Extraction via mssql-scripter

Needs a previous Python installation on your system. This method is recommended for MacOS and Linux.

1. Install mssql-scripter using pip package installer for Python, follow this guide here: https://pypi.org/project/mssql-scripter/

2. Open a new terminal and type the following command to validate the installation.

   ```
   mssql-scripter --version
   ```

3. With the following command, we can check all the possible arguments we can use with this script:

   ```
   mssql-scripter --h
   ```

4. The minimum required arguments we recommend to execute the tool, are the following:

   ```
   -S, --server

   -d, --database

   -U, --user

   -P, --password

   -f, --file-path [A descriptive value, e.g.,
   <server_name>_<instance_name>_<database_name>]

   --file-per-object
   ```

5. Add your arguments depending on your SQL Server instance and the database you want to extract as a script and run the command, for example:

   ```
   mssql-scripter --server localhost --database AdventureWorks2019 --file-
   per-object --user sa --password myPassword123  --file-path
   ./AdventureWorks.sql
   ```

6. If the terminal does not display an error message, the execution was successful. You can check the output file and validate the extraction

7. Repeat step 5 for each desired database (using a different file name for each). When all databases have been extracted successfully, proceed to the next step.
8. Transmit the resulting file(s) to Snowflake for further analysis.