# Setup *WideWorldImporters* with Docker in MacOS

1. **Pull and run Azure SQL Edge image from Docker**

    a. Open a terminal window and type in the following command (make sure Docker is running):

    *docker pull mcr.microsoft.com/azure-sql-edge*

    b. Create a folder named “SQL_Server” or any name (recommended without spaces).
    c. Create a file named “docker-compose.yml” and save it with the following:

```
version: "3"
services:
    sqlserver:
        container_name: azure_sqlserver_db
        image: mcr.microsoft.com/azure-sql-edge:latest
        platform: linux/arm64
        ports:
            - "1433:1433"
        environment:
            MSSQL_SA_PASSWORD: strong$P@ssw0rd!
            ACCEPT_EULA: Y
        volumes:
            - ./mssql_data:/var/opt/mssql/data
        user: 0:0
```

    Note: If you are using a M1 Macbook make sure to specify the platform *linux/arm64*. If you are using an intel Macbook you can delete the platform line or specify *linux/amd64.*
    Also make sure to specify some strong password.

    d. Inside SQL_Server folder create a new folder named “mssql_data” or any other name. If you choose another name, make sure to change it in the docker-compose.yml file at the volumes section:

```
volumes:
      - ./<your_folder_name>:/var/opt/mssql/data
```

    e. Go to the terminal and move to the SQL_Server directory:

    *cd /Users/<username>/<path>/<to>/SQL_Server*

f. Run the docker image:

*docker-compose up -d*

## 2. Download WideWorldImporters

a. Download: https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak
b. Save this file to the "mssql_data" folder.

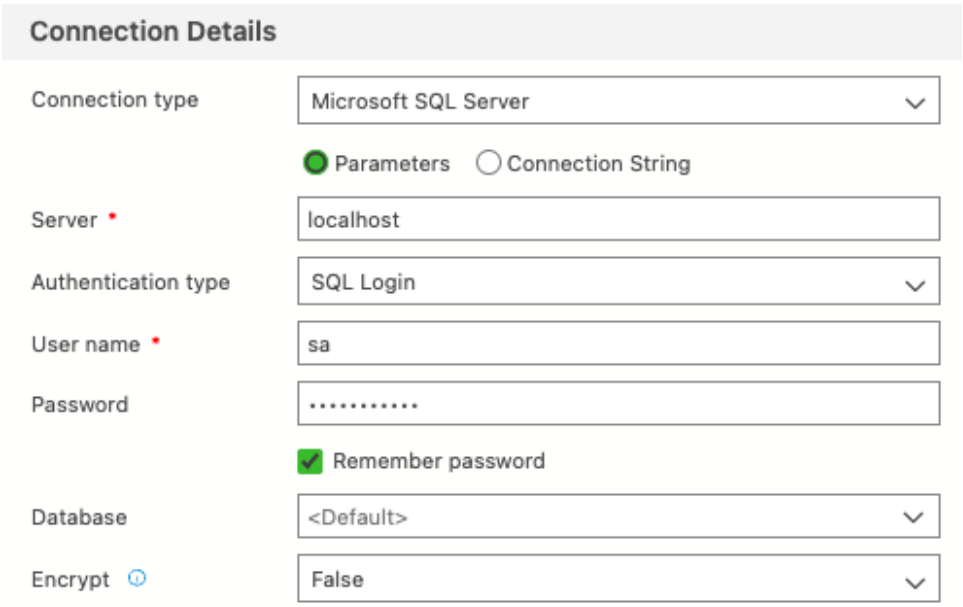
## 3. Download and Install Azure Data Studio

a. Using homebrew: *brew install --cask azure-data-studio*
b. Downloading from Microsoft: https://learn.microsoft.com/en-us/sql/azure-data-studio/download-azure-data-studio?view=sql-server-ver16&tabs=redhat-install%2Credhat-uninstall

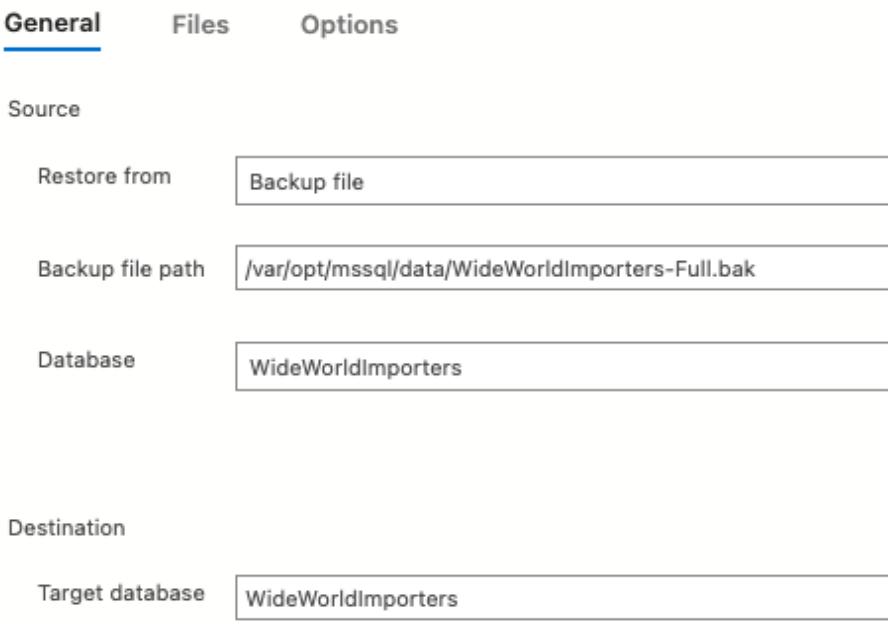
Note: For this specific case, I feel better performance downloading and installing from Microsoft.

## 4. Connect with Data Studio

a. Select New Connection
b. Fill in with the following values (use the password from the docker-compose.yml file). And connect.

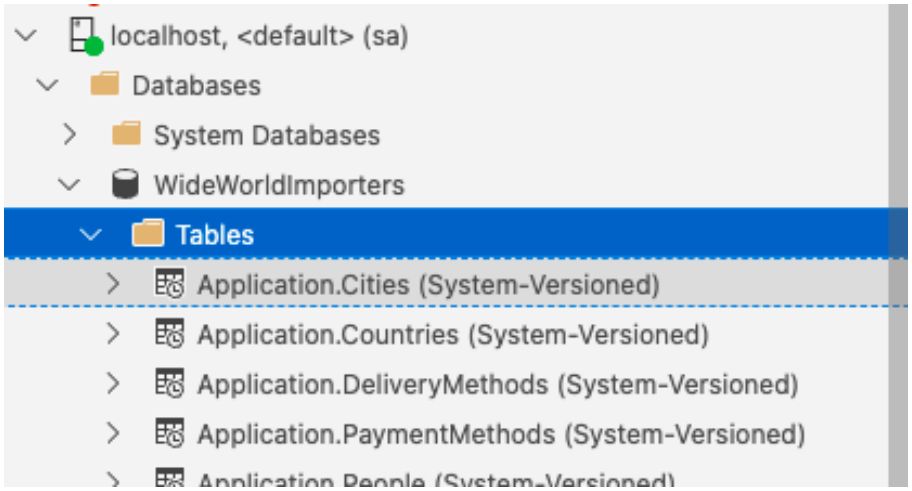



5. **Import database**
    a. Right click on the connection created and select “Manage”
    b. In the “Home” section, select the “Restore” tab
    c. Fill with the following values:



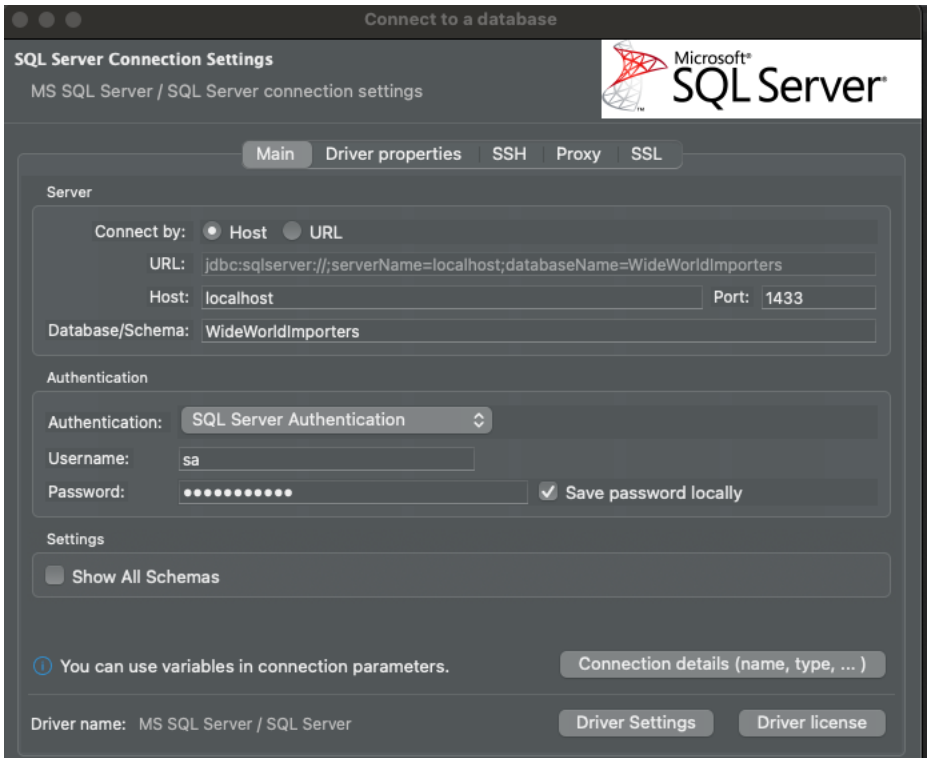

    d. Click “Restore”

e. After success, you should see the following:

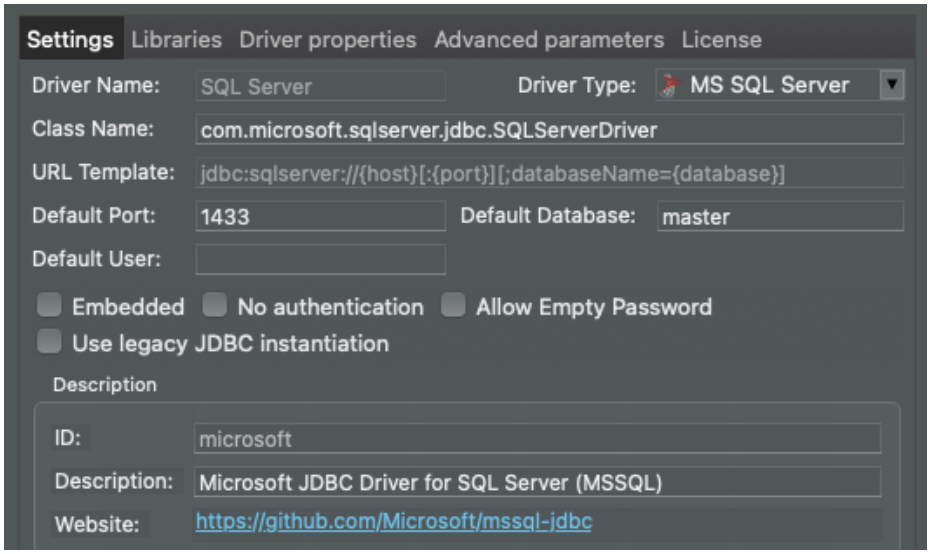



## 6. Connect with DBeaver

a. If you prefer, you can use DBeaver
b. Create a new connection and Select SQL Server (Not Azure SQL Server).
c. Fill with the following values:



d. Make sure to download the JDBC driver for SQL Server. Click in Driver Settings and save with the following values:




## 7. Data Warehouse

a. Follow the steps 2 and 5 to setup the DW using the following download link: https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImportersDW-Full.bak

## 8. Setup in Windows with SQL Server Managemente Studio

a. Download https://github.com/Microsoft/sql-server-samples/releases/download/wide-world-importers-v1.0/WideWorldImporters-Full.bak
b. Open SQL Server Management Studio and connect to the target SQL Server instance (localhost).
c. Right-click on the Databases node, and select Restore Database.
d. Select Device and click on the button ...
e. In the dialog Select backup devices, click Add, navigate to the database backup in the filesystem of the server, and select the backup. Click OK.
f. If needed, change the target location for the data and log files, in the Files pane. Note that it is best practice to place data and log files on different drives.
g. Click OK. This will initiate the database restore. After it completes, you will have the database WideWorldImporters installed on your SQL Server instance.