

UNIVERSIDAD PRIVADA-DE-TACNA



INGENIERIA DE SISTEMAS

TITULO:

**INFORME DE LABORATORIO N° 04**

**CURSO:**

BASE DE DATOS II

**DOCENTE(ING):**

Patrick Cuadros Quiroga

Integrantes:

Escalante Alanoca, Jesus Humberto

(2015050641)

# Índice

<b>1. INFORMACION GENERAL</b>	<b>1</b>
1.1. Objetivos:	1
1.2. Requerimientos	1
<b>2. MARCO TEORICO</b>	<b>1</b>
<b>3. PROCEDIMIENTO</b>	<b>2</b>
3.1. Instalacion de Docker	2
3.2. Iniciando en Docker	4
3.3. Creando un contenedor Microsoft SQL para Linux	5
3.4. Adicionando una Persistencia	11
3.5. Creando un contenedor con Microsoft SQL para Windows	12
3.6. Actividades Encargadas	18
3.6.1. ¿Con qué comando(s) exportaría la imagen de Docker de Microsoft SQLServer a otra PC o servidor?	18
3.6.2. ¿Con qué comando(s) podría generar dos volúmenes para un contenedor?	19
3.6.3. Genere un nuevo contenedor con las siguientes características:	20
<b>4. CUESTIONARIO</b>	<b>21</b>
<b>5. CONCLUSIONES</b>	<b>22</b>
<b>6. BIBLIOGRAFIA</b>	<b>23</b>
<b>7. WEBGRAFIA</b>	<b>24</b>

- |   |    |
|---|----|
| 4.1. ¿Con qué comando(s) exportaría la imagen de Docker de Microsoft SQL Server a otra PC o servidor? .....   | 21 |
| 4.2. ¿Con qué comando(s) podría generar dos volúmenes para un contenedor para distribuir en un volumen el Archivo de Datos (mdf) y en otro el Archivo Log (ldf)? .. | 21 |
| 4.3. Genere un nuevo contenedor y cree la base de datos con las siguientes características  | 21 |

# **1. INFORMACION GENERAL´**

## **1.1. Objetivos:**

- Aprender sobre como iniciar la Instalacion de una Instancia de Microsoft SQL

## **1.2. Requerimientos**

### **Conocimientos**

Para el desarrollo de esta pr´actica se requerir´a de los siguientes conocimientos ba´sicos:

- Conocimientos ba´sicos de administraci3n de base de datos Microsoft SQL Server.
- Conocimientos ba´sicos de SQL.

### **Hardware**

- Virtualization activada en el BIOS..
- CPU SLAT-capable feature.
- Al menos 4GB de RAM.

### **Software**

Asimismo se necesita los siguientes aplicativos:

- Windows 10 64bit: Pro, Enterprise o Education (1607 Anniversary Update, Build 14393 o Superior)
- Docker Desktop (Para lo cual se debe primero crear una cuenta en Docker Hub - <https://hub.docker.com/s>)
- Microsoft SQL Server Management Studio en su u´ltima versio´n

# **2. MARCO TEORICO**

Una instancia de Motor de base de datos es una copia del ejecutable de sqlservr.exe que se ejecuta como un servicio de sistema operativo. Cada instancia administra varias bases de datos del sistema y una o varias bases de datos de usuario. Cada equipo puede ejecutar varias instancias de Motor de base de datos. Las aplicaciones se conectan a la instancia para realizar el trabajo en una base de datos administrada por la instancia.

## **Instancias**

Una instancia de Motor de base de datos funciona como un servicio que controla todas las solicitudes de aplicación para trabajar con datos de cualquiera de las bases de datos administradas por dicha instancia. Es el destino de las solicitudes de conexión (inicios de sesión) de aplicaciones. La conexión se ejecuta en una conexión de red si la aplicación y la instancia están en equipos independientes. Si la aplicación y la instancia están en el mismo equipo, la conexión de SQL Server se puede ejecutar como una conexión de red o una conexión en memoria. Cuando una conexión se ha completado, una aplicación envía instrucciones Transact-SQL a través de la conexión hasta la instancia. La instancia resuelve las instrucciones de Transact-SQL en operaciones con los datos y objetos de las bases de datos y, si se han concedido los permisos necesarios a las credenciales de inicio de sesión, realiza el trabajo. Los datos recuperados se devuelven a la aplicación, junto con cualesquiera mensajes como errores. Puede ejecutar múltiples instancias de Motor de base de datos en un equipo. Una instancia puede ser la instancia predeterminada. La instancia predeterminada no tiene nombre. Si una solicitud de conexión especifica solo el nombre del equipo, se establece la conexión a la instancia predeterminada. Una instancia con nombre es una instancia en la que se especifica un nombre de instancia al instalar la instancia. Una solicitud de conexión debe especificar el nombre del equipo y el nombre de instancia para conectar a la instancia. No hay ningún requisito para instalar una instancia predeterminada; todas las instancias que se ejecutan en un equipo pueden ser instancias con nombre.

## **3. PROCEDIMIENTO**

### **3.1. Instalacion de Docker**

– Instalar Docker desde la siguiente direccion : <https://docs.docker.com/docker-for-windows/install/>



## Docker Desktop for Windows

By Docker

The fastest and easiest way to get started with Docker on Windows

Edition

Windows

x86-64

### Get Docker Desktop for Windows

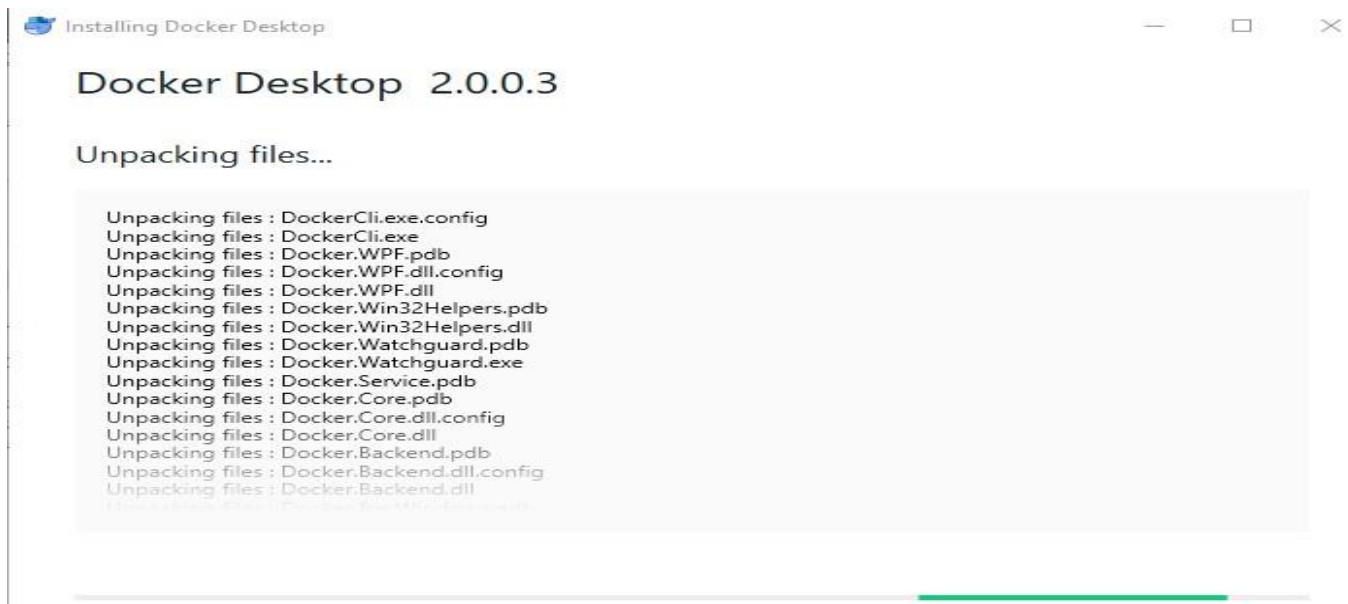
Docker Desktop for Windows is available for free.

Requires Microsoft Windows 10 Professional or Enterprise 64-bit. For previous versions get Docker Toolbox.

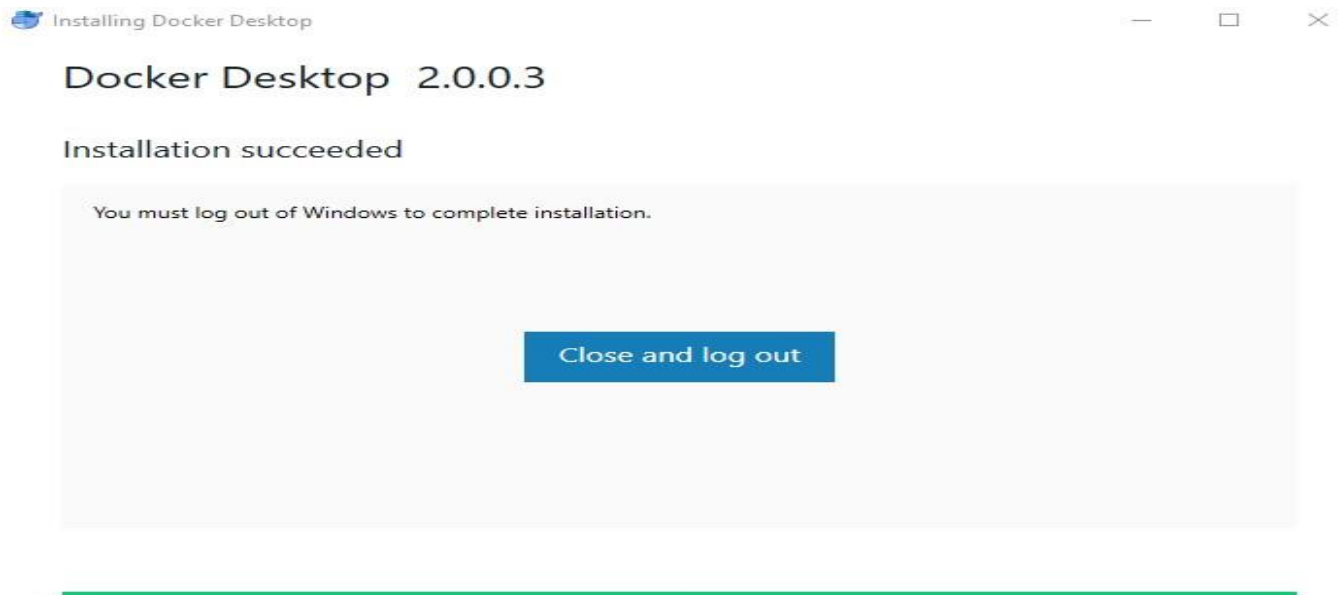
By downloading this, you agree to the terms of the [Docker Software End User License Agreement](#) and the [Docker Data Processing Agreement \(DPA\)](#).

↓ Get Docker

– Seguir el proceso de Instalacion :



– Reiniciar la PC :



- Comprobar que docker ha sido instalado correctamente:

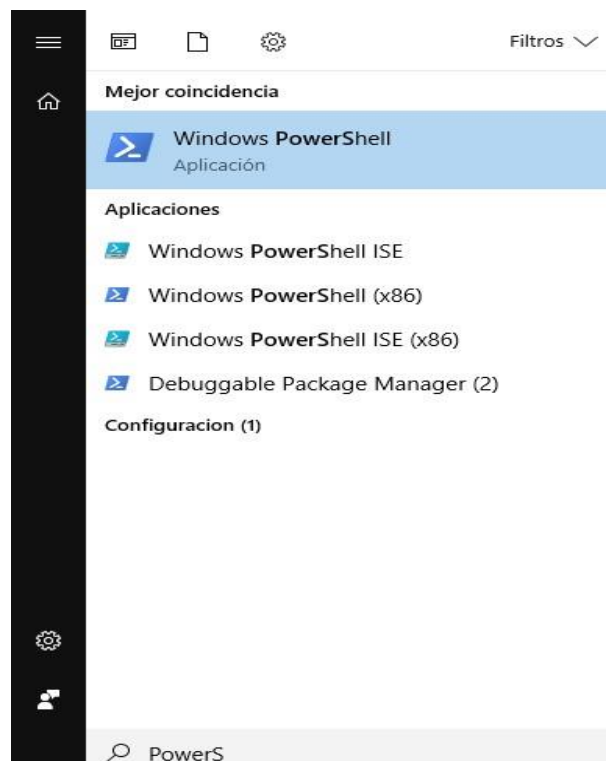


### 3.2. Iniciando en Docker

- Logearse con su cuenta y contraseña Respectiva:



– Iniciar la consola PowerShell de Windows:



### 3.3. Creando un contenedor Microsoft SQL para Linux

– Como primer comando usaremos: "docker version" para ver la version de docker que acabamos de instalar:

```

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

PS C:\Users\Usuario> 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 1.12)
  Go version:   go1.10.6
  Git commit:   6247962
  Built:        Sun Feb 10 04:13:06 2019
  OS/Arch:      linux/amd64
  Experimental: false
PS C:\Users\Usuario>

```

- Ahora crearemos un contenedor con Microsoft SQL server para Linux, para esto usaremos primero el comando "docker search mssql":

```

Windows PowerShell
PS C:\Users\Usuario> docker search mssql

```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
microsoft/mssql-server-linux	Deprecated SQL Server on Linux Container Rep...	1122		
microsoft/mssql-server-windows-developer	Official Microsoft SQL Server Developer Edit...	315		
microsoft/mssql-server-windows-express	Official Microsoft SQL Server Express Editio...	300		
microsoft/mssql-tools	Official images for Microsoft SQL Server Com...	51		
rsmoorthy/mssql	MSSQL Database (version SQL2000)	11		[OK]
datagrip/mssql-server-linux	SQL Server and SQL Server tools on Linux(201...	9		[OK]
gantrion/mssql-server-2014-express-windows-with-iis	mssql 2014 + IIS	4		
microsoft/mssql-monitoring-influxdb	Sample Image for Influxdb, This image is des...	3		
tsgkadot/mssql-tools	SQL Server tools on Linux (sqlcmd)	3		[OK]
jboesi/mssql-server-linux	mssql-server-linux with mssql-tools installe...	2		[OK]
mcroe/mssqldocker	Builds on microsoft/mssql-server-linux and a...	2		[OK]
microsoft/mssql-monitoring-collectd	This Sample image is designed to work with t...	1		
awaragi/prometheus-mssql-exporter	prometheus-mssql-exporter	1		[OK]
mondora/sandman2-mssql	Docker image for running sandman2 to get a R...	0		[OK]
ansibleplaybookbundle/mssql-apb	MS SQL Server on Linux (APB)	0		[OK]
r2dbc/r2dbc-mssql		0		
bitwarden/mssql	The Bitwarden database.	0		
tchughesiv/mssql-server-linux	CentOS build	0		[OK]
ansibleplaybookbundle/mssql-remote-apb	An APB that deploys Microsoft SQL Server	0		[OK]
softwareplant/mssql	SQL Server test database	0		[OK]
mileiq/ubuntu16-python3-mssql-kafka	Base image built on top of mileiq/ubuntu16-p...	0		
ncia/anet-mssql-linux	Container image for running a mssql database...	0		
langdon/fedora-mssqlserver	Microsoft SQL Server running on Fedora. You ...	0		[OK]
liaisonintl/mssql-server-linux	mssql-server-linux	0		[OK]
astronomerio/mssql-source	MSSQL source.	0		[OK]

```

PS C:\Users\Usuario>

```

- Luego descargaremos la imagen del contenedor de Microsoft SQL en un servidor Linux con el siguiente comando "docker pull microsoft/mssql-server-linux":



Windows PowerShell

```
PS C:\Users\Usuario> docker pull microsoft/mssql-server-linux
Using default tag: latest
latest: Pulling from microsoft/mssql-server-linux
59ab41dd721a: Downloading [====>] 3.439MB/42.22MB
57da90bec92c: Download complete
06fe57530625: Download complete
5a6315cba1ff: Downloading [=====>] 513B/862B
739f58768b3f: Download complete
0b751601bca3: Waiting
bcf04a22644a: Waiting
6b5009e4f470: Waiting
a9dca2f6722a: Waiting
```

– esperar un determinado tiempo a que descargue todo:

Windows PowerShell

```
PS C:\Users\Usuario> docker pull microsoft/mssql-server-linux
Using default tag: latest
latest: Pulling from microsoft/mssql-server-linux
59ab41dd721a: Pull complete
57da90bec92c: Pull complete
06fe57530625: Pull complete
5a6315cba1ff: Pull complete
739f58768b3f: Pull complete
0b751601bca3: Pull complete
bcf04a22644a: Pull complete
6b5009e4f470: Pull complete
a9dca2f6722a: Pull complete
Digest: sha256:9b700672670bb3db4b212e8aef841ca79eb2fce7d5975a5ce35b7129a9b90ec0
Status: Downloaded newer image for microsoft/mssql-server-linux:latest
PS C:\Users\Usuario>
```

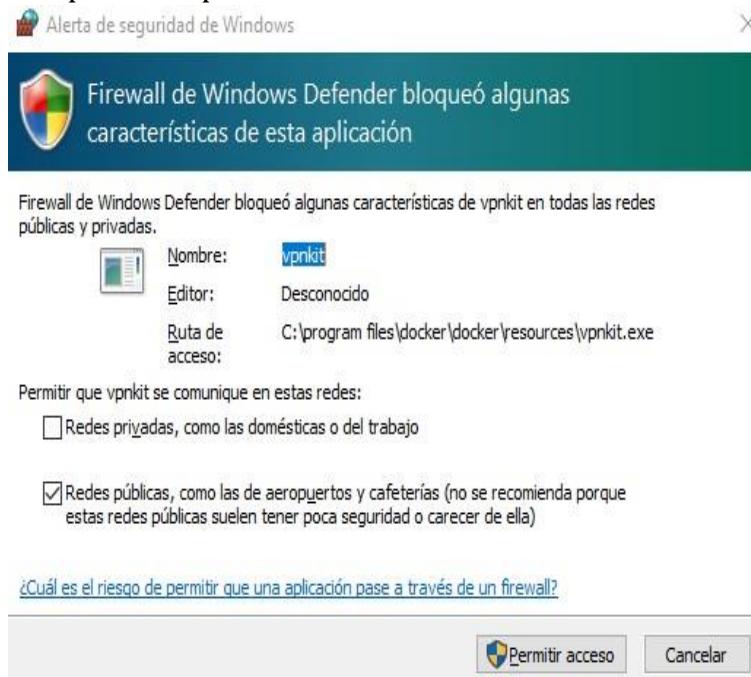
Para ver la imagen que acabamos de descargar, usaremos el siguiente comando: "docker images":

```
Windows PowerShell
PS C:\Users\Usuario> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
microsoft/mssql-server-linux  latest             314918ddaedf       5 months ago       1.35GB
PS C:\Users\Usuario>
```

Ahora crearemos credenciales los cuales usaremos mas adelante para autenticar nuestra entrada a SQL server, usaremos el siguiente comando:

```
Windows PowerShell
PS C:\Users\Usuario> docker run -d -p 16111:1433 -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=Tacna.2019' --name SQLNX01 microsoft/mssql-server-linux
64108766dc344a30ae93e0f7785737d718a8bb334f603ac590bd214df23772f7
PS C:\Users\Usuario>
```

Accedemos a dar los permisos para el firewall de Windows



Verificamos la correcta ejecucion del contenedor con el comando "docker ps":

```
Windows PowerShell
PS C:\Users\Usuario> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS                    NAMES
64108766dc34   microsoft/mssql-server-linux        "/opt/mssql/bin/sqls... 7 minutes ago  Up 7 minutes  0.0.0.0:16111->1433/tcp  SQLNX01
PS C:\Users\Usuario>
```

– Accedemos a Sql server con los siguientes credenciales:

Connect to Server

SQL Server

Server type: Database Engine

Server name: (local),16111

Authentication: SQL Server Authentication

Login: sa

Password: \*\*\*\*\*

☐ Remember password

Connect Cancel Help Options >>

– En sql iniciaremos un nuevo query para hacer una consulta sobre la version:



Ahora cerraremos Sql server y procederemos a eliminar el contenedor creado con el siguiente comando: "docker rm -f SQLLN01" despues comprobaremos que este ha sido eliminado:

Windows PowerShell

```
PS C:\Users\Usuario> docker rm -f SQLLN01
```

```
SQLLN01
```

```
PS C:\Users\Usuario> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
PS C:\Users\Usuario> █						

### 3.4. Adicionando una Persistencia

- Crearemos un nuevo contenedor, verificaremos que este ha sido creado correctamente y luego iniciaremos sesion con los respectivos credenciales:

```
Seleccionar Windows PowerShell
PS C:\Users\Usuario> docker run -d -p 16111:1433 -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=Tacna.2019' -v F:\DATA\NX\var/opt/mssql --name SQLLNX02 microsoft/mssql-server-linux
51ffc1db6e635d097766ce8c7deac1185050cbd4ab163707f7dffc607c862d6e
PS C:\Users\Usuario>
```

```
Windows PowerShell
PS C:\Users\Usuario> docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
51ffc1db6e63        microsoft/mssql-server-linux   "/opt/mssql/bin/sqls..."   About a minute ago   Up About a minute   0.0.0.0:16111->1433/tcp   SQLLNX02
PS C:\Users\Usuario>
```

Connect to Server

SQL Server

Server type: Database Engine

Server name: (local),16111

Authentication: SQL Server Authentication

Login: sa

Password: \*\*\*\*\*

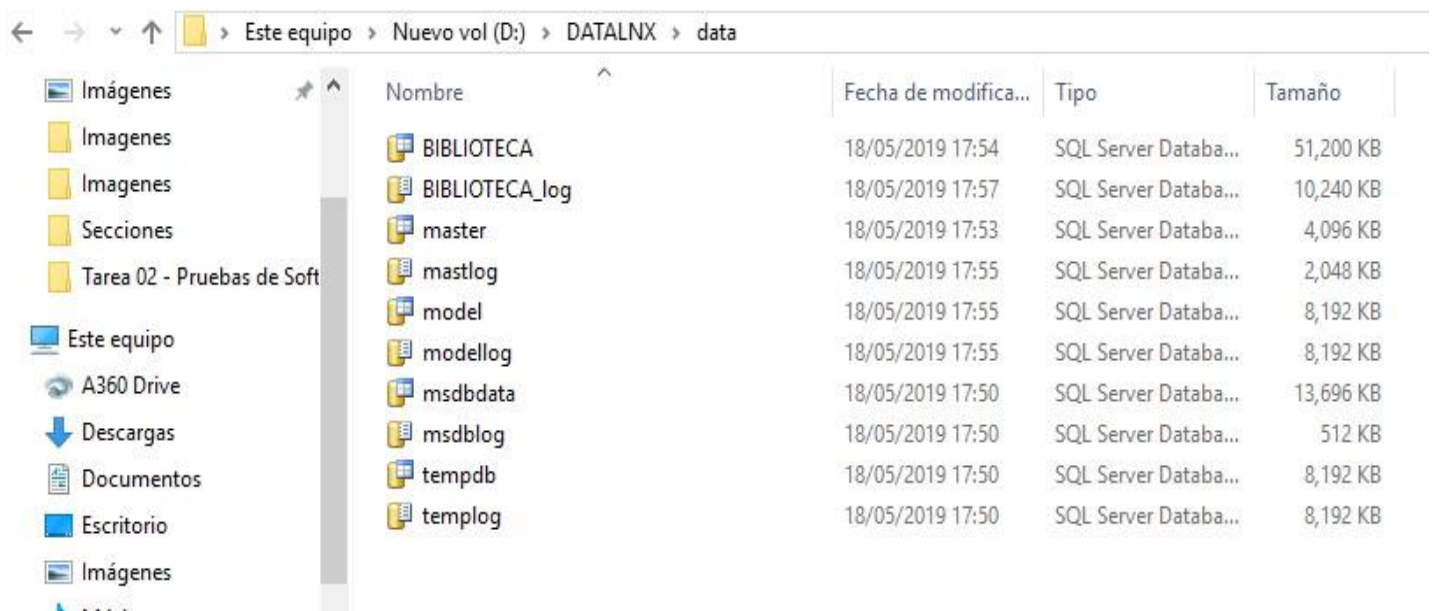
☐ Remember password

Connect Cancel Help Options >>

Ahora crearemos una base de datos con el siguiente Script:

```
SQLQuery1.sql - (lo...111.master (sa (52))*) X
CREATE DATABASE BIBLIOTECA ON
PRIMARY (
    NAME = N'BIBLIOTECA',
    FILENAME = N'/var/opt/mssql/data/BIBLIOTECA.mdf',
    SIZE = 50MB ,
    FILEGROWTH = 10240KB
) LOG ON (
    NAME = N'BIBLIOTECA_log',
    FILENAME = N'/var/opt/mssql/data/BIBLIOTECA_log.ldf',
    SIZE = 10MB ,
    FILEGROWTH = 5MB
)
GO
```

– Verificaremos que la carpeta DATALNX contenga esta base de datos:



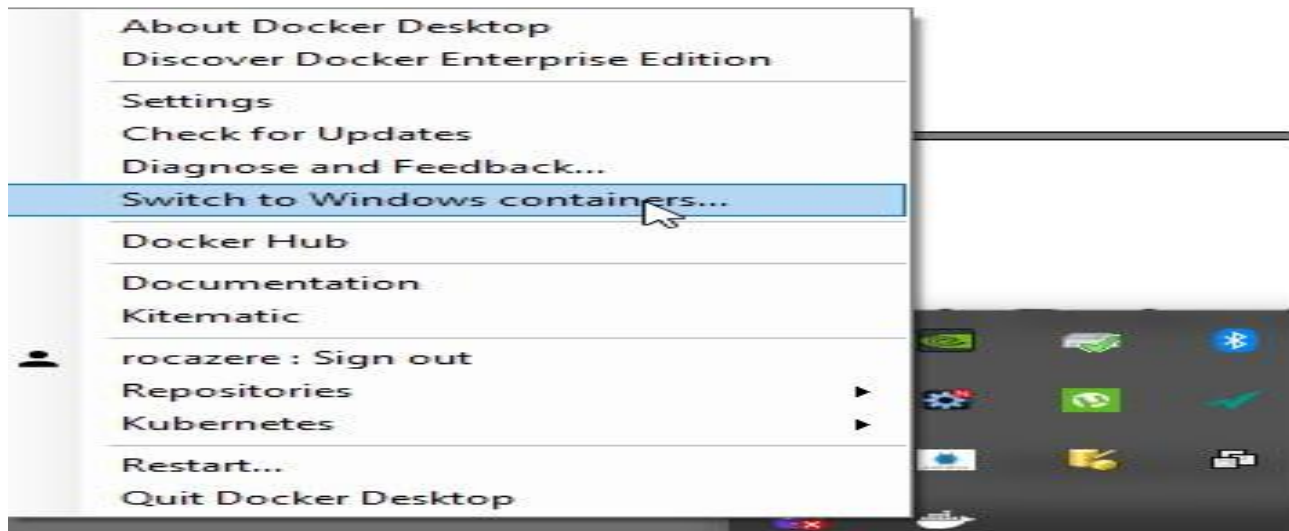
Nombre	Fecha de modifica...	Tipo	Tamaño
BIBLIOTECA	18/05/2019 17:54	SQL Server Databa...	51,200 KB
BIBLIOTECA_log	18/05/2019 17:57	SQL Server Databa...	10,240 KB
master	18/05/2019 17:53	SQL Server Databa...	4,096 KB
mastlog	18/05/2019 17:55	SQL Server Databa...	2,048 KB
model	18/05/2019 17:55	SQL Server Databa...	8,192 KB
modellog	18/05/2019 17:55	SQL Server Databa...	8,192 KB
msdbdata	18/05/2019 17:50	SQL Server Databa...	13,696 KB
msdblog	18/05/2019 17:50	SQL Server Databa...	512 KB
tempdb	18/05/2019 17:50	SQL Server Databa...	8,192 KB
templog	18/05/2019 17:50	SQL Server Databa...	8,192 KB

– Por ultimo eliminaremos este contenedor.

### 3.5. Creando un contenedor con Microsoft SQL para Windows

– En la parte inferior derecha encontraremos el icono de Docker el cual al hacerle click derecho, abrira un menu desplegable en el que seleccionaremos Switch to windows containers... y esperaremos a que docker se reinicie:





– Ahora en la ventana de PowerShell usaremos los siguientes comandos::

```
Windows PowerShell
PS C:\Users\Usuario> docker search mssql
```

NAME	DESCRIPTION	STARS	OFFICIAL	AUTOMATED
microsoft/mssql-server-linux	Deprecated SQL Server on Linux Container Rep...	1122		
microsoft/mssql-server-windows-developer	Official Microsoft SQL Server Developer Edit...	315		
microsoft/mssql-server-windows-express	Official Microsoft SQL Server Express Editio...	300		
microsoft/mssql-tools	Official images for Microsoft SQL Server Com...	51		
rsmoorthy/mssql	MSSQL Database (version SQL2000)	11		[OK]
datagrip/mssql-server-linux	SQL Server and SQL Server tools on Linux(201...	9		[OK]
gantrior/mssql-server-2014-express-windows-with-iis	mssql 2014 + IIS	4		
microsoft/mssql-monitoring-influxdb	Sample Image for Influxdb, This image is des...	3		
tsgkadot/mssql-tools	SQL Server tools on Linux (sqlcmd)	3		[OK]
jboesl/mssql-server-linux	mssql-server-linux with mssql-tools installe...	2		[OK]
mcmoe/mssqldocker	Builds on microsoft/mssql-server-linux and a...	2		[OK]
microsoft/mssql-monitoring-collectd	This Sample image is designed to work with t...	1		
awaragi/prometheus-mssql-exporter	prometheus-mssql-exporter	1		[OK]
mondora/sandman2-mssql	Docker image for running sandman2 to get a R...	0		[OK]
ansibleplaybookbundle/mssql-apb	MS SQL Server on Linux (APB)	0		[OK]
r2dbc/r2dbc-mssql		0		
bitwarden/mssql	The Bitwarden database.	0		
tchughesiv/mssql-server-linux	CentOS build	0		[OK]
ansibleplaybookbundle/mssql-remote-apb	An APB that deploys Microsoft SQL Server	0		[OK]
softwareplant/mssql	SQL Server test database	0		[OK]
mileiq/ubuntu16-python3-mssql-kafka	Base image built on top of mileiq/ubuntu16-p...	0		
ncia/anet-mssql-linux	Container image for running a mssql database...	0		
langdon/fedora-mssqlserver	Microsoft SQL Server running on Fedora. You ...	0		[OK]
liaisonintl/mssql-server-linux	mssql-server-linux	0		[OK]
astronomerio/mssql-source	MSSQL source.	0		[OK]

```
PS C:\Users\Usuario>
```

Instalaremos el contenedor de Microsoft sql para un servidor Windows:

```
Windows PowerShell
PS C:\Users\Usuario> docker pull microsoft/mssql-server-windows-developer
Using default tag: latest
latest: Pulling from microsoft/mssql-server-windows-developer
3889bb8d808b: Pulling fs layer
449343c9d7e2: Pulling fs layer
08883151461d: Download complete
bafeb45a72fc: Download complete
f5c5aa235c5b: Waiting
158fead2ffa0: Waiting
746db9597cec: Waiting
9e96edbd8781: Waiting
c6dabab6234f: Waiting
975d0dccd859: Waiting
5b747cfb01b7: Waiting
c77992bbfd0f: Waiting
```

– Comprobaremos la correcta instalacion del contenedor con el comando "docker images":

```
Windows PowerShell
PS C:\Users\Usuario> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
microsoft/mssql-server-windows-developer	latest	19873f41b375	16 months ago	15.1GB

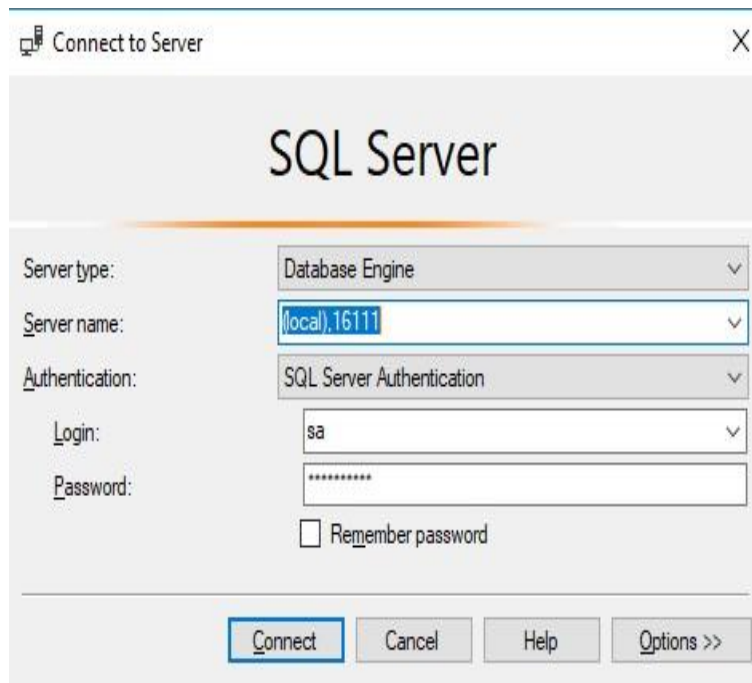
```
PS C:\Users\Usuario>
```



– Crearemos nuevas credenciales para este nuevo contenedor Sql para servidores windows:

```
Windows PowerShell
PS C:\Users\Usuario> docker run -d -p 16111:1433 -e 'ACCEPT_EULA=Y' -e 'SA_PASSWORD=Tacna.2019' -v D:\DATAWIN:C:\DATA --name SQLWIN01 microsoft/mssql-server-windows-developer
b1f6e0e7a00935f36caf90514383f92e7b8672aec266128f71fd7c07b5c91990
```

– Iniciaremos sesion en Sql con las credenciales que hemos creado:



revisamos la version:

```
SQLQuery1.sql - (lo...111.master (sa (52))) * X
Select @@VERSION

100 %
Results Messages
(No column name)
1 Microsoft SQL Server 2017 (RTM-CU3-GDR) (KB4052987) - 14.0.3015.40 (X64) Dec 22 2017 16:13:22 Copyright (C) 2017 Microsoft Corporation Developer Edition (64-bit) on Windows Server 2016 Datacenter 10.0 <X64> (Build 14393;) (Hypervisor)
```

– Mediante el siguiente scrip generaremos una base de datos de prueba:

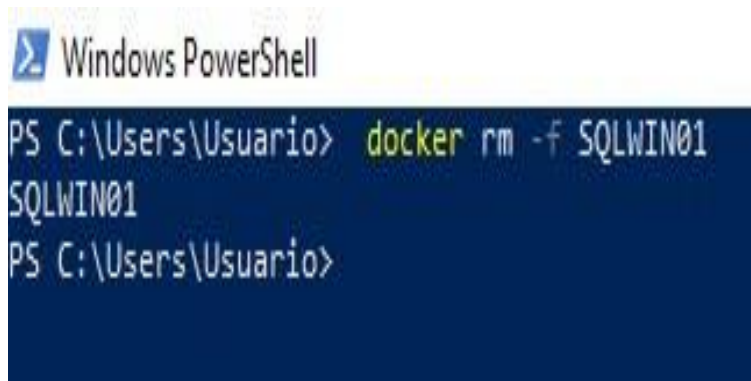
```
SQLQuery1.sql - (lo...111.master (sa (52))) * X
CREATE DATABASE BIBLIOTECA ON
PRIMARY (
    NAME = N'BIBLIOTECA',
    FILENAME = N'C:\DATA\BIBLIOTECA.mdf ',
    SIZE = 50MB ,
    FILEGROWTH = 10240KB
) LOG ON (
    NAME = N'BIBLIOTECA_log',
    FILENAME = N'C:\DATA\BIBLIOTECA_log.ldf',
    SIZE = 10MB ,
    FILEGROWTH = 5MB
)
GO
```

Comprobaremos que la base de datos ha sido creada:

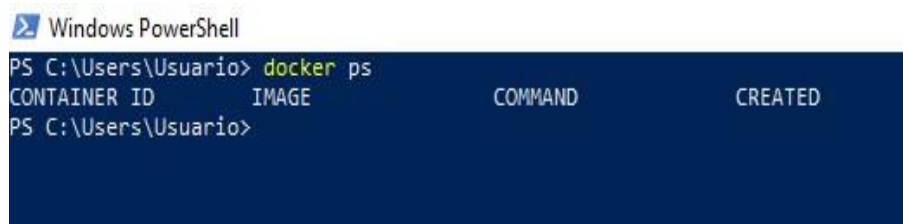


Este equipo > Nuevo vol (D:) > DATAWIN					
Nombre	Fecha de modifica...	Tipo	Tamaño		
BIBLIOTECA	19/05/2019 16:29	SQL Server Databa...	51,200 KB		
BIBLIOTECA_log	19/05/2019 16:29	SQL Server Databa...	10,240 KB		

Finalmente procederemos con la eliminacion del contenedor y verificaremos que esta ha sido eliminada:



```
Windows PowerShell
PS C:\Users\Usuario> docker rm -f SQLWIN01
SQLWIN01
PS C:\Users\Usuario>
```



```
Windows PowerShell
PS C:\Users\Usuario> docker ps
CONTAINER ID      IMAGE               COMMAND             CREATED
```

## 3.6. Actividades Encargadas

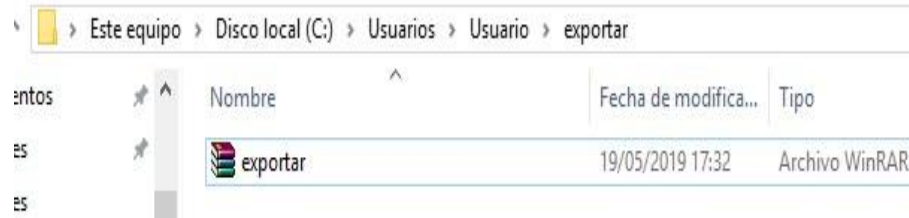
### 3.6.1. ¿Con qué comando(s) exportaría la imagen de Docker de Microsoft SQL Server a otra PC o servidor?

– uno de los comandos usados para exportar un contenedor sería:



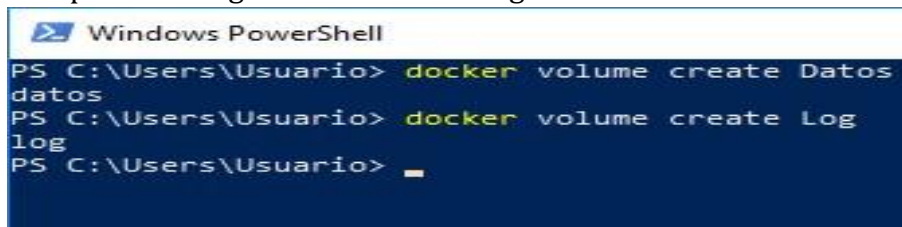
```
Windows PowerShell
PS C:\Users\Usuario> docker export SQLWIN01 > exportar/exportar.tar
```

Podemos observar que hemos guardado un archivo .tar en nuestra carpeta usuarios. Luego esto podrá ser transportado a otra máquina ya sea Windows o Linux.



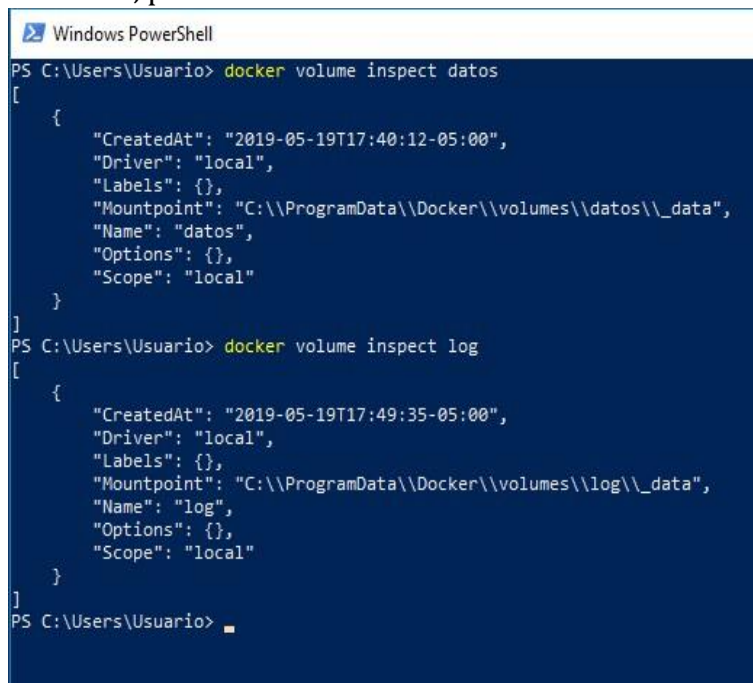
### 3.6.2. ¿Con qué comando(s) podríamos generar dos volúmenes para un contenedor?

- Los volúmenes pueden ser gestionados con el siguiente comando:



```
Windows PowerShell
PS C:\Users\Usuario> docker volume create Datos
datos
PS C:\Users\Usuario> docker volume create Log
log
PS C:\Users\Usuario> _
```

- Con el siguiente comando, podremos ver donde estos han sido creados:



```
Windows PowerShell
PS C:\Users\Usuario> docker volume inspect datos
[
  {
    "CreatedAt": "2019-05-19T17:40:12-05:00",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "C:\\ProgramData\\Docker\\volumes\\datos\\_data",
    "Name": "datos",
    "Options": {},
    "Scope": "local"
  }
]
PS C:\Users\Usuario> docker volume inspect log
[
  {
    "CreatedAt": "2019-05-19T17:49:35-05:00",
    "Driver": "local",
    "Labels": {},
    "Mountpoint": "C:\\ProgramData\\Docker\\volumes\\log\\_data",
    "Name": "log",
    "Options": {},
    "Scope": "local"
  }
]
PS C:\Users\Usuario> _
```

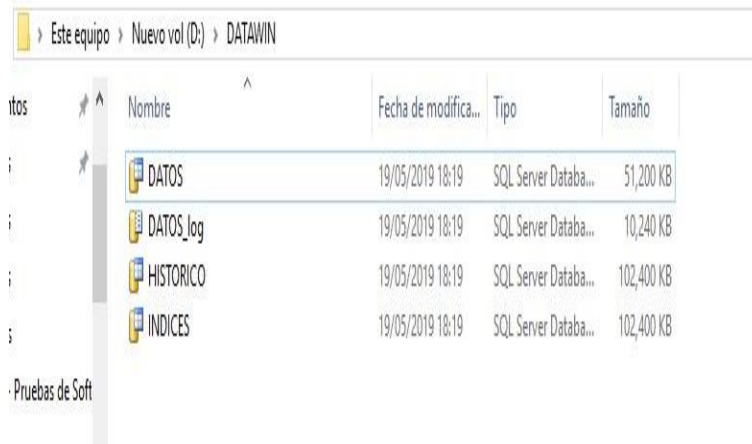
Ahora podremos usar estos volúmenes creados para crear nuestros archivos .mdf y .log en sus respectivos directorios.

### 3.6.3. Genere un nuevo contenedor con las siguientes características:

El Script es:

```
SQLQuery1.sql - (lo...111.master (sa (52))) * X
CREATE DATABASE FINANCIERA ON
PRIMARY (
    NAME = N'DATOS',
    FILENAME = N'C:\DATA\DATOS.mdf',
    SIZE = 50MB,
    FILEGROWTH = 10240KB
),
(
    NAME = N'INDICES',
    FILENAME = N'C:\DATA\INDICES.ndf',
    SIZE = 100MB,
    FILEGROWTH = 1000MB
),
(
    NAME = N'HISTORICO',
    FILENAME = N'C:\DATA\HISTORICO.ndf',
    SIZE = 100MB,
    FILEGROWTH = 51200KB
)
LOG ON (
    NAME = N'DATOS_log',
    FILENAME = N'C:\DATA\DATOS_log.ldf',
    SIZE = 10MB,
    FILEGROWTH = 10240KB
)
GO
```

– Verificamos que haya sido creado correctamente:



Nombre	Fecha de modifica...	Tipo	Tamaño
DATOS	19/05/2019 18:19	SQL Server Databa...	51,200 KB
DATOS_log	19/05/2019 18:19	SQL Server Databa...	10,240 KB
HISTORICO	19/05/2019 18:19	SQL Server Databa...	102,400 KB
INDICES	19/05/2019 18:19	SQL Server Databa...	102,400 KB

## **4. CUESTIONARIO**

**4.1. ¿Con qué comando(s) exportaría la imagen de Docker de Microsoft SQL Server a otra PC o servidor?**

– linea

**4.2. ¿Con qué comando(s) podría generar dos volúmenes para un contenedor para distribuir en un volumen el Archivo de Datos (mdf) y en otro el Archivo Log (ldf)?**

– linea

**4.3. Genere un nuevo contenedor y cree la base de datos con las siguientes características**

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

## **5. CONCLUSIONES**

Aprendimos a crear contenedores para distintos tipos de Servidores, los contenedores nos ayudan a transportar de manera fácil y eficiente las bases de datos entre diferentes tipos de Sistemas Operadores y servidores evitando así la pérdida y el desorden de datos.



## 6. BIBLIOGRAFIA

Aplicar y Desarrollar Consultas con Pivot y Grouping Sets

-

## **7. WEBGRAFIA**

<https://dockertips.com/volumenes>

- <https://cerebro-digital.com/panel/knowledgebase/64/ExportarorImportar-contenedor-de-Dockervia-archivo-TAR.html>

- 
- <https://www.docker.com/>
- <https://www.campusmvp.es/recursos/post/los-beneficios-de-utilizar-docker-y-contenedores-ala-hora-de-programar.aspx>