

UNIVERSIDAD PRIVADA-DE-TACNA



INGENIERIA DE SISTEMAS

TITULO:

**INFORME DE LABORATORIO No 05**

**CURSO:**

BASE DE DATOS II

**DOCENTE(ING):**

Patrick Cuadros Quiroga

Integrantes:

Nilson Laura Atencio

(2015053846)

# Índice

<b>1. INFORMACIÓN GENERAL</b>	<b>1</b>
1.1. Objetivos:.....	1
1.2. Equipos y recursos utilizados: .....	1
<b>2. MARCO TEORICO</b>	<b>2</b>
2.1. Docker: .....	2
2.2. Oracle en Docker: .....	2
2.3. Docker en Windows 10: .....	2
2.4. Construir la imagen:.....	2
<b>3. PROCEDIMIENTO</b>	<b>4</b>
3.1. Parte 1: Iniciando Docker .....	4
3.2. Parte 2: Creando un contenedor con Oracle Database para Linux .....	4
3.3. Parte 3: Adicionando persistencia .....	8
<b>4. ANALISIS E INTERPRETACION DE RESULTADOS</b>	<b>10</b>
4.1. Parte 1: Actividades Encargadas .....	10
<b>5. CONCLUSIONES</b>	<b>13</b>
<b>6. REFERENCIAS</b>	<b>14</b>

# **1. INFORMACIÓN GENERAL**

## **1.1. Objetivos:**

- Conocer las funciones sobre contenedores y Docker.
- Realizar la instalación de una instancia correctamente.

## **1.2. Equipos, materiales, programas y recursos utilizados:**

- Virtualización activada en el BIOS.
- Windows 10 64bit: Pro, Enterprise o Education, con al menos 4GB de RAM.
- Docker Desktop
- Microsoft SQL Server 2017 o superior

## 2. MARCO TEORICO

### 2.1. Docker:

La idea detrás de Docker es crear contenedores ligeros y portables para las aplicaciones software que puedan ejecutarse en cualquier máquina con Docker instalado, independientemente del sistema operativo que la máquina tenga por debajo, facilitando así también los despliegues.

¡Ala! Definición traducida de la documentación oficial, demasiada información en pocas frases, no me entero de nada. Bien, empecemos por lo primero ¿qué es un contenedor? Este concepto ya es antiguo, y viene de Linux, pero por hacerte un símil con el mundo real, imagina en tu cabeza un contenedor de esos que suelen llevar los barcos de mercancías, que contiene distintos productos.

### 2.2. Oracle con Docker:

Oracle Database Server 12c R2 es un servidor de bases de datos relacionales líder en la industria.

La imagen de Oracle Database Server Docker contiene Oracle Database Server 12.2.0.1 Enterprise Edition que se ejecuta en Oracle Linux 7. Esta imagen contiene una base de datos predeterminada en una configuración de varios inquilinos con una sola base de datos conectable.

### 2.3. Docker en Windows 10:

Docker tiene dos versiones para Windows:

Versión completa:

<https://download.docker.com/win/stable/Docker%20for%20Windows%20Installer.exe>

Toolbox:

<https://docs.docker.com/toolbox/overview/>

Para usar la versión completa es necesario habilitar Microsoft Hyper-V, lo que implica deshabilitar la virtualización por hardware de nuestro PC.

Si estamos usando *VirtualBox* en el mismo host, con este cambio deja de funcionar.

*Docker Toolbox* no tiene esta restricción, aunque se mantiene como una versión antigua (Legacy), y Docker recomienda usar la versión completa.

Otra diferencia de *Docker Toolbox* es que necesita una VM *VirtualBox* para ejecutar. Esta VM se crea de forma automática al usar Toolbox, de nombre *default*, y se usa como host para los containers que creamos. Más adelante veremos detalles sobre las implicaciones de esto.

En este artículo vamos a usar *Docker Toolbox* (v 18.01.0-ce), ya que tengo en el mismo host otras VMs *VirtualBox* que me interesa seguir usando. La versión de Windows es 10.0.16299.309 64 bits.

## 2.4. Construir la imagen:

- La compilación espera el siguiente sistema de archivos. Tendrá que descargar la base de datos Oracle 19c y el software APEX usted mismo y colocarlo en el directorio "software".

```
$ arbol
.
├── Dockerfile
├── README.md
├── guiones
│   ├── healthcheck.sh
│   └── start.sh
└── software
    ├── apex_19.1_en.zip
    ├── LINUX.X64_193000_db_home.zip
    └── put_software_here.txt

PS
```

## 3. PROCEDIMIENTO

### 3.1. Parte 1: Iniciando Docker

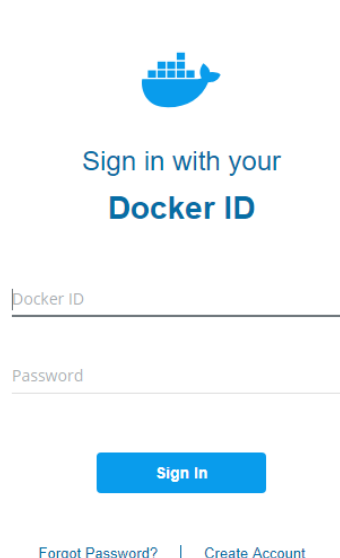
- Abrir el menu inicio y buscar la aplicación Docker for Windows.
- Ubicar la aplicación PowerShell, ejecutarla como Administrador. En la ventana de comandos de PowerShell escribir lo siguiente.

```
PS C:\> docker version
Client: Docker Engine - Community
 Version:      18.09.2
 API version:  1.39
 Go version:   go1.10.8
 Git commit:   6247962
 Built:        Sun Feb 10 04:12:31 2019
 OS/Arch:     windows/amd64
 Experimental: false

Server: Docker Engine - Community
 Engine:
  Version:      18.09.2
  API version:  1.39 (minimum version 1.24)
  Go version:   go1.10.6
  Git commit:   6247962
  Built:        Sun Feb 10 04:28:48 2019
  OS/Arch:     windows/amd64
  Experimental: false
```

### 3.2. Parte 2: Creando un contenedor con Oracle Database para Linux

- En un navegador de internet acceder a la dirección <https://hub.docker.com/>. Iniciar sesión o crear una cuenta nueva



- Buscar el repositorio para Oracle Database. Ingresar y proceder con el CheckOut, completar los datos y aceptar las condiciones obligatorias para obtener el acceso al contenido.

Welcome to Docker Hub

## Download and Take a Tutorial

Get started by downloading Docker Desktop, and learn how you can build, tag and share a sample image on Hub.

[Get started with Docker Desktop](#)



### Create a Repository

Push container images to Docker Hub



### Create an Organization

Manage Docker Hub repositories with your team

– En la ventana de PowerShell, escribir el siguiente comando:

```
PS C:\> docker login
Authenticating with existing credentials...
Login Succeeded
```

- Ejecutar el siguiente comando en Powershell, para descargar la imagen del contenedor de Oracle Database en un servidor Linux

```
PS C:\> docker pull store/oracle/database-enterprise:12.2.0.1
12.2.0.1: Pulling from store/oracle/database-enterprise
4ce27fe12c04: Pull complete
9d3556e8e792: Pull complete
fc60a1a28025: Pull complete
0c32e4ed872e: Pull complete
b465d9b6e399: Pull complete
Digest: sha256:40760ac70dba2c4c70d0c542e42e082e8b04d9040d91688d63f728af764a2f5d
Status: Downloaded newer image for store/oracle/database-enterprise:12.2.0.1
```

- Seguidamente ejecutar el comando Docker run, como respuesta se visualizará un ID que corresponde al contenedor.

```
PS C:\> docker run -d -it --name ORACLEDB01 -p 1521:1521 -p 5500:5500 store/oracle/database-enterprise:12.2.0.1
79c978c3eda27e6ee1f4506313fd66c970827f60993dc54e0f9c4bfebe47ce02
```

- Verificar que el contenedor se esté ejecutando correctamente mediante el comando Docker ps:

```
PS C:\> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED
STATUS        PORTS                    NAMES
79c978c3eda2   store/oracle/database-enterprise:12.2.0.1  "/bin/sh -c '/bin/ba..."  7 minutes ago
Up 7 minutes (healthy)   0.0.0.0:1521->1521/tcp, 0.0.0.0:5500->5500/tcp  ORACLEDB01
```

- Cuando el estado del contenedor sea “healthy”, en la consola de Powershell, ejecutar el siguiente comando:

```
PS C:\> docker exec -it ORACLEDB01 bash -c "source /home/oracle/.bashrc; sqlplus / as sysdba"

SQL*Plus: Release 12.2.0.1.0 Production on Fri May 31 21:40:55 2019

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL>
```

- En la línea de comandos de SQL\*Plus, escribir lo siguiente



```

Administrador: Windows PowerShell

OWNER
-----
TABLE_NAME
-----
TABLESPACE_NAME
-----
CLUSTER_NAME
-----
IOT_NAME
-----
STATUS      PCT_FREE  PCT_USED  INI_TRANS  MAX_TRANS  INITIAL_EXTENT  NEXT_EXTENT
-----
IN_EXTENTS  MAX_EXTENTS  PCT_INCREASE  FREELISTS  FREELIST_GROUPS  LOG_B  NUM_ROWS
-----
BLOCKS  EMPTY_BLOCKS  AVG_SPACE  CHAIN_CNT  AVG_ROW_LEN
-----
AVG_SPACE  FREELIST_BLOCKS  NUM_FREELIST_BLOCKS
-----
DEGREE
-----
INSTANCES
-----
SAMPLE_SIZE  LAST_ANAL  PAR  IOT_TYPE  T  S  NES  BUFFER  FLASH_C  CELL_FL  ROW_MOVE
-----
ELO  USE  DURATION  SKIP_COR  MON
-----
CLUSTER_OWNER
-----
DEPENDEN  COMPRESS  COMPRESS_FOR  DRO  REA  SEG  RESULT_  CLU
-----
CTIVITY_TRACKING  DML_TIMESTAMP  HAS  CON  INMEMORY  INMEMORY
-----
INMEMORY_DISTRI  INMEMORY_COMPRESS  INMEMORY_DUPL
-----
DEFAULT_COLLATION
-----
S  EXT  CELLMEMORY  CON  CON  EXT  EXT  INMEMORY_SER
-----
INMEMORY_SERVICE_NAME
-----
CON
-----

572 rows selected.
SQL> SQL> SQL> SP2-0042: unknown command "sadv" - rest of line ignored.
SQL> SQL>

```

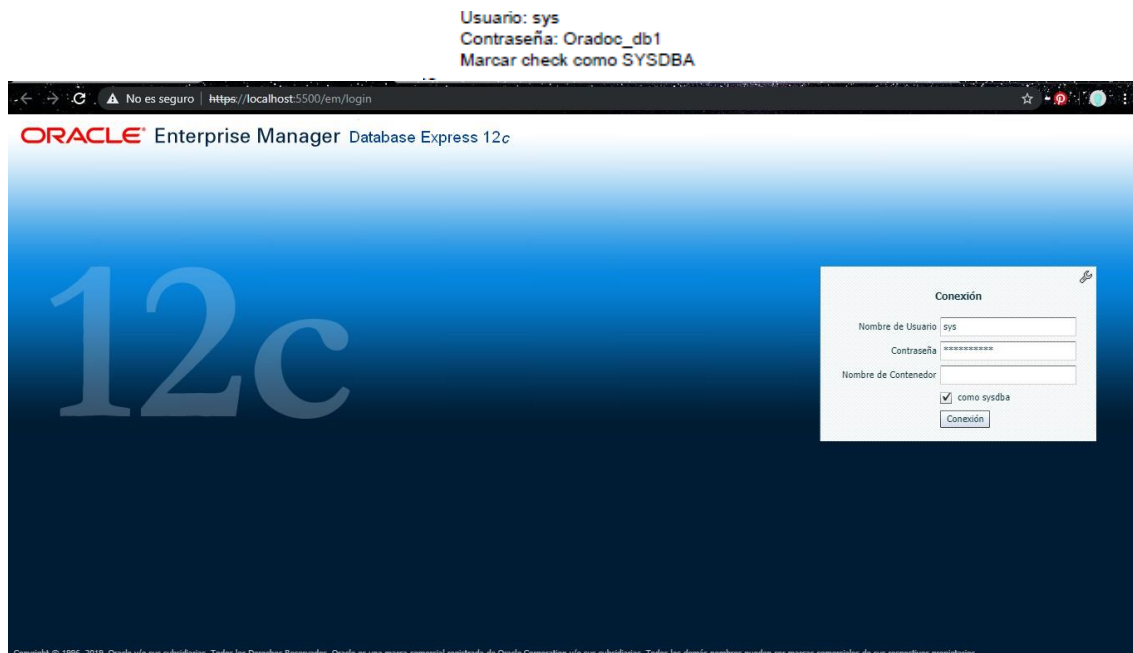
- Escribir el comando quit para cerrar la sesión de SQL\*Plus

```

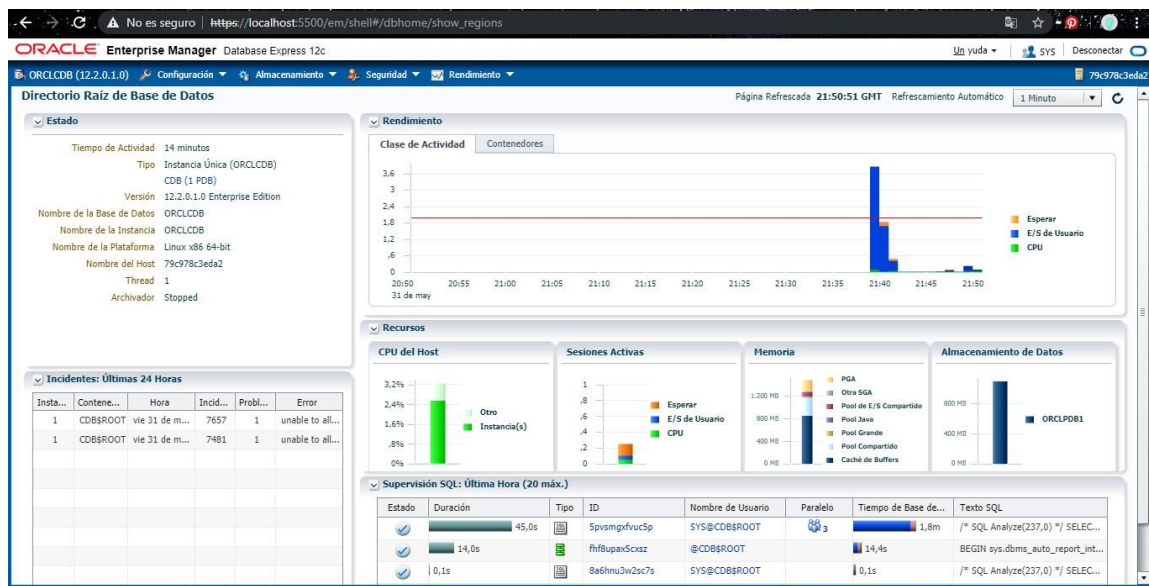
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
PS C:\>

```

- En una pestaña nueva del navegador de internet acceder a la siguiente dirección:<https://localhost:5500/e>  
Iniciar sesión con los siguientes datos:

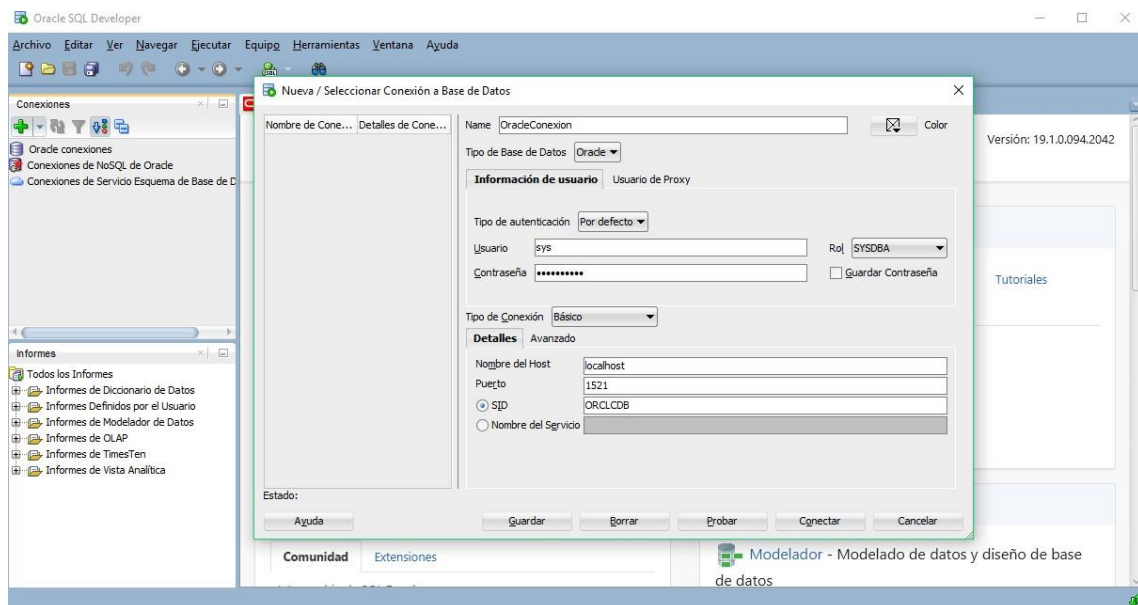


- Luego se visualizará la siguiente ventana. Cerrar sesión y la pestaña del navegador de internet.



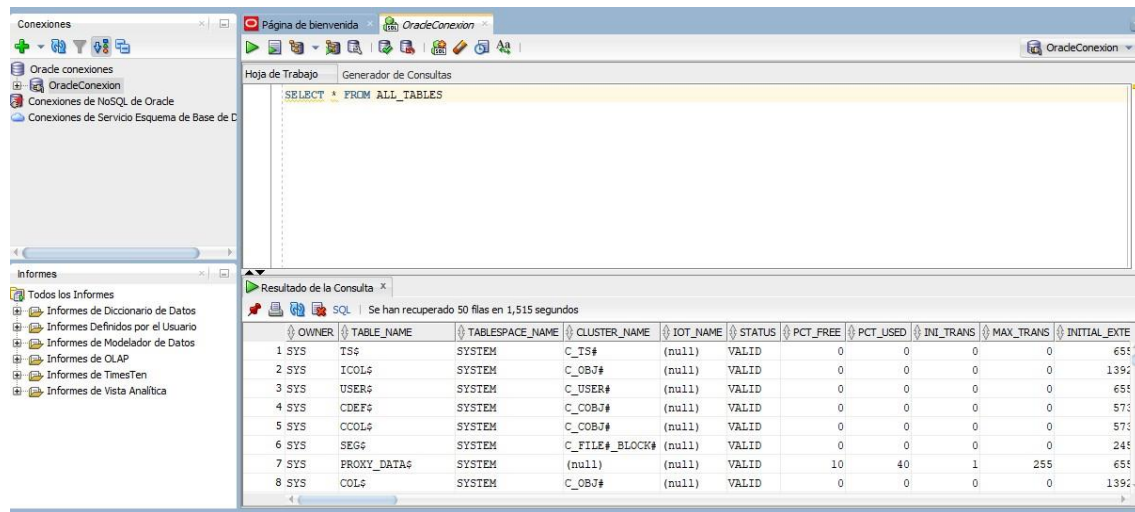
- Iniciar el aplicativo Oracle SQL Developer, crear una nueva conexión con los siguientes parámetros:

Name : OracleConexion  
 Usuario: sys  
 Contraseña: Oradoc\_db1  
 Rol: SYSDBA  
 Nombre del Host: localhost  
 Puerto: 1521  
 SID: ORCLCDB



- Iniciar una nueva consulta, escribir y ejecutar lo siguiente; deberá retornar varios registros

que representan las tablas de las base de datos



- Cerrar la aplicación Oracle SQL Developer
- En PowerShell ejecutar el siguiente comando. Y verificar la eliminación del contenedor con ejecutando

```
PS C:\> docker rm -f ORACLEDB01
ORACLEDB01
PS C:\> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS
PORTS
NAMES
```

### 3.3. Parte 3: Adicionando persistencia

- En PowerShell ejecutar el siguiente comando, lo cual dara como respuesta se visualizará un ID que corresponde al contenedor

```
PS C:\> docker run -d -it --name ORACLEDB01 -p 1521:1521 -p 5500:5500 -v D:\ORCL:\ORCL store/oracle/database-enterprise:12.2.0.1
8f80d3a62c7497bcd62d9216d7312551557611ec52e5f95f429b926af99e5312
```

- Repetir el paso 13 y modificar la contraseña del usuario SYS

```
PS C:\> docker exec -it ORACLEDB01 bash -c "source /home/oracle/.bashrc; sqlplus / as sysdba"

SQL*Plus: Release 12.2.0.1.0 Production on Fri Jun 7 20:00:26 2019

Copyright (c) 1982, 2016, Oracle. All rights reserved.

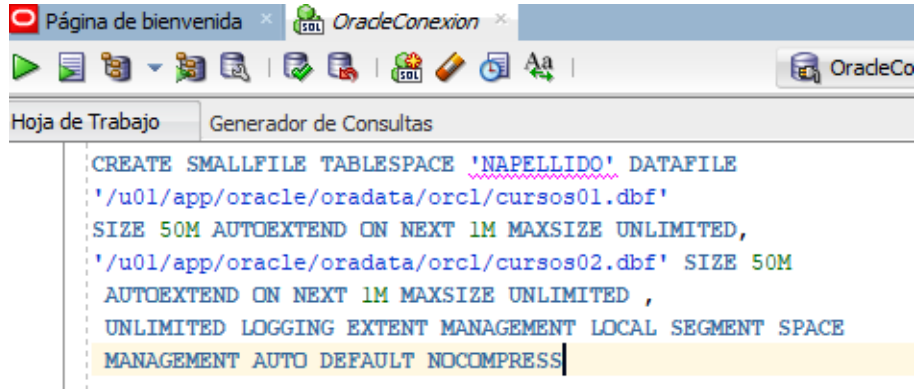
Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL> alter user sys identified by 123;

User altered.

SQL> quit
Disconnected from Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
PS C:\>
```

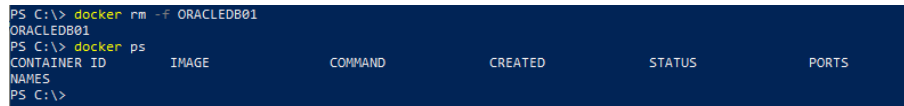
- Iniciar el aplicativo Oracle SQL Developer, conectarse como el usuario SYS y ejecutar el siguiente comando



The screenshot shows the Oracle SQL Developer interface. The 'Generador de Consultas' (Query Generator) tab is active, displaying a SQL script to create a smallfile tablespace named 'NAPELLIDO'. The script is as follows:

```
CREATE SMALLFILE TABLESPACE 'NAPELLIDO' DATAFILE
'/u01/app/oracle/oradata/orcl/cursos01.dbf'
SIZE 50M AUTOEXTEND ON NEXT 1M MAXSIZE UNLIMITED,
'/u01/app/oracle/oradata/orcl/cursos02.dbf' SIZE 50M
AUTOEXTEND ON NEXT 1M MAXSIZE UNLIMITED ,
UNLIMITED LOGGING EXTENT MANAGEMENT LOCAL SEGMENT SPACE
MANAGEMENT AUTO DEFAULT NOCOMPRESS
```

- Verificar el contenido de la carpeta ORCL
- En PowerShell ejecutar el siguiente comando. Verificar la eliminación del contenedor con ejecutando



The screenshot shows a PowerShell terminal with the following commands and output:

```
PS C:\> docker rm -f ORACLEDB01
ORACLEDB01
PS C:\> docker ps
```

The output of the `docker ps` command is a table with the following columns: CONTAINER ID, IMAGE, COMMAND, CREATED, STATUS, and PORTS.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
NAMES					

- Cerrar la aplicación Oracle SQL Developer.

## 4. ANALISIS E INTERPRETACION DE RESULTADOS

### 4.1. Parte 1: Actividades Encargadas

- ¿Con qué comando(s) puedo iniciar y detener una instancia de Oracle, detalle cada uno de los pasos y opciones, utilizando Docker?
- Para poder Iniciar una instancia de Oracle Database Server y también iniciar una instancia de servidor de base de datos Oracle al ejecutar "docker run -d -it --name oracle-db store/oracle/database-enterprise:12.2.0.1" donde oracle01-db está el nombre del contenedor y 12.2.0.1 es la etiqueta de imagen de Docker.
- Los comandos docker ps -a -q detendrá todos los contenedores Docker en ejecución :

```
PS C:\> docker ps -a
```

CONTAINER ID	IMAGE	COMMAND NAMES	CREATED
ae51e85914b9	store/oracle/database-enterprise:12.2.0.1	"/bin/sh -c '/bin/ba..."	13 minutes ago
es (healthy)	1521/tcp, 5500/tcp	oracle01-db	
e84e7aa3262c	store/oracle/database-enterprise:12.2.0.1	"/bin/sh -c '/bin/ba..."	46 hours ago
14 minutes ago	0.0.0.0:1521->1521/tcp, 0.0.0.0:5500->5500/tcp	ORACLEDB02	
e4227989c407	store/oracle/database-enterprise:12.2.0.1	"/bin/sh -c '/bin/ba..."	2 days ago
		oracle-db	
25a413f30f4f	store/oracle/database-enterprise:12.2.0.1	"/bin/sh -c '/bin/ba..."	3 days ago
		ORACLEDB01	

- ¿Con qué comando(s) puedo iniciar y detener el Listener y el Enterprise manager, detalle cada uno de los pasos y opciones, utilizando Docker?
- El comando para iniciar el listener es : "lsnrctl start" para detener "lsnrctl stop"

```

core@ip-172-30-4-75 ~ $ docker exec -it orclcdb bash
[oracle@5240ac3f3479 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 14-MAY-2017 02:36:24

Copyright (c) 1991, 2016, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC)))
STATUS of the LISTENER
-----
Alias                     LISTENER
Version                   TNSLSNR for Linux: Version 12.2.0.1.0 - Production
Start Date                14-MAY-2017 02:21:33
Uptime                    0 days 0 hr. 14 min. 50 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /opt/oracle/product/12.2.0.1/dbhome_1/network/admin/li
stener.ora
Listener Log File         /opt/oracle/diag/tnslsnr/5240ac3f3479/listener/alert/l
og.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=0.0.0.0)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=5240ac3f3479)(PORT=5500))(Security=
(my_wallet_directory=/opt/oracle/admin/ORCLCDB/xdw_wallet))(Presentation=HTTP)(S
ession=RAW))
Services Summary...
Service "4f73af0b1ee50737e053020011acf8ce" has 1 instance(s).
  Instance "ORCLCDB", status READY, has 1 handler(s) for this service...
Service "ORCLCDB" has 1 instance(s).
  Instance "ORCLCDB", status READY, has 1 handler(s) for this service...

PS C:\> docker exec -it ORACLEDB05 bash -c "source /home/oracle/.bashrc; sqlplus / as sysdba"

SQL*Plus: Release 12.2.0.1.0 Production on Sun Jun 9 04:23:59 2019

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

```

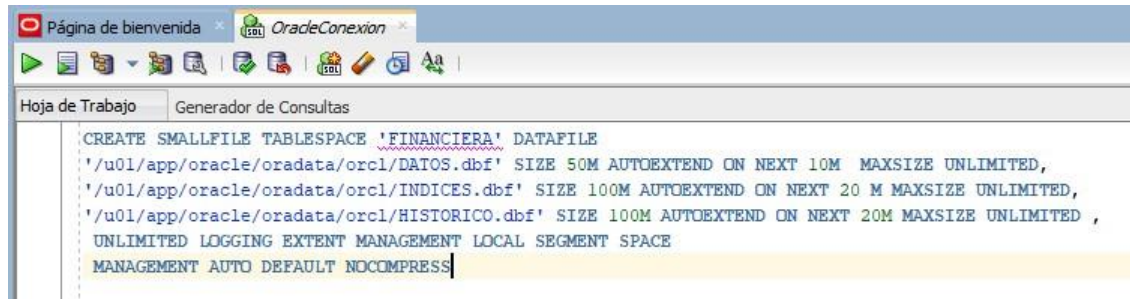
- Genere un nuevo contenedor y cree un espacio de tablas con las siguientes características.



Nombre : FINANCIERA:

- DATOS (dbf) : Tamaño Inicial : 50MB, Incremento: 10MB, Ilimitado
- INDICES (dbf) Tamaño Inicial : 100MB, Incremento: 20MB, Maximo: 1GB
- HISTORICO (dbf) Tamaño Inicial : 100MB, Incremento: 50MB, Ilimitado

¿Cuál sería el script SQL que generaría esta base de datos?



The screenshot shows the Oracle SQL Developer interface. The top toolbar includes icons for running, saving, and editing. The 'Generador de Consultas' (Query Generator) window is active, displaying a SQL script to create a tablespace named 'FINANCIERA'. The script specifies three datafiles: 'DATOS.dbf' (50MB initial, 10MB increment, unlimited maxsize), 'INDICES.dbf' (100MB initial, 20MB increment, 1GB maxsize), and 'HISTORICO.dbf' (100MB initial, 50MB increment, unlimited maxsize). The tablespace is configured with unlimited logging, local segment space management, and no compression.

```
CREATE SMALLFILE TABLESPACE 'FINANCIERA' DATAFILE
'/u01/app/oracle/oradata/orcl/DATOS.dbf' SIZE 50M AUTOEXTEND ON NEXT 10M MAXSIZE UNLIMITED,
'/u01/app/oracle/oradata/orcl/INDICES.dbf' SIZE 100M AUTOEXTEND ON NEXT 20 M MAXSIZE UNLIMITED,
'/u01/app/oracle/oradata/orcl/HISTORICO.dbf' SIZE 100M AUTOEXTEND ON NEXT 20M MAXSIZE UNLIMITED ,
UNLIMITED LOGGING EXTENT MANAGEMENT LOCAL SEGMENT SPACE
MANAGEMENT AUTO DEFAULT NOCOMPRESS;
```

## 5. CONCLUSIONES

En el desafío Docker hemos visto de manear clara cómo Docker y sus contenedores están transformando las operaciones de los DevOps como una herramienta importante en el mundo del desarrollo web. Los casos de uso para los contenedores de Docker dentro del ámbito de DevOps son muchos, ejecutar aplicaciones en un contenedor y luego desplegar en cualquier lugar (la nube o en las instalaciones o cualquier distribución de Linux) es una realidad ahora.

Sin dudas Docker ha aportado mucho al mundo de la virtualización, sin embargo dudamos que al menos en el corto plazo los usuarios dejen de usar soluciones de virtualización tradicionales como vmware, openvz, etc.

Algunas características:

- Instalación simple y capacidad de ejecutar múltiples aplicaciones en entornos aislados sobre un mismo sistema operativo, permitiéndonos ahorrar horas de trabajo en la administración de Infraestructura.
- Independiente a la plataforma, permite contar con soluciones más portables.
- Despliegue de Aplicaciones mucho más rápida y flexible.
- Disponible en múltiples proveedores de Nube.



## 6. REFERENCIAS

- [ 1 ] Hat, R. (2017). ¿Qué es Docker?. Recuperado de <https://www.redhat.com/es/topics/containers/what-is-docker>
- [ 2 ] código chido. (2019). Docker Oracle. Recuperado de <https://codigochido.com/post/2019-01-21-docker-oracle/>
- [ 3 ] Nelson, C. (2018). Usando Oracle 12c en Docker sobre Windows 10. Recuperado de <https://www.docker-win10-4485487-esa.html>
- [ 4 ] The ORACLE-BASE Blog. (2018). Oracle Database en Docker. Recuperado de <https://oracle-base.com/articles/linux/docker-oracle-database-on-docker>