# Guía para levantar SQL Server en Docker con restauración de base de datos y persistencia

## Requisitos

- Docker instalado en tu máquina.

- Archivo .bak de la base de datos que deseas restaurar.

- SQL Server Management Studio (opcional para crear el backup, restaurar y crear usuario).

## Paso 1: Crear el contenedor Docker con SQL Server y volumen persistente

Ejecuta el siguiente comando para crear el contenedor con persistencia de datos:

docker run -e "ACCEPT\_EULA=Y" -e "MSSQL\_SA\_PASSWORD=SaPassword123" -p 1433:1433 --name sqlserver -v NAMEDatabase-data:/var/opt/mssql -d mcr.microsoft.com/mssql/server:2022-latest

## Observaciónes:

- Para la creación del contenedor de sql server vamos a requerir un usuario sa, la creación del mismo será relizada en el comando de docker.

- Será necesario recordar este usuario y su contraseña si quieres acceder al servidor sql server generado en docker desde SSMS en tu dispositivo.

- El comando -v es necesario para crear el volumen en docker, esto para agregar persistencia en el contenedor y mantener los datos y cambios en bd.

- NAMEDatabase será el nombre de tu base de datos, cambiar por el nombre real de tu base de datos y en caso del backup tambien deberá cambiarse por el nombre de tu backup.

## Paso 2: Copiar el archivo .bak al contenedor

Usa el siguiente comando para copiar el archivo de respaldo al contenedor:

docker cp C:\DockerSqlServer\Backup\NAMEDatabase .bak sqlserver:/var/opt/mssql/backups/BACKUPDB .bak

Si la carpeta no existe o recibes un mensaje en consola como

Error response from daemon: Could not find the file /var/opt/mssql/backups in container sqlserver

Entonces deberás crear esa carpeta en el directorio /var/opt/mssql de Linux

Para eso usa este comando

docker exec -it sqlserver mkdir /var/opt/mssql/backups

## Paso 3: Acceder al contenedor con sqlcmd

Ejecuta el siguiente comando para ingresar al contenedor y usar sqlcmd:

docker exec -it sqlserver /opt/mssql-tools/bin/sqlcmd -S localhost -U sa -P "SaPassword123"

## Paso 4: Restaurar la base de datos

Dentro de sqlcmd, ejecuta el siguiente script para restaurar la base de datos:

RESTORE DATABASE NAMEDatabase  
FROM DISK = '/var/opt/mssql/backups/NAMEDatabase .bak'  
WITH MOVE 'NAMEDatabase' TO '/var/opt/mssql/data/ NAMEDatabase.mdf',  
 MOVE 'NAMEDatabase\_log' TO '/var/opt/mssql/data/NAMEDatabase\_log.ldf',  
 REPLACE;  
GO

Nota: Si no puedes acceder a sqlcmd tambien puedes ingresar al servidor local generado por docker usando Sql Server Management Studio(SSMS) conectandote a:

Server Name: localhost, 1433 – o el puerto establecido

login: sa

password: SaPassword123

Abres un nuevo query y ejecutas el restore database.

## Paso 5: Crear usuario con acceso limitado a la base de datos

Dentro de sqlcmd o ssms, ejecuta el siguiente script para crear el usuario:

-- Conectarse a master para crear el login

USE master;

GO

CREATE LOGIN NAMEDatabaseUser WITH PASSWORD = 'ContraseñaSegura$1234';

GO

-- Conectarse a la base específica

USE NAMEDatabase;

GO

CREATE USER NAMEDatabaseUser FOR LOGIN NAMEDatabaseUser;

GO

EXEC sp\_addrolemember 'db\_datareader', 'NAMEDatabaseUser';  
EXEC sp\_addrolemember 'db\_datawriter', 'NAMEDatabaseUser';  
-- O si deseas que tenga control total:  
-- EXEC sp\_addrolemember 'db\_owner', 'NAMEDatabaseUser';  
GO

## Paso 6: Conexión desde tu API, servicio o aplicación

Usa la siguiente cadena de conexión en tu aplicación:

Server=localhost,1433;Database=NAMEDatabase;User Id=NAMEDatabaseUser;Password=ContraseñaSegura123!;