: 1
  Running: 0
  Paused: 0
  Stopped: 1
Images: 1
Server Version: 18.09.7
```

```
Successfully tagged oracle/database:11.2.0.2-xe
```

```
Oracle Database Docker Image for 'xe' version 11.2.0.2 is ready to be extended:
```

```
--> oracle/database:11.2.0.2-xe
```

```
Build completed in 179 seconds.
```

Después de un par de minutos, la imagen es creada e instalada localmente. Puedes comprobarlo usando el comando:

```
docker images
```

```
paco@pacocceed:~/docker-images/OracleDatabase/SingleInstance/dockerfiles$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
oracle/database      11.2.0.2-xe        2f6738f064f3       About a minute ago  1.13GB
oraclelinux          7-slim             874477adb545       3 months ago       118MB
hello-world          latest             fce289e99eb9       10 months ago      1.84kB
paco@pacocceed:~/docker-images/OracleDatabase/SingleInstance/dockerfiles$
```

A partir de este punto ya puedes crear cuantos contenedores necesites de la imagen que se acaba de construir. Para ello, podemos ejecutar el siguiente comando. Más información acerca de los params usados [aquí](#):

```
docker run -d --name oraclexe \  
--shm-size=1g \  
-p 1521:1521 -p 8080:8080 \  
-e ORACLE_PWD=password \  
-v /u01/app/oracle/oradata \  
-v /opt/oracle/oradata \  
-v /opt/oracle/scripts/setup \  
-v /home/paco/oracle/myscripts /u01/app/oracle/scripts/startup \  
oracle/database:11.2.0.2-xe
```

Una vez ejecutado, la creación del contenedor configurará la base de datos de acuerdo a los parámetros establecidos. El proceso puede tomar alrededor de 3-5 minutos, como la ejecución se realizó usando la opción -d (Detached) se debe monitorear los logs del contenedor para saber cuando se puede comenzar a usar:

```
docker container logs -tf oraclexe
```

Cuando veas el siguiente mensaje en los logs, quiere decir que la base de datos está lista:

```
#####  
DATABASE IS READY TO USE!  
#####
```

Ahora, si quieres activar el usuario/esquema HR puedes hacerlo conectándote a sqlplus del contenedor y ejecutando el comando que activa dicha cuenta:

```
docker exec -ti oraclexe sqlplus sys/password@//localhost:1521/XE as sysdba  
ALTER USER hr IDENTIFIED BY password ACCOUNT UNLOCK;
```

```
paco@pacoced:~$ docker exec -ti oraclexe sqlplus sys/password@//localhost:1521/XE as sysdba

SQL*Plus: Release 11.2.0.2.0 Production on Mon Nov 11 10:30:46 2019

Copyright (c) 1982, 2011, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
SQL> █
```

Borrado instancia del contenedor que hemos realizado con docker run:

```
docker rm oraclexe
```

Más información en:

- Tutorial <http://www.acelopez.com/docker/oracle-express-docker-container/>

## 4. PRIMERA TOMA DE CONTACTO

Si hemos llegado oasta aquí quiere decir que oemos instalado correctamente Oracle y hemos abierto la línea de comandos.

Recuerda que en la teoría explicamos que existan dos usuarios administradores: SYS y SYSTEM. Bien, al abrir la línea de comandos nos hemos conectado como SYSTEM. Prodemos corroborarlo ejecutando el comando show user para ver el usuario con el que estamos conectados.

```
SQL> show user
USER is "SYSTEM"
SQL> █
```

El usuario SYSTEM equivale a SYS.

Sin embargo, la parte que se explica en la teoría a continuación sobre los roles especiales SYSDBA y SYSOPER, en la imagen de Docker que os hemos proporcionado no funcionan, pero el usuario SYSTEM está preparado para ejecutar todos los comandos que en una instalación real de Oracle podríamos a través de estos roles:

Oracle tiene dos roles especiales como son el de Administrador de la Base de Datos (SYSDBA) y el de Operador de Base de Datos (SYSOPER) que tiene privilegios especiales para el mantenimiento y la administración de Oracle.

Si quieres conectar como SYSDBA puedes utilizar el comando conn de la siguiente forma. Si

hemos entrado con docker y linux , ya somos sysdba.

```
SQL> connect as sysdba
Enter user-name: system
Enter password:
Connected.
SQL>
```

Si quieres conectarte con el rol SYSDBA sin utilizar un usuario concreto puedes emplear conn / as sysdba.

```
SQL> conn / as sysdba
Connected.
SQL> show user
USER is "SYS"
SQL> _
```

## 5. COMENZANDO CON ORACLE

### 5.1 Aspectos generales

Para comenzar a trabajar con Oracle debemos tener en cuenta los siguientes aspectos generales:

- La línea de comandos de Oracle en Windows no distingue entre mayúsculas y minúsculas ni en las palabras reservadas ni en los nombre de los objetos (tablas, campos, etc.), pero sí en otros sistemas operativos como Linux. Intentad respetarlo como principio.
- Existe una tabla llamada dual que es visible por todos los usuarios y que permite seleccionar las variables del sistema.

```
SQL> select sysdate, user from dual;

SYSDATE    USER
-----
10-NOV-19  SYS
```

- Las instrucciones terminarán con punto y coma ;
- Podemos añadir comentarios de una línea tecleando dos guiones seguidos -- y podemos crear comentarios de bloque que afecten a una o varias líneas comenzando el comentario con /\* y terminando con \*/
- Las instrucciones se pueden teclear en varias líneas, Oracle irá mostrando líneas numeradas hasta completar la instrucción. Si dejamos una línea en blanco la línea de comandos interpretará que deseamos cancelar el comando que estamos escribiendo y volveremos al prompt SQL>

```
SQL> select sysdate,  
2 user  
3 from dual;  
  
SYSDATE    USER  
-----  
10-NOV-19  SYS  
  
SQL> 
```

## 5.2 Aspectos a tener en cuenta

En Oracle solamente tendremos una base de datos con toda la información relevante de la organización. Sin embargo, con MySQL podemos crear todas las bases de datos que consideremos oportunas.

- Cada usuario de una base de datos posee un esquema. El esquema tendrá el mismo nombre que el usuario y servirá para almacenar todos los objetos que cree el usuario, por ejemplo tablas, vistas, índices, sinónimos, etc.

- Los objetos que son creados por un usuario solamente podrán ser manipulados por el usuario y los administradores de la base de datos. Aunque el propietario o los administradores podrán proporcionar privilegios específicos a otros usuarios para poder acceder a ellos e incluso para modificarlos.

- Para crear un usuario nos conectaremos a Oracle como Administradores (sysdba), pero no es necesario desde la imagen de Docker . Una vez creado podremos asignarle los permisos que consideremos oportunos para trabajar.

## 5.3 Creación de un TABLESPACE

Para Oracle, un TABLESPACE es un espacio lógico de almacenamiento de tablas que puede estar compuesto por uno o varios ficheros.

Siempre que se crea un nuevo usuario, hay que indicarle qué espacio de tablas va a utilizar. En caso de no especificarse ninguno, se utiliza el espacio por defecto USERS.

Aunque para crear nuestras tablas podemos utilizar el espacio por defecto USERS, resulta más recomendable que nos creamos nuestro propio espacio de tablas.

Vamos a llamar a nuestro espacio de tablas datosalumno y lo almacenaremos en la ruta por defecto donde se encuentran el resto de espacios de tablas:

C:\oraclexe\app\oracle\oradata\XE

El comando a ejecutar es el siguiente:

```
CREATE TABLESPACE datosalumno DATAFILE 'C:\oraclexe\app\oracle\oradata\XE\datosalumno01.dbf'  
SIZE 50M AUTOEXTEND ON NEXT 50M MAXSIZE 300M EXTENT MANAGEMENT LOCAL UNIFORM  
SIZE 10M SEGMENT SPACE MANAGEMENT AUTO;
```



Para ubuntu seria:

```
CREATE TABLESPACE datosalumo DATAFILE '/opt/oracle/oradata/datosalumeo01.dbf' SIZE 50M  
AUTOEXTEND ON NEXT 50M MAXSIZE 300M EXTENT MANAGEMENT LOCAL UNIFORM SIZE 10M  
SEGMENT SPACE MANAGEMENT AUTO;
```

Vamos a comentar un poco el comando anterior:

- Crea un espacio de nombres llamado datosalumno.
- El fichero que almacenará las tablas que se creen dentro de disco espacio se llamará datosalumno.dbf y se encontrará en la ruta donde están los espacios de tablas que Oracle tiene por defecto.
- El tamaño inicial del fichero será de 50MD, pudiendo aumentar otros 50MD si se llena. Este proceso se puede repetir hasta llegar a un tamaño máximo de 300MD.
- La gestión de las extensiones de este espacio de tablas se hará localmente.
- La gestión de segmentos de este espacio de tablas se hará automáticamente.

Nota: Si en algún momento nos quedamos sin espacio dentro del fichero creado, Oracle nos dará error al crear nuevas tablas o agregar más información. La solución reside en redimensionar el fichero, no el TABLESPACE (de nada sirve hacer más grande el TABLESPACE si el fichero tiene un límite inferior).

```
ALTER DATABASE DATAFILE AUTOEXTEND ON NEXT 5M MAXSIZE 1G;
```

#### 5.4 Creación de un usuario

Mientras no sea estrictamente necesario, es conveniente utilizar otro usuario que no sea system por varios motivos:

1. El usuario system tiene el espacio SYSTEM como espacio de tablas por defecto, estando aquí las tablas del propio sistema gestor de base de datos. El estar trabajando sobre este espacio puede ser peligroso ya que podríamos borrar o modificar alguna tabla del sistema de manera accidental.
2. No tiene sentido mezclar nuestras tablas de usuario con las tablas del sistema.
3. Como usuario administrador, tenemos control y acceso sobre todas las tablas del sistema, pudiendo borrar o modificar tablas no deseadas de manera accidental.

Por todo lo anterior, es conveniente crear un nuevo usuario. A continuación crearemos el usuario alumno, con contraseña alumno, a través de la consola de Oracle.

```
CREATE USER alumno IDENTIFIED BY alumno  
DEFAULT TABLESPACE datosalumno;
```

Queremos que el usuario tenga permisos de conexión y que pueda gestionar sus propios recursos (tablas, vistas, etc) así que, desde la consola, tecleamos el siguiente comando:

```
GRANT CONNECT, RESOURCE TO alumno;
```

Ahora ya podemos hacer la prueba de conectarnos con nuestro nuevo usuario y hacer una consulta sobre una tabla de uso general:

```
SQL> CREATE USER alumno IDENTIFIED BY alumno
2  DEFAULT TABLESPACE datosalumno;

User created.

SQL> GRANT CONNECT, RESOURCE TO alumno;

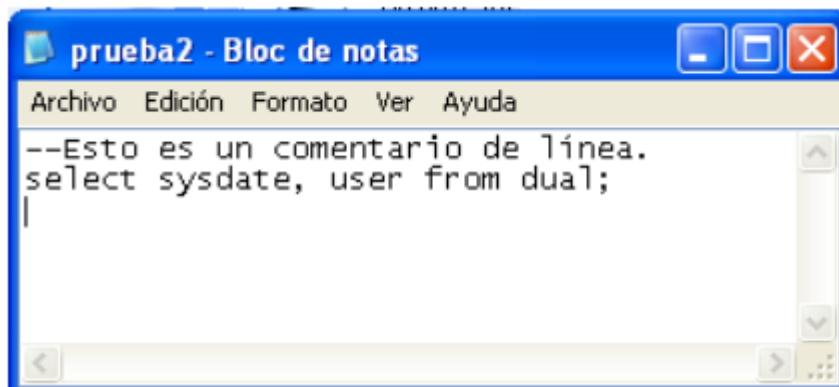
Grant succeeded.

SQL> connect alumno
Enter password:
Connected.
SQL> show user
USER is "ALUMNO"
SQL>
```

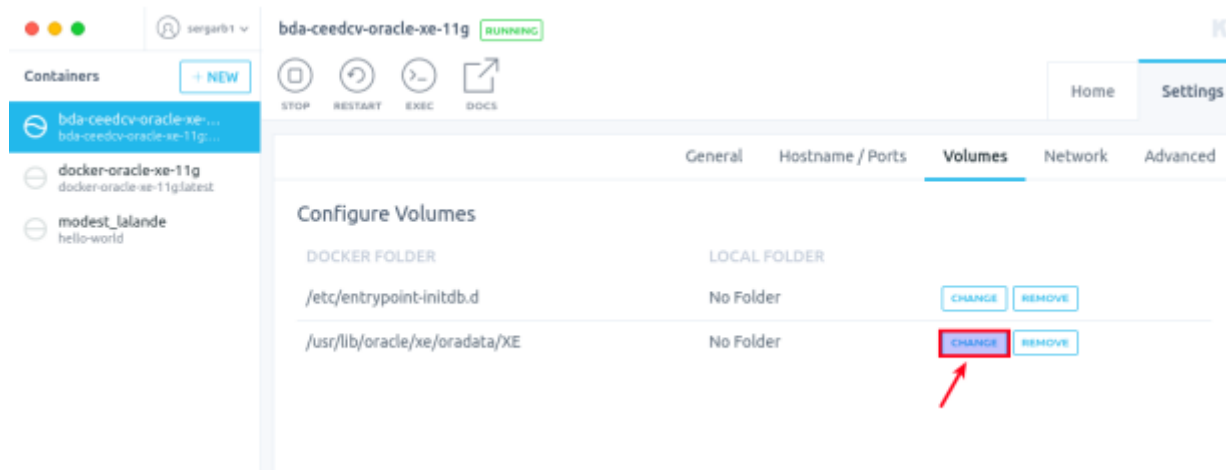
### 5.5 Ejecución de Scripts

Una de las operaciones más frecuentes que se realiza si trabajamos con la consola será ejecutar scripts. Un script es simplemente un archivo de texto (normalmente con extensión .sql) cuyo contenido es un conjunto de instrucciones que deseamos ejecutar en la línea de comandos.

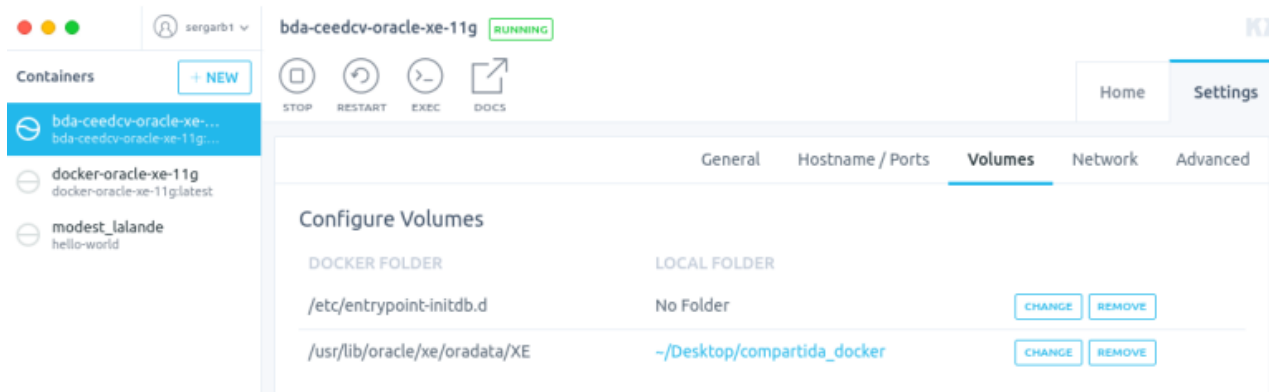
El archivo lo podemos crear con cualquier editor, desde el bloc de notas a editores específicos. Por ejemplo con el bloc de notas podemos crear el siguiente archivo de prueba (prueba2.sql).



Al trabajar con Docker deberemos habilitar una carpeta compartida entre la máquina física y la de Docker para poder almacenar allí nuestros scripts y poder ejecutarlos en Docker. Para ello vamos a Kitematic y hacemos clic en el botón Settings y después en Volumes:



Seguidamente hacemos clic en el botón CHANGE de la carpeta /usr/lib/oracle/xe/oradata/XE tal y como se indica en la imagen anterior, e introducimos una carpeta de nuestro equipo donde vayamos a ir almacenando todos los scripts de Oracle (en mi caso Desktop/compartda\_docker)



Copiaremos nuestro fichero prueba2.sql en la ruta compartida de nuestra máquina3 y para ejecutarlo utilizaremos el símbolo @ seguido de la ruta y el nombre del archivo que contiene el script a ejecutar. Por ejemplo:

```
SQL> @/usr/lib/oracle/xe/oradata/XE/prueba2.sql  
  
SYSDATE    USER  
-----  
16-FEB-18  ALUMNO  
  
SQL> █
```

Como puedes observar el resultado obtenido es el mismo que si hubiésemos realizado su ejecución directamente desde el prompt de Oracle por línea de comando.

Por último, para terminar esta toma de contacto con Oracle, saldremos de la línea de comandos tecleando **quit**.

## 6. BIBLIOGRAFÍA

- Docker: Installation and Basic Usage on Ubuntu 18.04 LTS  
<https://www.howtoforge.com/tutorial/ubuntu-docker/>
- Oracle Instalación  
<https://www.oracle.com/technology/tech/linux/install/xe-on-kubuntu.html>
- Repositorios de docker  
<https://hub.docker.com/>
- Curso de Introducción a Docker  
<https://openwebinars.net/cursos/docker-introduccion/>
-