

INFORME DE LABORATORIO 09 INSTALACIÓN Y GESTION DE UNA BASE DE DATOS MONGODB

Huichi Contreras, Franklin Carlos (2016054948)

Escuela Profesional de Ingeniería de Sistemas

Universidad Privada de Tacna

Tacna, Perú

1. INFORMACIÓN GENERAL

1.1. *Objetivos*

- Instalación y gestion de una base de datos MongoDB

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

- Computadora con sistema operativo Windows XP, Vista, Windows 7, Windows 8 y/o Windows 8.1.
- CPU SLAT-capable feature al menos 4GB de RAM
- Docker Desktop (Para lo cual se debe primero crear una cuenta en Docker Hub)
- Microsoft SQL Server Management Studio en su última versión

2. Marco Teórico

2.1. *Docker*

Docker se define como un proyecto de código abierto que proporciona una capa de abstracción y virtualización a nivel de sistema operativo, a través de la instalación de contenedores de software.

3. PROCEDIMIENTO

3.0.1. Paso 1: Instalar imagen de mongodb

```
PS C:\Users\Usuario> docker pull mongo
Using default tag: latest
latest: Pulling from library/mongo
7ddbc47eeb70: Downloading [====>] 1.653MB/26.69MB
c1bbdc448b72: Download complete
8c3b70e39044: Download complete
45d437916d57: Waiting
e119fb0e0a55: Waiting
91f0b9bae1ea: Waiting
53e7c2967f11: Waiting
69a945568374: Waiting
93333bc225a7: Waiting
b9c10bd6c9bd: Waiting
7f4e3538e99c: Waiting
1164b51d180a: Waiting
b715a7d71f27: Waiting
```

Figura 1: Ingresar a Docker Setup

```
PS C:\Users\Usuario> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
mongo                latest              965553e202a4        5 weeks ago        363MB
microsoft/mssql-server-linux latest              314918ddaedf        11 months ago      1.35GB
mongo                4.0.4              525bd2016729        12 months ago      383MB
```

Figura 2: Comprobamos que esta arrancando el Docker

```
PS C:\Users\Usuario> docker run -d -p 27017-27019:27017-27019 --name mongoddb mongo:4.0.4
c4c4549f9020297e540adaea0aaf7be004c6d9bede56e0b7c999f0d825fee185
PS C:\Users\Usuario>
```

Figura 3: Ingresamos nuestra cuenta de Docker

```
PS C:\Users\Usuario> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS
c4c4549f9020   mongo:4.0.4 "docker-entrypoint.s..." 31 seconds ago Up 29 seconds 0.0.0.0:27017-2
```

Figura 4: Como se ve podemos ver los ajustes del Docker.

3.0.2. Paso 2: Gestionar los contenedores mediante PowerShell

```
PS C:\Users\Usuario> docker exec -it mongodb bash
root@c4c4549f9020:/# mongo
MongoDB shell version v4.0.4
connecting to: mongodb://127.0.0.1:27017
Implicit session: session { "id" : UUID("3d31ae3b-e05e-43ac-9d12-a0c2f743ab4f") }
MongoDB server version: 4.0.4
Welcome to the MongoDB shell.
For interactive help, type "help".
For more comprehensive documentation, see
  http://docs.mongodb.org/
Questions? Try the support group
  http://groups.google.com/group/mongodb-user
Server has startup warnings:
2019-12-12T14:59:17.425+0000 I STORAGE  [initandlisten] ** WARNING: Using the XFS filesystem is strongly recommended with the WiredTiger storage engine
2019-12-12T14:59:17.425+0000 I STORAGE  [initandlisten] ** See http://dochub.mongodb.org/core/prodnotes-filesystem
2019-12-12T14:59:18.445+0000 I CONTROL  [initandlisten]
2019-12-12T14:59:18.445+0000 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled for the database.
2019-12-12T14:59:18.445+0000 I CONTROL  [initandlisten] ** Read and write access to data and configuration is unrestricted.
2019-12-12T14:59:18.445+0000 I CONTROL  [initandlisten]
...
Enable MongoDB's free cloud-based monitoring service, which will then receive and display metrics about your deployment (disk utilization, CPU, operation statistics, etc).
The monitoring data will be available on a MongoDB website with a unique URL accessible to you and anyone you share the URL with. MongoDB may use this information to make product improvements and to suggest MongoDB products and deployment options to you.
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
```

Figura 5: Ingresamos “Docker versión” para ver si tenemos Docker

NAME	DESCRIPTION	STARS
OFFICIAL		
microsf/mssql-server-linux	Deprecated SQL Server on Linux Container Rep...	1157
microsf/mssql-server-windows-developer	Official Microsoft SQL Server Developer Edit...	365
microsf/mssql-server-windows-express	Official Microsoft SQL Server Express Editio...	329
microsf/mssql-tools	Official images for Microsoft SQL Server Com...	51
msmoorthy/mssql	MSSQL Database (version SQL2000)	12
datagrip/mssql-server-linux	SQL Server and SQL Server tools on Linux(201...	9
gantrior/mssql-server-2014-express-windows-with-iis	mssql 2014 + IIS	4
microsf/mssql-monitoring-influxdb	Sample Image for Influxdb, This image is des...	3
mcroe/mssqldocker	Builds on microsf/mssql-server-linux and a...	3
tsgkadot/mssql-tools	SQL Server tools on Linux (sqlcmd)	3
ansibleplaybookbundle/mssql-apb	MS SQL Server on Linux (APB)	2
bitwarden/mssql	The Bitwarden database.	2
mlitellovinx/mssql-server-linux	microsf/mssql-server-linux with mssql-tool...	2
jboesl/mssql-server-linux	mssql-server-linux with mssql-tools installe...	2
microsf/mssql-monitoring-collectd	This Sample image is designed to work with t...	1
ansibleplaybookbundle/mssql-remote-apb	An APB that deploys Microsoft SQL Server	1

Figura 6: Buscamos la Iso para descargar

```
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 [=====>] 11.21MB/42.22MB
57da90bec92c: Download complete
06fe57530625: Download complete
5a6315c8a1ff: Waiting
739f58768b3f: Waiting
0b751601bca3: Waiting
bcf04a22644a: Waiting
6b5009e4f470: Waiting
a9dca2f6722a: Waiting
```

Figura 7: Descargamos la iso definida

```
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
5a6315c8a1ff: Pull complete
739f58768b3f: Pull complete
0b751601bca3: Pull complete
bcf04a22644a: Pull complete
6b5009e4f470: Pull complete
a9dca2f6722a: Pull complete
Digest: sha256:9b70672670bb3db4b212e8aef841ca79eb2fce7d5975a5ce35b7129a9b90ec0
Status: Downloaded newer image for microsoft/mssql-server-linux:latest
docker.io/microsoft/mssql-server-linux:latest
PS C:\Users\Usuario>
```

Figura 8: Descargamos la iso definida

```
PS C:\Users\Usuario> docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
microsoft/mssql-server-linux  latest             314918ddaedf       10 months ago      1.35GB
PS C:\Users\Usuario>
```

Figura 9: Revisamos si tenemos descargado la ISO

```
PS C:\Users\Usuario> docker run -e "ACCEPT_EULA=Y" -e "SA_PASSWORD=Epis.2019" \
> -p 1433:1433 --name SQLNX01 \
> -d microsoft/mssql-server-linux
```

Figura 10: Instalamos nuestro primer contenedor con MSSQL-Server

```
PS C:\Users\Usuario> docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED          STATUS          PORTS          NAMES
279452ab7716   microsoft/mssql-server-linux        "/opt/mssql/bin/sqls..." 6 seconds ago    Up 5 seconds    0.0.0.0:1433->1433/tcp   SQLNX01
```

Figura 11: Verificamos que tenemos instalado

3.0.3. Paso 3: Conectarnos a la base de datos mediante Microsoft SQL Server Management Studio

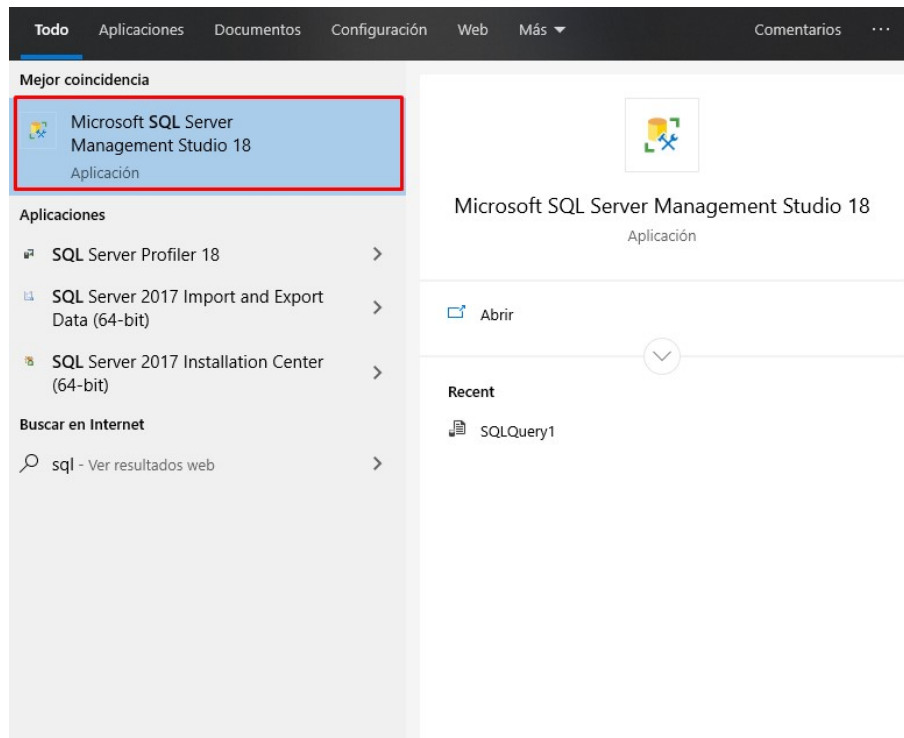


Figura 12: Ingresamos a Management Studio

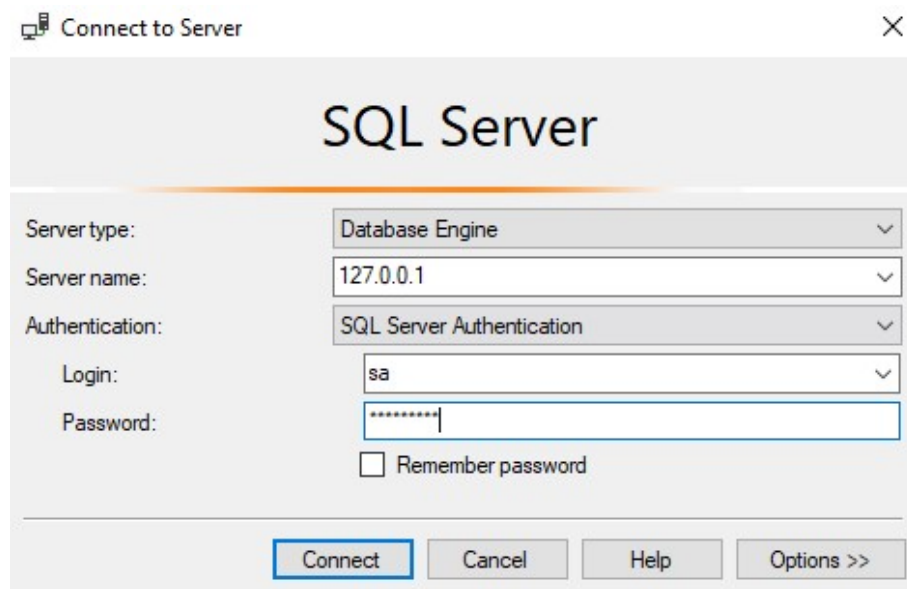


Figura 13: Ingresamos la ip local nuestro cuenta y contraseña establecida al momento de instalación

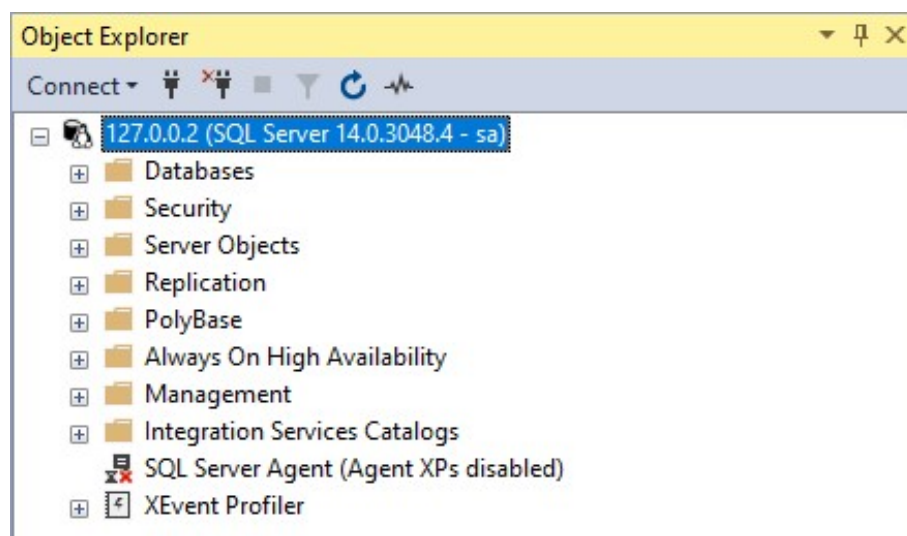


Figura 14: Como podemos visualizar ya nos podemos conectar a nuestra base de datos y visualizar que sale la ID del contenedor

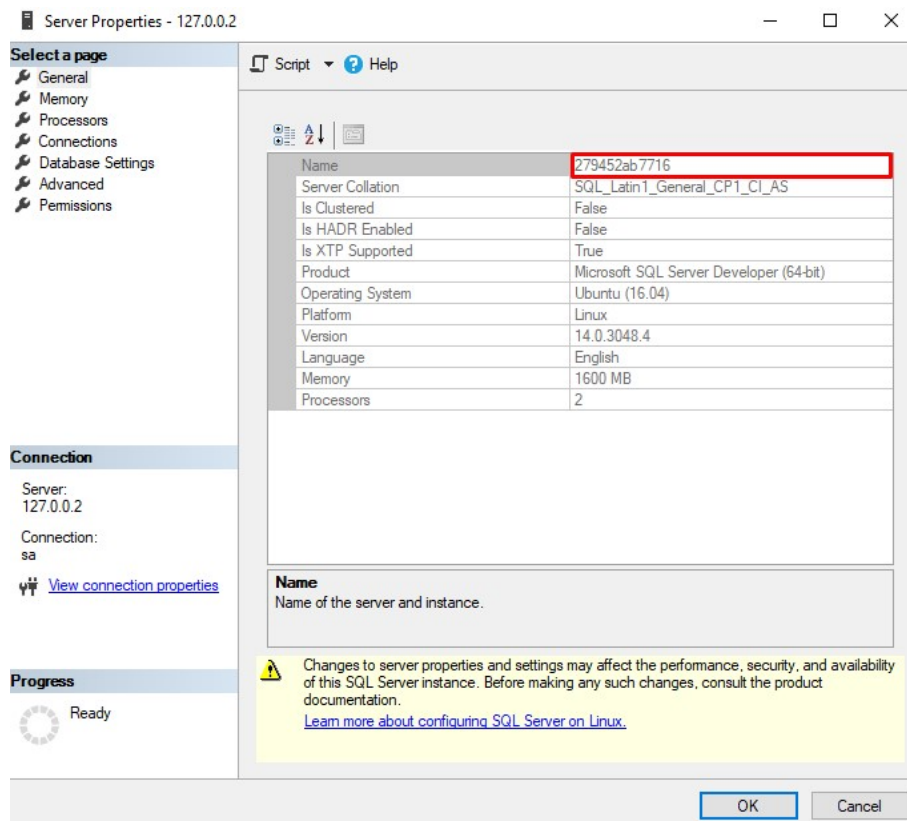


Figura 15: Como podemos visualizar ya nos podemos conectar a nuestra base de datos y visualizar que sale la ID del contenedor

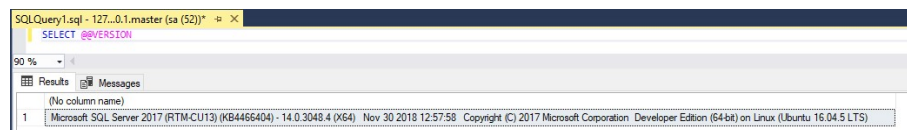


Figura 16: También podemos visualizar nuestra versión

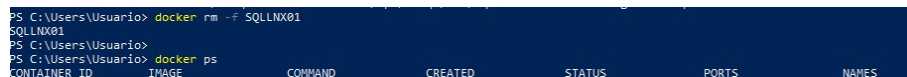


Figura 17: Por último eliminamos nuestra contenedor.

4. ANALISIS E INTERPRETACION DE RESULTADOS

- ¿Qué indican los resultados?
Pudimos realizar exitosamente la conexión de nuestro contenedor a la base de datos
- ¿Que se ha encontrado?
Encontré una manera más rápida de poder tener una base de datos SQL Server sin necesidad de estar haciendo toda la instalación necesaria del MSSQL en mi computadora.

5. CUESTIONARIO

- ¿Con qué comando(s) exportaría la imagen de Docker de Microsoft SQL Server a otra PC o servidor?
Con el comando: `docker push Tunombre/my-first-repo`
- ¿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)?
`docker volume create ArchivodeDatos.mdf`
`docker volume create ArchivodeLog.ldf`
- Genere un nuevo contenedor y cree la base de datos con las siguientes características.
Nombre : FINANCIERA
Archivos :
DATOS (mdf) : Tamaño Inicial : 50MB, Incremento: 10MB, Ilimitado
INDICES (ndf) Tamaño Inicial : 100MB, Incremento: 20MB, Maximo: 1GB
HISTORICO (ndf) Tamaño Inicial : 100MB, Incremento: 50MB, Ilimitado
LOG (ldf) Tamaño Inicial : 10MB, Incremento: 10MB, Ilimitado
¿Cuál sería el script SQL que generaría esta base de datos?


```

CREATE DATABASE FINANCIERA ON
PRIMARY (
    NAME='Datos',
    FILENAME='C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\DATA\Datos.mdf',
    SIZE=50MB,
    FILEGROWTH = 10MB
),
FILEGROUP Indices(
    NAME = 'Indices',
    FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\DATA\Indices.ndf',
    SIZE = 100MB,
    FILEGROWTH=20MB,
    MAXSIZE=1GB
),
FILEGROUP Indices(
    NAME = 'Historico',
    FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\DATA\Historico.ndf',
    SIZE = 100MB,
    FILEGROWTH=50MB
) LOG ON (
    NAME = 'LOGFinanciera',
    FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Log\LOGFinanciera.ldf',
    SIZE = 10MB ,
    FILEGROWTH = 10MB
);
GO

```

Figura 18: Script planteado

6. CONCLUSION

En conclusión, los contenedores nos ayudan a montar nuestra base de datos de forma mas rápida para poder manejar nuestros diversos sistemas a implementarlos y conectarlos, a su vez también aprendí que los ISO nos vienen a permitir con Docker subirlas para poder usarlas en otras ocasiones.