

Requête SQL en tableau Excel

Guide utilisateur

Nom du projet: Requête SQL vers tableau Excel

Auteur: Adrien Choinière

Version : 1.0

Date: 2024-07-22

Table des matières

Requête SQL en tableau Excel	1
Mise en contexte	3
Objectif	3
Documentation de l'application de conversion SQL en Tableau.....	4
Introduction	4
Prérequis	4
Instructions d'utilisation.....	4
Lancement de l'application	4
Menu de l'application.....	5
Explication des balises	6
Section Excel.....	7
Section Exemple	8
Exemple de requête SQL	8
Exemple de fichier Excel fourni	9
Exemple de protocole avec exemple :	9
Conseils d'expert :	14
Conclusion	14

Mise en contexte

L'équipe TI de la Zone Entreprise au sein du ministère de l'Emploi et de la Solidarité sociale (MESS) doit concevoir des scripts SQL afin de réaliser diverses tâches sur le serveur. Pour des raisons de sécurité, ces scripts doivent être transmis à l'équipe de la Direction de l'expertise des données numériques afin d'être évalués et exécutés. La transmission des scripts s'effectue de manière que chaque ligne doit être copiée-collée dans un tableau Excel/Word dans la première colonne. Dans la deuxième colonne, le résultat attendu est indiqué par un 1 ou 0. Ce processus est long, répétitif, non optimal et augmente les chances d'erreurs lors du recopiage. Ces pertes de temps sont accentuées si des erreurs surviennent et qu'il faut refaire une partie de la requête SQL.

Objectif

L'objectif de ce programme est d'améliorer l'efficacité du processus, en réduisant, d'une part, la perte de temps liée au copier-coller de la requête SQL vers le tableau Excel/Word. D'autre part, il réduit les chances d'erreur. De plus, l'expérience de l'employé est grandement améliorée. Enfin, un gain monétaire est réalisé, puisque l'employé peut se concentrer sur d'autres tâches.

Documentation du programme de conversion SQL en Tableau

Introduction

Bienvenue dans la documentation du programme de conversion SQL en Tableau. Ce document fournit des instructions détaillées sur l'utilisation de l'application pour convertir le contenu SQL en feuilles de calcul Excel.

Prérequis

Assurez-vous que les prérequis suivants sont remplis avant d'utiliser l'application :

- .NET Framework installé sur votre machine.
- Accès aux fichiers SQL (.sql) et Excel (.xlsx) pour l'entrée et la sortie. (Non testé sur LibreOffice)

Instructions d'utilisation

Lancement de l'application

Lancer le programme soit par l'exécutable (Sql_vers_tableau_words.exe) soit via l'invite de commande PowerShell.

Méthode par exécutable

1. Naviguer vers le répertoire du programme :
 - Ouvrir l'Explorateur de fichiers et aller dans le répertoire où Sql_vers_tableau_words.exe est situé.
2. Exécuter l'application :
 - Double-cliquez sur Sql_vers_tableau_words.exe pour démarrer le programme.

Méthode par invite de commande

1. Ouvrir Terminal/PowerShell :
 - Ouvrir PowerShell sur votre ordinateur.
2. Naviguer vers le répertoire du programme:
 - `CD "C:\MonDossierPersonnalisé\...\DossierOùLAppEstSituée"`
3. Exécuter l'application :
 - `.\Sql_vers_tableau_words.exe`

Menu du programme

Après avoir lancé l'application, un menu avec différentes options :

- **1. Fournir des chemins de fichiers personnalisés :**
 - Permet de spécifier manuellement les chemins des fichiers SQL et Excel.
- **2. Exécuter avec des fichiers :**
 - Lance le processus de conversion en utilisant les fichiers SQL et Excel spécifiés.
- **3. Activer/désactiver le mode débogage :**
 - Active ou désactive le mode débogage pour des sorties détaillées.
- **4. Modifier le nombre de lignes à effacer :**
 - Ajuste le nombre de lignes à effacer dans la feuille de calcul Excel avant d'écrire les données.
- **5. Choisir la langue :**
 - Bascule entre les langues française et anglaise pour l'interface du programme.
- **0. Quitter :**
 - Quitte le programme.

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 :**
 - Ajuster le nombre de lignes à effacer dans la feuille de calcul Excel avant d'écrire de nouvelles données. Par exemple, si une ancienne requête prend 100 cellules et que la nouvelle en prend seulement 10. Il y aura 90 cellules non vierges avec

l'ancienne requête. Le but de cette commande est de modifier le nombre de cellules à rendre vierge sur Excel avant que le programme copier-coller les requête SQL dans les cellules.

5. **Choisir la langue :**
 - Sélectionner la langue du programme entre le français et l'anglais.
6. **Quitter :**
 - Quitter le programme de manière appropriée.

Explication des balises

- **--%%C_TabBalise**
 - La balise --%%C_TabBalise sert de délimiteur indiquant que le contenu suivant sera placé dans la ligne suivante de la feuille de calcul Excel générée par l'application.
- **--%%Result_Attendu:**
 - Paramètre optionnel. La balise --%%Result_Attendu: **DOIT** être suivie de 0 ou 1 (par exemple, --%%Result_Attendu:0 ou --%%Result_Attendu:1).
 - Spécifier si un message de résultat attendu doit être inclus dans la colonne suivante de la feuille de calcul Excel.
 - Si --%%Result_Attendu:0 est présent, la valeur « 0 » sera écrite dans la colonne suivante.
 - Si --%%Result_Attendu:1 est présent, la valeur « 1 » sera écrite dans la colonne suivante.
 - Si --%%Result_Attendu: n'est pas inclus après --%%C_TabBalise, rien ne sera écrit dans la colonne suivante, car il s'agit d'un paramètre optionnel.
 - Ce système de balisage assure clarté et organisation dans le fichier Excel de sortie, alignant chaque segment de sortie de la requête SQL avec son statut de résultat attendu correspondant dans la feuille de calcul générée. Voir l'exemple ci-dessous pour plus de détails.

Section Excel

Pour assurer le bon fonctionnement du programme, suivez ces directives pour gérer le fichier Excel utilisé par le programme :

1. Gestion du fichier Excel :

- Le fichier Excel fourni par le programme sert de copie de sauvegarde et doit toujours être dupliqué à un autre endroit sur votre ordinateur avant utilisation. Cela garantit que la sauvegarde originale reste intacte pour référence et récupération.

2. Point de départ :

- Le programme commencera toujours à écrire des données à partir de la 3e ligne de la feuille de calcul Excel. Cela assure la cohérence et empêche l'écrasement accidentel de toute donnée existante dans les premières lignes de la feuille de calcul.

3. Copie du fichier Excel :

- Avant de lancer le programme, si ce n'est pas déjà fait, copier manuellement le fichier Excel fourni par le programme à un emplacement désigné sur votre ordinateur. Vous pouvez ensuite utiliser cette copie indéfiniment.
- Utiliser cette version copiée du fichier Excel comme entrée pour l'exécution du programme. Cette façon de faire garantit que les modifications apportées par le programme n'affectent pas le fichier de sauvegarde original.
- Assurez-vous que le fichier Excel est fermé avant de lancer l'application. Sinon, une erreur sera levée.

En suivant ces façons de faire, vous maintenez l'intégrité de vos données et assurez que le programme fonctionne efficacement sans compromettre la sauvegarde originale du fichier Excel.

Section Exemple

Exemple de requête SQL :

Voici un exemple de requête SQL qui démontre comment le programme traite et modifie les entrées de la base de données :

```
--%%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;
```

Exemple de fichier Excel fourni

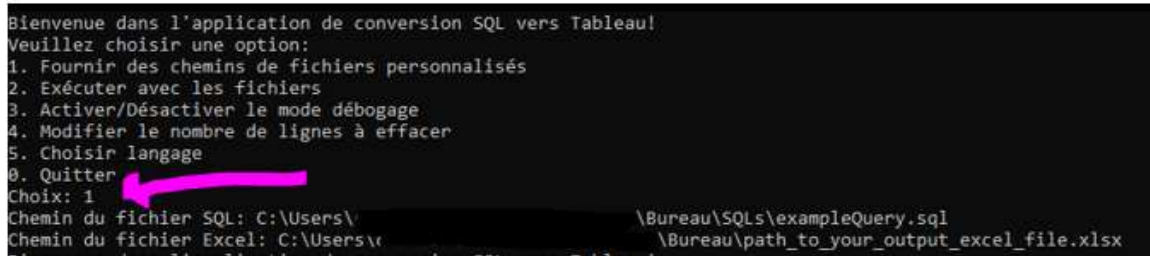
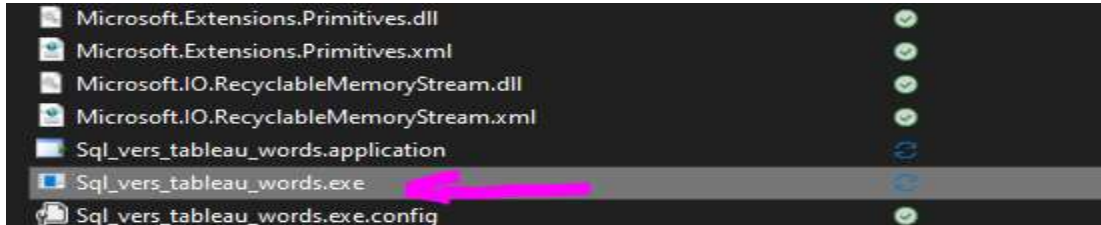
Requête SQL	Nb. rangées affectées
<i>Faire un SELECT des champs touchés avant et après la mise à jour</i>	
Le programme va effacer cette ligne et débiter ici.	

Exemple de protocole avec exemple :

1. **Configuration initiale :**
 - Si c'est la première fois que vous utilisez le programme, copiez le fichier Excel fourni à l'emplacement de votre choix. Utilisez cette version copiée pour toutes les opérations futures.
2. **Préparer la requête SQL :**
 - Identifier le chemin vers le fichier de requête .SQL que vous souhaitez transformer.
 - Assurez-vous que les balises --%%C_TabBalise et --%%Result_Attendu: sont présentes dans le fichier SQL là où c'est nécessaire.

3. Exécuter l'application :

- Ouvrir le programme `Sql_vers_tableau_words.exe`.
- Sélectionner l'option 1.
- Entrer le chemin complet, y compris le nom du fichier, de votre fichier de requête SQL lorsque vous y êtes invité.
- Entrer le chemin et le nom du fichier de votre fichier Excel copié lorsque vous y êtes invité.



4. Exécuter le programme :

- Sélectionner l'option 2 pour exécuter le programme.
- Vous pouvez choisir de laisser le programme ouvert ou la fermer après l'exécution.

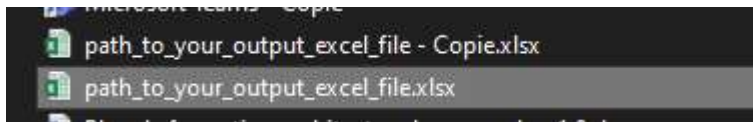
```

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: 1
Chemin du fichier SQL: C:\Users\... \Bureau\SQLs\exampleQuery.sql
Chemin du fichier Excel: C:\Users\... \Bureau\path_to_your_output_excel_file.xlsx
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: 2
Execution en cours...
Fichier Excel créé à: C:\Users\... \Bureau\path_to_your_output_excel_file.xlsx
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. Vérifier la sortie :

- Ouvrir votre fichier Excel pour vérifier les résultats.
- Vérifier si la sortie répond à vos attentes.



Résultat attendu de l'exemple:

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
<pre>SELECT * FROM [RandomSchema].GenericTable WHERE ID = @Item1 AND Description LIKE '%http://www.example- url.com%';</pre>	0

<pre>-- ***** --***** Modify Cost1 ***** -- ***** PRINT 'Modify Cost1' SELECT @Cost1 = ID FROM [RandomSchema].GenericTable WHERE ItemName = 'Cost1';</pre>	1
<pre>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%';</pre>	1
<pre>SELECT * FROM [RandomSchema].GenericTable WHERE ID = @Cost1 AND URL LIKE '%https://www.example-site.com%';</pre>	0
<pre>ROLLBACK TRANSACTION;</pre>	

6. Copier le tableau dans un fichier Word

- Une fois satisfait de la sortie dans Excel, copier le tableau de cellules dans votre fichier Word à des fins de documentation.

Conseils d'expert :

- Les chemins des fichiers SQL et Excel sont stockés en mémoire pendant toute la durée de la session du programme. Une fois saisis, vous pouvez exécuter l'option 2 sans ressaisir les chemins.
- Il est recommandé de garder le programme ouvert jusqu'à ce que vous soyez satisfait des résultats dans Excel, permettant ainsi des ajustements itératifs si nécessaire.
- Assurez-vous que le fichier Excel est fermé avant d'exécuter l'application. Sinon, une erreur sera levée.
- Il est possible de changer le nombre de lignes à effacer dans le tableau Excel avec le choix 4 du menu.

Conclusion

Ce document a fourni les instructions complètes sur l'utilisation de l'application de conversion SQL en Tableau Excel. Pour une assistance supplémentaire ou un dépannage, consultez les messages d'erreur affichés ou consultez le code.

Adrien Choinière