

SQL Query to Excel Table

User Guide

Project Name: Requête SQL vers tableau Excel

Author: Adrien Choinière

Version : 1.0

Date: 2024-07-23

Table of Contents

SQL Query to Excel Table User Guide.....	1
Context	3
Objective.....	3
SQL to Excel Table Conversion Application Documentation	4
Introduction.....	4
Prerequisites.....	4
Usage Instructions	4
Launching the Application	4
Application Menu	5
Detailed Steps.....	5
Explanation of Tags.....	7
Excel Section.....	7
Example Section	8
Example SQL Query:	8
Example of Provided Excel File	9
Live Example Protocol:	10
Expected Result of the Example:	13
Expert Tips:	15
Conclusion	15

Context

The IT team at the Business Zone within the Ministry of Employment and Social Solidarity (MESS) needs to design SQL scripts to perform various tasks on the server. For security reasons, these scripts must be sent to the Digital Data Expertise team for evaluation and execution. The transmission of the scripts is done in such a way that each line must be copied and pasted into an Excel/Word table in the first column. In the second column, the expected result is indicated by a 1 or 0. This process is long, repetitive, non-optimal, and increases the chances of errors during copying. These time losses are exacerbated if errors occur and part of the SQL query needs to be redone.

Objective

The objective of this program is to improve the efficiency of the process by reducing the time loss related to copying and pasting the SQL query into the Excel/Word table. Additionally, it reduces the chances of errors. Moreover, the employee's experience is greatly improved. Finally, monetary savings are achieved as the employee can focus on other tasks.

SQL to Excel Table Conversion Application Documentation

Introduction

Welcome to the SQL to Tableau Conversion Application documentation. This document provides detailed instructions on how to use the application to convert SQL content into Excel spreadsheets.

Prerequisites

Ensure the following prerequisites are met before using the application:

- .NET Framework installed on your machine.
- Access to SQL files (*.sql*) and Excel files (*.xlsx*) for input and output.
(Not tested on LibreOffice)

Usage Instructions

Launching the Application

You can launch the application either through the executable (`Sql_vers_tableau_words.exe`) or via PowerShell.

Executable Method

1. **Navigate to the Application Directory:**
 - Open File Explorer and go to the directory where `Sql_vers_tableau_words.exe` is located.
2. **Run the Application:**
 - Double-click on `Sql_vers_tableau_words.exe` to start the application.
 -

PowerShell Method

1. **Open PowerShell:**
 - Open PowerShell on your computer.
2. **Navigate to the Application Directory:**
 - `cd "C:\MyCustumFolder\...\FolderWhereAppIsLocated"`

3. Execute the Application:

- `.\Sql_vers_tableau_words.exe`

Application Menu

Upon launching the application, you will see a menu with various options:

- **1. Provide Custom File Paths:**
 - Allows you to specify the paths for SQL and Excel files manually.
- **2. Execute with Files:**
 - Starts the conversion process using the specified SQL and Excel files.
- **3. Toggle Debug Mode:**
 - Activates or deactivates debug mode for detailed output.
- **4. Modify the Number of Lines to Clear:**
 - Adjusts the number of lines to clear in the Excel worksheet before writing data.
- **5. Choose Language:**
 - Switches between French and English languages for the application interface.
- **0. Quit:**
 - Exits the application.

Detailed Steps

- 1. Set Custom File Paths:**
 - Enter the full path for the SQL file when prompted.
 - Enter the full path for the Excel file when prompted.
- 2. Execute with Files:**
 - Validates the file extensions (.sql for SQL file and .xlsx for Excel file).
 - Initiates the conversion process and writes data to the Excel file.
- 3. Toggle Debug Mode:**
 - Enables or disables debug messages to the console for troubleshooting purposes.
- 4. Modify the Number of Lines to Clear:**
 - Adjust the number of rows to clear in the Excel spreadsheet before writing new data. For example, if an old query takes up 100 cells and the new one only takes 10, there will be 90 non-blank cells with the old query. The purpose of this command is to modify the number of cells to be cleared in Excel before the program copy-pastes the SQL queries into the cells.
- 5. Choose Language:**
 - Selects the application language between French and English.
- 6. Quit:**
 - Exits the application gracefully.

Étapes détaillées

1. **Définir des chemins de fichiers personnalisés :**
 - Entrer le chemin complet du fichier SQL lorsque vous y êtes invité.
 - Entrer le chemin complet du fichier Excel lorsque vous y êtes invité.
2. **Exécuter avec des fichiers :**
 - Valider les extensions de fichier (.SQL pour le fichier SQL et .xlsx pour le fichier Excel).
 - Démarrer le processus de conversion.
3. **Activer/désactiver le mode débogage :**
 - Activer ou désactiver les messages de débogage dans la console pour le débogage.
4. **Modifier le nombre de lignes à effacer :**
 - Adjust the number of rows to clear in the Excel spreadsheet before writing new data. For example, if an old query takes up 100 cells and the new one only takes 10, there will be 90 non-blank cells with the old query. The purpose of this command is to modify the number of cells to be cleared in Excel before the program copy-pastes the SQL queries into the cells.
5. **Choisir la langue :**
 - Sélectionner la langue du programme entre le français et l'anglais.
6. **Quitter :**
 - Quitter le programme de manière appropriée.

Explanation of Tags

- **--%%C_TabBalise**

- The --%%C_TabBalise tag serves as a delimiter indicating that the following content will be placed in the next row of the Excel spreadsheet generated by the application.

- **--%%Result_Attendu:**

- Optional parameter. The --%%Result_Attendu: tag **MUST** be followed by 0 or 1 (for example, --%%Result_Attendu:0 or --%%Result_Attendu:1).
- Specify whether an expected result message should be included in the next column of the Excel spreadsheet.
- If --%%Result_Attendu:0 is present, the value "0" will be written in the next column.
- If --%%Result_Attendu:1 is present, the value "1" will be written in the next column.
- If --%%Result_Attendu: is not included after --%%C_TabBalise, nothing will be written in the next column, as it is an optional parameter.
- This tagging system ensures clarity and organization in the output Excel file, aligning each segment of the SQL query output with its corresponding expected result status in the generated spreadsheet. See the example below for more details.

Excel Section

To ensure the proper functioning of the program, follow these guidelines for managing the Excel file used by the program:

1. **Managing the Excel File:**

- The Excel file provided by the program serves as a backup copy and must always be duplicated to another location on your computer before use. This ensures that the original backup remains intact for reference and recovery.

2. **Starting Point:**

- The program will always start writing data from the 3rd row of the Excel spreadsheet. This ensures consistency and prevents accidental overwriting of any existing data in the first few rows of the spreadsheet.

3. **Copying the Excel File:**

- Before running the program, if not already done, manually copy the Excel file provided by the program to a designated location on your computer. You can then use this copy indefinitely.

- Use this copied version of the Excel file as input for running the program. This approach ensures that the changes made by the program do not affect the original backup file.
- Make sure the Excel file is closed before launching the application. Otherwise, an error will be raised.

By following these practices, you maintain the integrity of your data and ensure that the program operates efficiently without compromising the original backup of the Excel file.

Example Section

Example SQL Query:

Here is an example of an SQL query that demonstrates how the program processes and modifies database entries:

```
--%%C_TabBalise
SET dateformat ymd
--%%C_TabBalise
DECLARE @Item1 int,
        @Cost1 int

BEGIN TRANSACTION;

--%%C_TabBalise
--%%Result_Attendu:1
--*****
--*****      Modify Item1      *****
--*****
PRINT 'Modify Item1'

SELECT @Item1 = ID
FROM [RandomSchema].GenericTable
WHERE ItemName = 'Item1';

--%%C_TabBalise
--%%Result_Attendu:1
UPDATE [RandomSchema].GenericTable
SET LastModified = getdate(),
    Description = REPLACE(Description, 'http://www.example-url.com',
    'https://www.new-url.com')
WHERE ID = @Item1 AND Description LIKE '%http://www.example-url.com%';

--%%C_TabBalise
--%%Result_Attendu:0
```



```

SELECT *
FROM [RandomSchema].GenericTable
WHERE ID = @Item1 AND Description LIKE '%http://www.example-url.com%';

--%%C_TabBalise
--%%Result_Attendu:1
_*****
_*****          Modify Cost1          *****
_*****
PRINT 'Modify Cost1'

SELECT @Cost1 = ID
FROM [RandomSchema].GenericTable
WHERE ItemName = 'Cost1';

--%%C_TabBalise
--%%Result_Attendu:1
UPDATE [RandomSchema].GenericTable
SET LastModified = getdate(),
    URL = REPLACE(URL, 'https://www.example-site.com', 'https://www.new-example-
site.com')
WHERE ID = @Cost1 AND URL LIKE '%https://www.example-site.com%';

--%%C_TabBalise
--%%Result_Attendu:0
SELECT *
FROM [RandomSchema].GenericTable
WHERE ID = @Cost1 AND URL LIKE '%https://www.example-site.com%';

--%%C_TabBalise
ROLLBACK TRANSACTION;

```

Example of Provided Excel File

Requête SQL	Nb. rangées affectées
--Faire un SELECT des champs touchés avant et après la mise à jour	
Le programme va effacer cette ligne et débiter ici.	

Live Example Protocol:

1. Initial Setup:

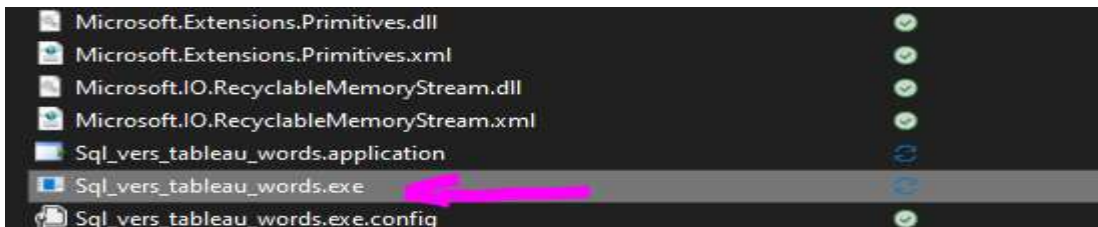
- If this is your first time using the program, copy the provided Excel file to a location of your choice. Use this copied version for all future operations.

2. Preparing the SQL Query:

- Identify the path to the .SQL query file you want to transform.
- Ensure that the `--%%C_TabBalise` and `--%%Result_Attendu:` tags are present in the SQL file where necessary.

3. Running the Application:

- Open the `Sql_vers_tableau_words.exe` program.
- Select option 5.
- Select option 2 for English.
- Select option 1.
- Enter the full path, including the file name, of your SQL query file when prompted.
- Enter the path and file name of your copied Excel file when prompted.



```
Bienvenue dans l'application de conversion SQL vers Tableau!
Veuillez choisir une option:
1. Fournir des chemins de fichiers personnalisés
2. Exécuter avec les fichiers
3. Activer/Désactiver le mode débogage
4. Modifier le nombre de lignes à effacer
5. Choisir langage
0. Quitter
Choix: 5
Veuillez choisir une option:
1. Français
2. English
Choix: 2
Welcome to SQL to Tableau conversion application!
Please choose an option:
1. Provide custom file paths
2. Execute with files
3. Toggle debug mode
4. Modify lines to clear
5. Change language
0. Quit
Choice: 1
SQL file path: C:\Users\... \Bureau\SQLs\exampleQuery.sql
Excel file path: C:\Users\... \Bureau\path_to_your_output_excel_file.xlsx
```

4. Executing the Program:

- Select option 2 to run the program.
- You can choose to leave the program open or close it after execution.

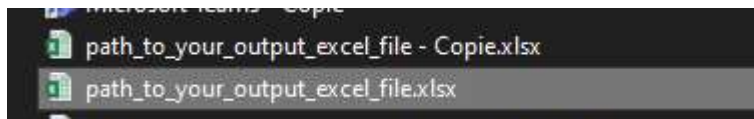
```

Bienvenue dans l'application de conversion SQL vers Tableau!
Veuillez choisir une option:
1. Fournir des chemins de fichiers personnalisés
2. Exécuter avec les fichiers
3. Activer/Désactiver le mode débogage
4. Modifier le nombre de lignes à effacer
5. Choisir langage
0. Quitter
Choix: 5
Veuillez choisir une option:
1. Français
2. English
Choix: 2
Welcome to SQL to Tableau conversion application!
Please choose an option:
1. Provide custom file paths
2. Execute with files
3. Toggle debug mode
4. Modify lines to clear
5. Change language
0. Quit
Choice: 1
SQL file path: C:\Users\... \Bureau\SQLs\exampleQuery.sql
Excel file path: C:\Users\... \Bureau\path_to_your_output_excel_file.xlsx
Welcome to SQL to Tableau conversion application!
Please choose an option:
1. Provide custom file paths
2. Execute with files
3. Toggle debug mode
4. Modify lines to clear
5. Change language
0. Quit
Choice: 2
Execution in progress...
Excel file created at: C:\Users\... \Bureau\path_to_your_output_excel_file.xlsx
Welcome to SQL to Tableau conversion application!
Please choose an option:
1. Provide custom file paths
2. Execute with files
3. Toggle debug mode
4. Modify lines to clear
5. Change language
0. Quit
Choice: _

```

5. Verifying the Output:

- Open your Excel file to check the results.
- Verify if the output meets your expectations.



6. Copying the Table to a Word File:

- Once satisfied with the output in Excel, copy the table of cells into your Word file for documentation purposes.

Expected Result of the Example:

Requête SQL	Nb. rangées affectées
<i>--Faire un SELECT des champs touchés avant et après la mise à jour</i>	
SET dateformat ymd	
DECLARE @Item1 int, @Cost1 int BEGIN TRANSACTION;	
<pre>-- ***** --***** Modify Item1 ***** -- ***** PRINT 'Modify Item1' SELECT @Item1 = ID FROM [RandomSchema].GenericTable WHERE ItemName = 'Item1';</pre>	1
<pre>UPDATE [RandomSchema].GenericTable SET LastModified = getdate(), Description = REPLACE(Description, 'http://www.example-url.com', 'https://www.new-url.com') WHERE ID = @Item1 AND Description LIKE '%http://www.example- url.com%';</pre>	1

SELECT * FROM [RandomSchema].GenericTable WHERE ID = @Item1 AND Description LIKE '%http://www.example-url.com%';	0
-- ***** --***** Modify Cost1 ***** -- ***** PRINT 'Modify Cost1' SELECT @Cost1 = ID FROM [RandomSchema].GenericTable WHERE ItemName = 'Cost1';	1
UPDATE [RandomSchema].GenericTable SET LastModified = getdate(), URL = REPLACE(URL, 'https://www.example-site.com', 'https://www.new-example-site.com') WHERE ID = @Cost1 AND URL LIKE '%https://www.example-site.com%';	1
SELECT * FROM [RandomSchema].GenericTable WHERE ID = @Cost1 AND URL LIKE '%https://www.example-site.com%'; ROLLBACK TRANSACTION;	0

Expert Tips:

- The paths of the SQL and Excel files are stored in memory for the duration of the program session. Once entered, you can execute option 2 without re-entering the paths.
- It is recommended to keep the program open until you are satisfied with the results in Excel, allowing for iterative adjustments if necessary.
- Ensure that the Excel file is closed before running the application. Otherwise, an error will be raised.
- It is possible to change the number of rows to clear in the Excel table with menu option 4.

Conclusion

This document has provided complete instructions on using the SQL to Excel Table conversion application. For further assistance or troubleshooting, refer to the displayed error messages or consult the code.