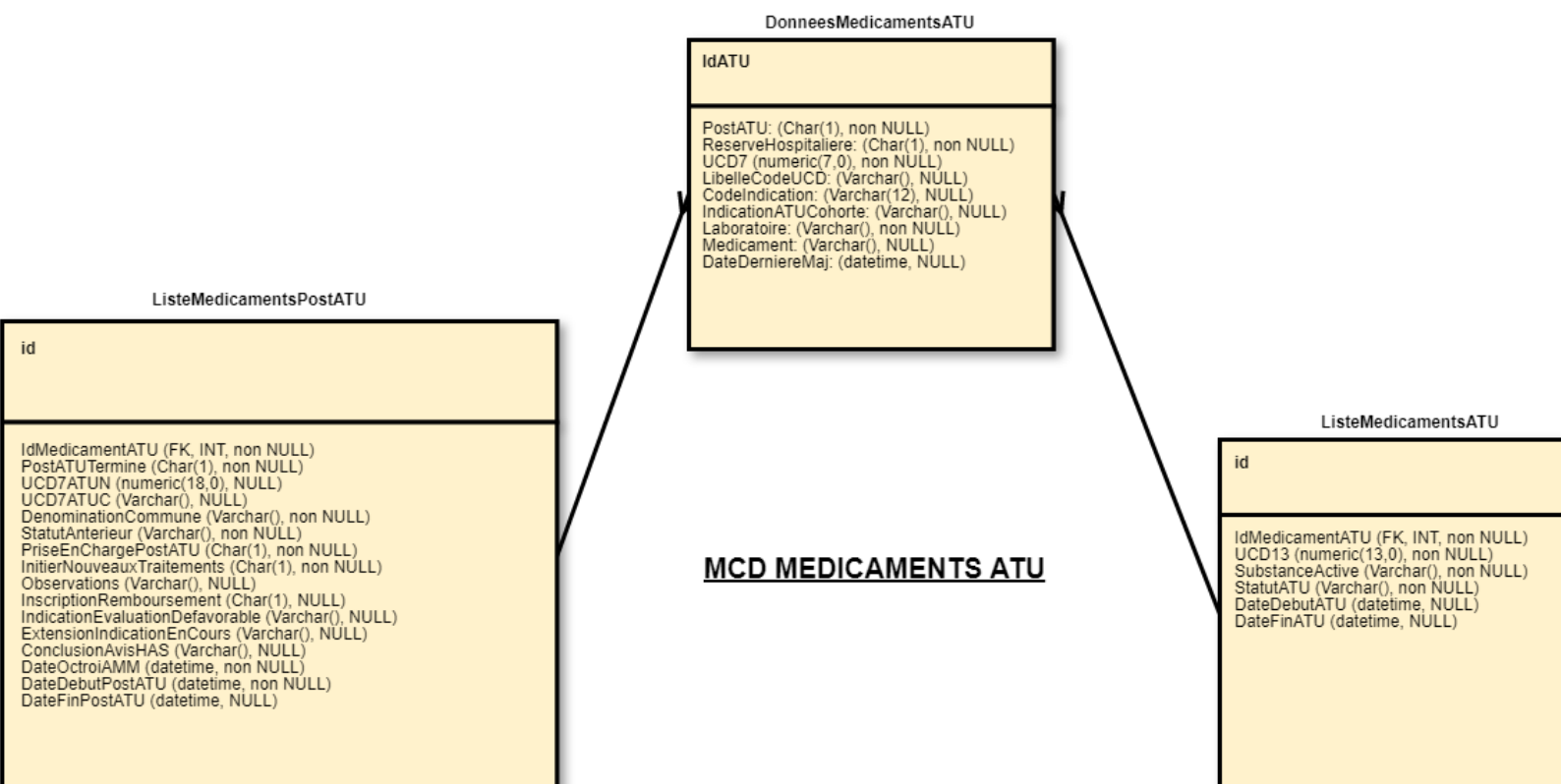


# Outil de génération d'INSERT en base SQL pour Médicament ATU et POST ATU

## Introduction :

Cet outil de génération de script INSERT SQL a été généré selon le modèle de conception de données (MCD) suivant :



Il est requis de faire un traitement sur les fichiers .xlsx fournis pour ensuite les convertir en fichiers .csv.

Dans l'exemple de traitement que j'expliciterais dans cette documentation, les fichiers suivants ont été utilisés :

- **dgos\_tableau\_codes-atu\_280219.xlsx**

(Contient les données des médicaments ATU)

- **tableau\_codage\_par\_indication\_post-atu\_1er\_avril\_2019corr1.xlsx**

(Contient 2 feuilles, l'une pour les médicaments Post ATU et l'autre pour les médicaments Post ATU Terminés).

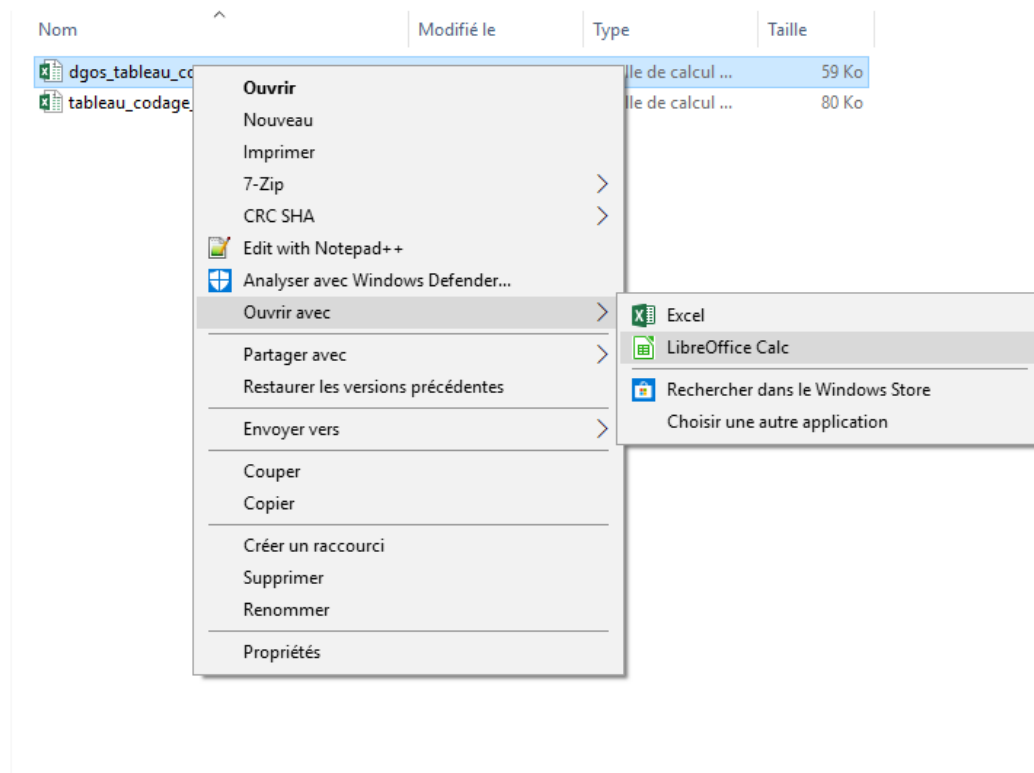
## I – Conversion des fichiers XLSX en fichiers de CSV

La première étape consiste en transformer les fichiers reçus sous format .xlsx en format .csv.

C'est une étape contraignante dans le sens où si les fichiers ne sont pas correctement convertis l'outil de génération de scripts d'Insert SQL ne pourra pas fonctionner, mais cette étape fait considérablement gagner du temps sur le processus de génération de scripts d'Insertion SQL.

A noter que les exemples de conversion utiliseront le logiciel LibreOffice Calc (car les options de conversion en format .csv sont les plus simples et les plus sûres d'utilisation par rapport aux autres logiciels, par exemple pour Excel on ne dispose que de très peu d'options de paramétrage).

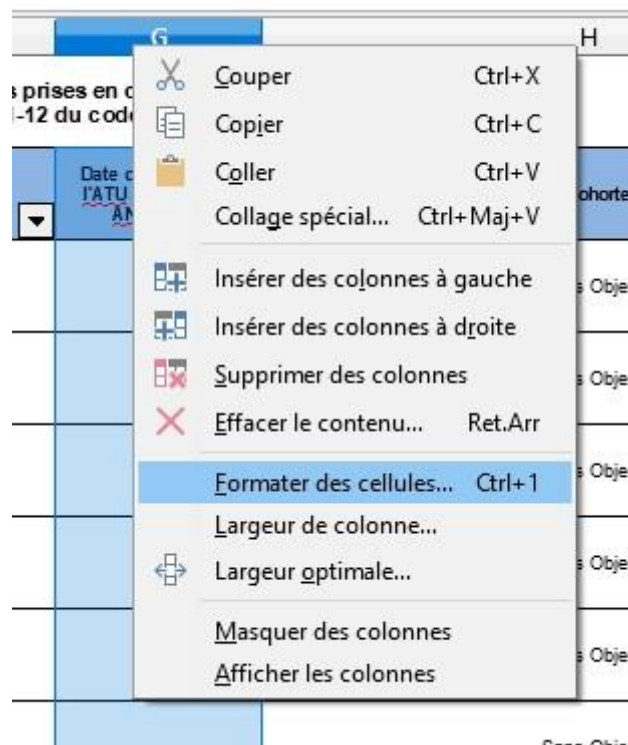
\*1) Tout d'abord commencez par ouvrir le fichier .xlsx d'origine avec Libre Office (Clic droit -> Ouvrir Avec -> Libre Office Calc)



\*2) Puis assurez vous bien que les colonnes regroupant des dates soient correctement formatées, pour cela, sélectionnez la colonne contenant une date avec un clic gauche puis faites un clic droit et cliquez sur Formater Des Cellules :

| Observations   | Date de dernière mise à jour |
|--|------------------------------|
| <p>s par le CHMP qui n'ont pas fait l'objet d'une autorisation</p> <p>thérapeutiques prises en charge par les régimes obligatoires<br/>prévention de routine des crises récurrentes d'angio-<br/>le type I, II et III, chez les patients âgés de 12 ans et plus,<br/>qués dans la prévention de routine sont efficaces ou<br/>ine prise en charge dérogatoire au titre du dispositif post-<br/>ette indication.</p> <p>thérapeutique prise en charge par les régimes obligatoires<br/>prévention de routine des crises récurrentes d'angio-<br/>le type III, chez les patients âgés de 12 ans et plus, lorsque<br/>is la prévention de routine sont inefficaces ou</p> <p>ir arrêté du 14 février 2019 sur la liste collectivités dans<br/>aitement des patients adultes atteints d'un cancer<br/>lules (CBNPC) localement avancé ou métastatique après<br/>re, les patients avec mutations activatrices de l'EGFR<br/>u une thérapie ciblée. Dès lors, aucune prise en charge</p> | <p>avr-19</p> <p>mars-19</p> |

(exemple d'une cellule mal formatée à l'ouverture du document)



Formatage des cellules

Nombres | Police | Effets de caractère | Alignement | Bordures | Arrière-plan | Protection de cellule

| Catégorie                | Format                    | Langue                         |
|--------------------------|---------------------------|--------------------------------|
| Défini par l'utilisateur | 31/12/1999                | Par défaut - Français (France) |
| Nombre                   | vendredi 31 décembre 1999 |                                |
| Pourcentage              | 31/12/99                  |                                |
| Monnaie                  | 31/12/1999                |                                |
| <b>Date</b>              | 31 déc. 99                |                                |
| Heure                    | 31 déc. 1999              |                                |
| Scientifique             | 31. déc. 1999             |                                |
| Fraction                 | 31 décembre 1999          |                                |
| Valeur logique           | 31. décembre 1999         |                                |
| Texte                    | ven. 31 déc. 99           |                                |
|                          | ven. 31/déc. 99           |                                |

Options

Nombre de décimales : 0 ☐ Nombres négatifs en rouge

Zéros non significatifs : 0 ☐ Séparateur de milliers

Description de format

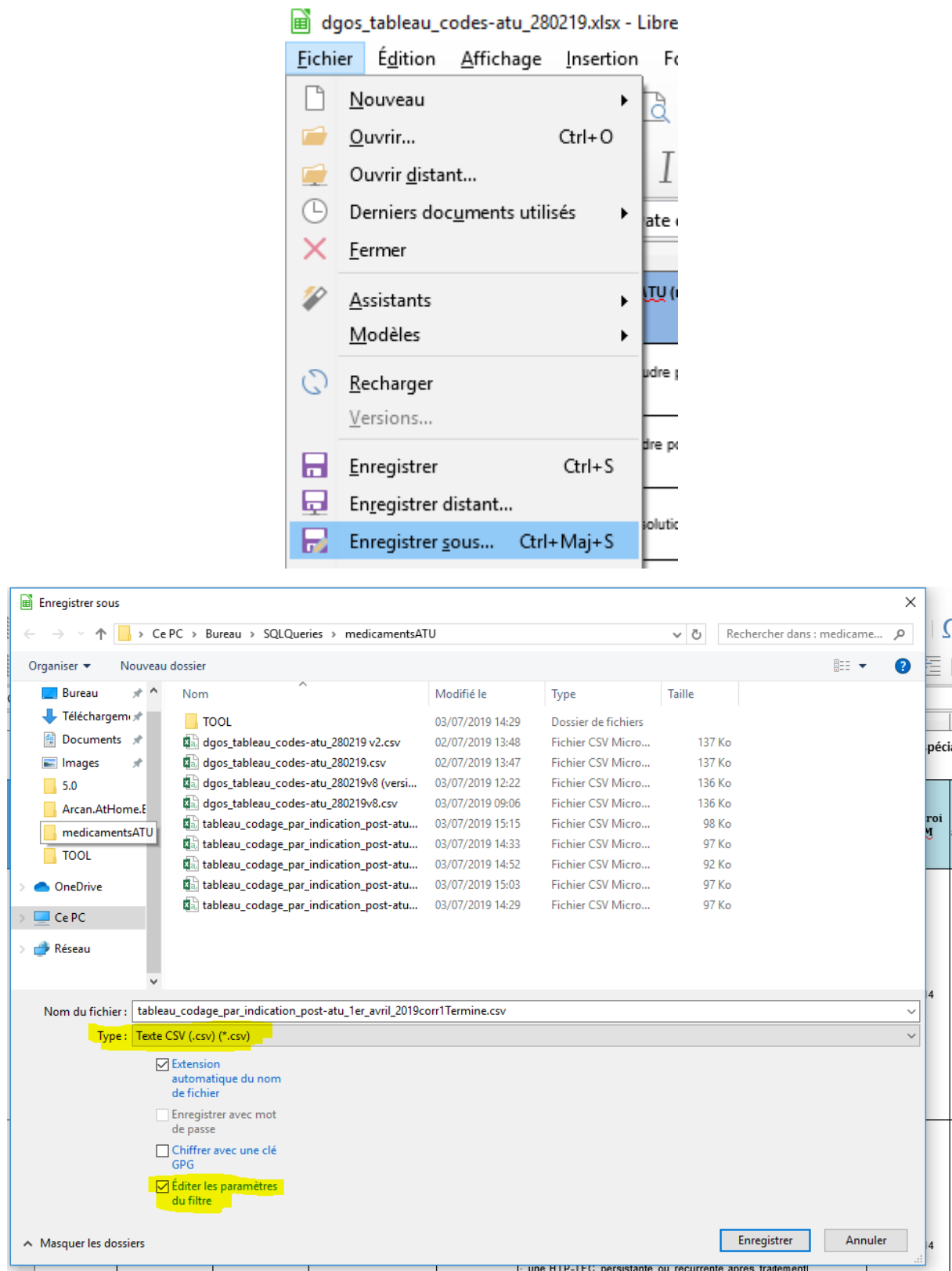
JJ/MM/AAAA

Aide OK Annuler Réinitialiser

Et formatez la cellule selon l'impression d'écran ci-dessus.

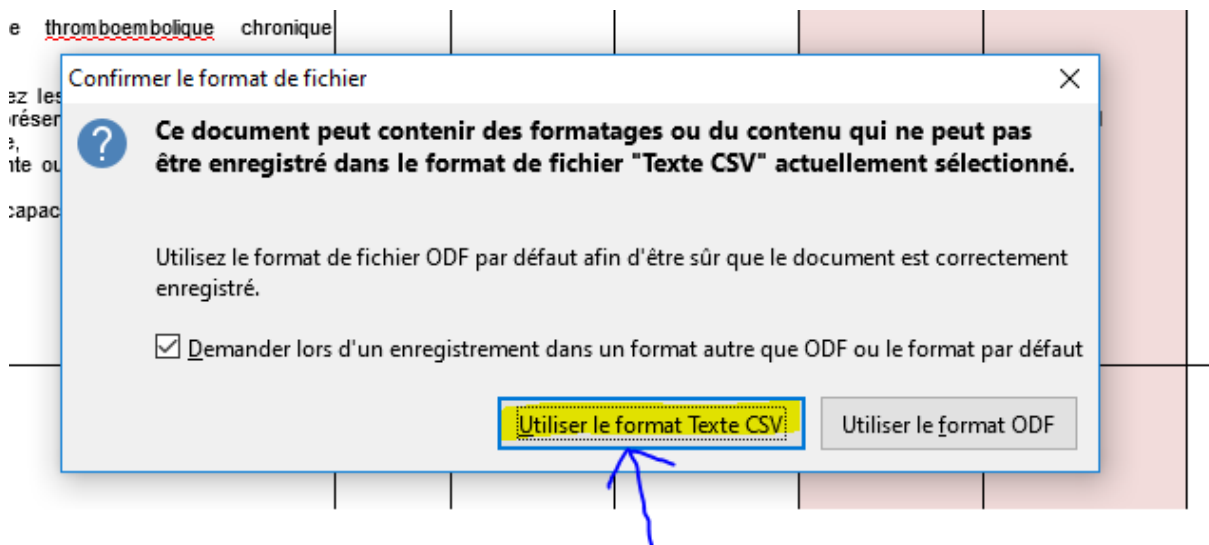
\*3) Maintenant nous allons enregistrer le fichier.xlsx en format csv.

Pour cela allez dans Fichier->Enregistrer Sous :



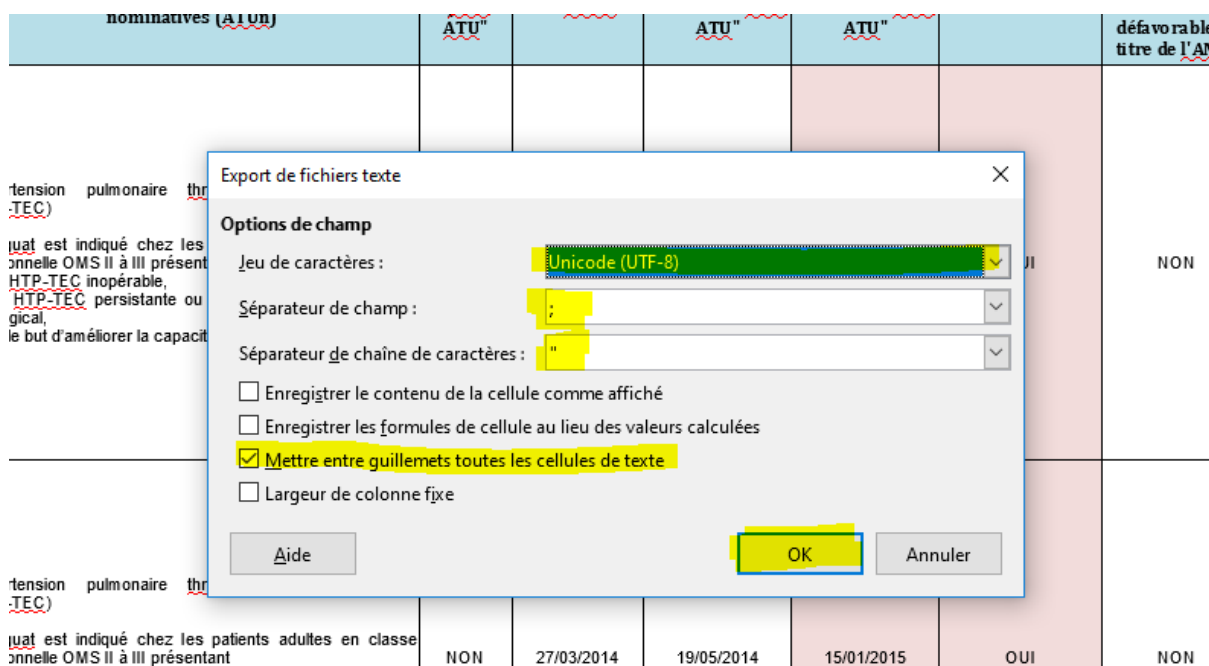
Sélectionnez Texte CSV(.csv) dans la liste déroulante « Type ». Puis cochez la case « Editer les paramètres du filtre ». Et cliquez sur Enregistrer.

Vous arrivez sur une nouvelle fenêtre qui vous demande dans quel type enregistrer le fichier, confirmez l'enregistrement en .csv :



(L'avertissement concerne la mise en forme colorisée du document... ce qui nous importe peu)

On arrive maintenant sur la fenêtre la plus importante de tout le processus :



Sélectionnez Unicode UTF-8 dans la liste déroulante de Jeu de caractères.

Dans le champ « séparateur de champ », rentrez le caractère « ; »

Dans le champ « Séparateur de chaîne de caractères », rentrez le caractère "

Puis décochez tout et cochez seulement « Mettre entre guillemets toutes les cellules de texte »

(une autre possibilité pour s'assurer de ne pas avoir d'erreur lors de la création des fichiers csv peut être de sélectionner toutes les colonnes et de formater toutes les cellules sous format «Texte », cf étape \*2) )

Enfin cliquez sur OK.

Le fichier est maintenant créé mais il reste encore de nombreuses étapes pour s'assurer que le fichier sera correctement pris en charge par l'outil.

\*4) Ouvrez maintenant le fichier nouvellement créé avec Libre Office Calc.

Import de texte - [dgos\_tableau\_codes-atu\_280219Exemple.csv]

**Importer**

Jeu de caractères : Unicode (UTF-8)

Langue : Par défaut - Français (France)

À partir de la ligne : 1

**Options de séparateur**

☐ Largeur fixe ☒ Séparé par

☐ Tabulation ☐ Virgule ☒ Point-virgule ☐ Espace ☐ Autre

☐ Fusionner les séparateurs ☐ Espaces superflus

Séparateur de chaîne de caractères : "

**Autres options**

☐ Formater les champs entre guillemets comme texte ☐ Détecer les nombres spéciaux

**Champs**

Type de colonne : Texte

|   | Texte                                      | Texte                                |
|---|--|--------------------------------------|
| 1 |  |                                      |
| 2 | Date de dernière mise à jour (Source DGOS) | Médicaments sous ATU (nom, dosage,   |
| 3 | 19/02/2019                                 | ABT-414 100 mg, poudre pour solution |
| 4 | 19/02/2019                                 | ABT-414 20 mg, poudre pour solution  |
| 5 | 19/02/2019                                 | ADAGEN 250 U/ml, solution injectable |
| 6 | 19/02/2019                                 | ADV7103, granulés à libération prol  |
| 7 | 19/02/2019                                 | ADV7103, granulés à libération prol  |
| 8 | 19/02/2019                                 | ALINIA 100 mg/5 ml, suspension buva  |
| 9 | 19/02/2019                                 | ALINIA 500 mg comprimé               |

Aide OK Annuler

Vous vous retrouvez avec une fenêtre d'importation du texte .csv pour libreOffice, vérifiez bien que tous les éléments correspondent bien à l'impression d'écran ci-dessus (Notamment Unicode UTF-8 pour le jeu de caractère. Pensez aussi à cliquer sur la petite case blanche en haut à gauche du tableau montrant un aperçu du CSV, cela permet de sélectionner l'intégralité du CSV pour ensuite le formater en format Texte en choisissant « Texte » dans la liste déroulante « Type de Colonne ».

Confirmez en cliquant sur OK.

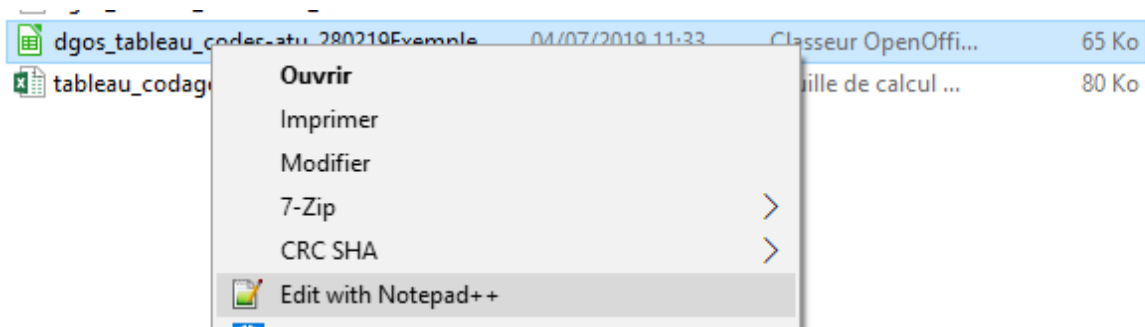
\*5) Supprimez la première ligne inutile (entête qui explique à quoi sert le document) en sélectionnant la première ligne et en faisant clic droit->supprimer des lignes

A1:AMJ1

| A   | B                              | C                         | D   | E                                    | F                                  | G          | H |
|---|--------------------------------|---------------------------|---|--------------------------------------|------------------------------------|------------|---|
| Tableau des codes UCD et codes indications des spécialités prises en charge au titre d'une ATU article L.5121-12 du code de la santé publique |                                |                           |   |                                      |                                    |            |   |
| ATU (nom, dosage, forme)  | Substance active (Source ANSM) | Laboratoire (Source ANSM) | Statut de l'ATU octroyée (nominative/cohortes) (source: ANSM) | Date de début de l'ATU (Source ANSM) | Date de fin de l'ATU (Source ANSM) | Indication |   |
| pour solution à diluer pour   | dépatuozumab mafodotine        | ABBVIE                    | Nominative  | Sans Objet                           |                                    |            |   |
| pour solution à diluer pour   | dépatuozumab mafodotine        | ABBVIE                    | Nominative  | Sans Objet                           |                                    |            |   |
| solution injectable   | pégademase bovine              | LEADIANT BIOSCIENCES LTD. | Nominative  | Sans Objet                           |                                    |            |   |

Pensez à enregistrer le fichier.

\*6) Maintenant on peut ouvrir le fichier avec Notepad++. On va effectuer quelques corrections que LibreOffice n'a pas pu faire pour nous pour s'assurer de la validité du fichier csv.



Faites un clic droit sur le fichier csv puis Edit With Notepad++

\*7) Dans un premier temps on constate que les entêtes sont complètement non fonctionnels en l'état :

```

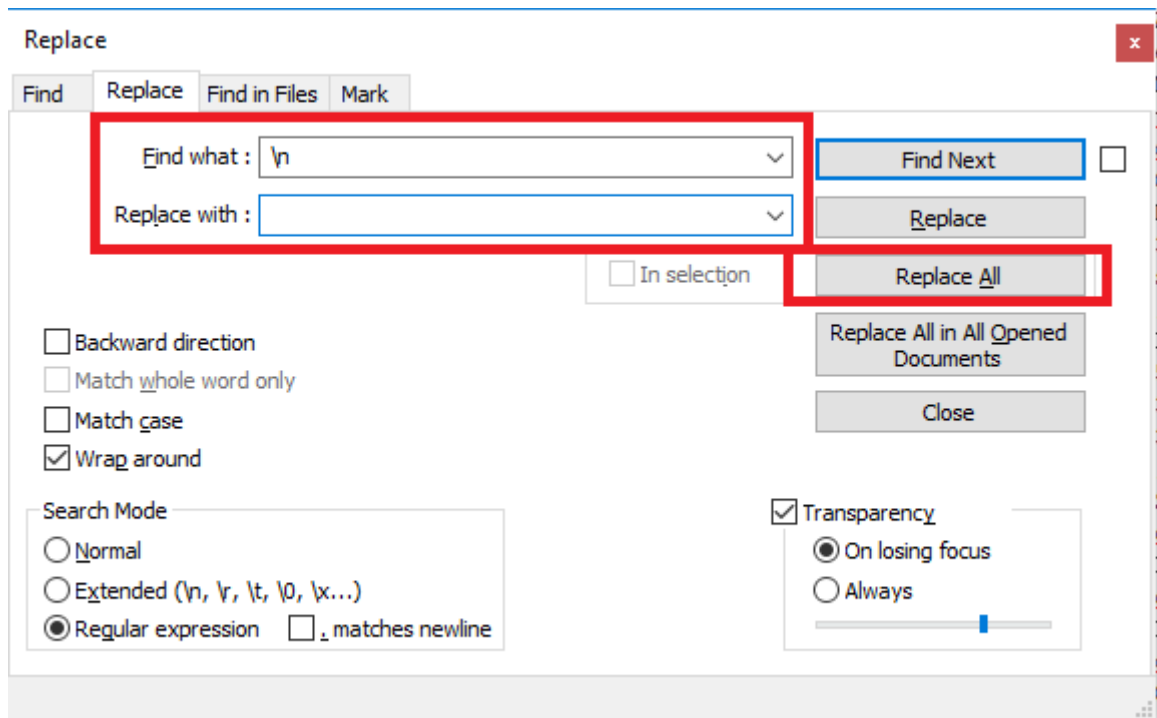
1 Date de dernière mise à jour (Source DGOS);"Médicaments sous ATU (
2 (Source ANSM)";"Substance active
3 (Source ANSM)";"Laboratoire
4 (Source ANSM)";"Statut de l'ATU octroyée (nominative/cohortes)
5 (source: ANSM)";Date de début de l'ATU (Source ANSM);Date de fin d
6 (Source CIP)";UCD 13 (Source CIP);Réserve hospitalière (oui/non)
7 19/02/2019;ABT-414 100 mg, poudre pour solution à diluer pour perf

```

Pour corriger cela on va utiliser l'outil de remplacement de texte.

Appuyer sur ctrl+F sur le clavier. Cela ouvre la fenêtre de « Recherche », mais il existe également un onglet « Replace », cliquez sur Replace.





Dans le champ « Find What » rentrez \n

Cela permet de repérer tous les sauts de ligne dans le fichier.

Puis dans Replace With assurez vous qu'il n'y ait aucun champ (si besoin cliquez dans le champ « Replace With » puis faites ctrl+A sur le clavier pour sélectionner tous les éléments du champ (au cas où il y ait un espace) puis appuyez sur la touche Supprime.

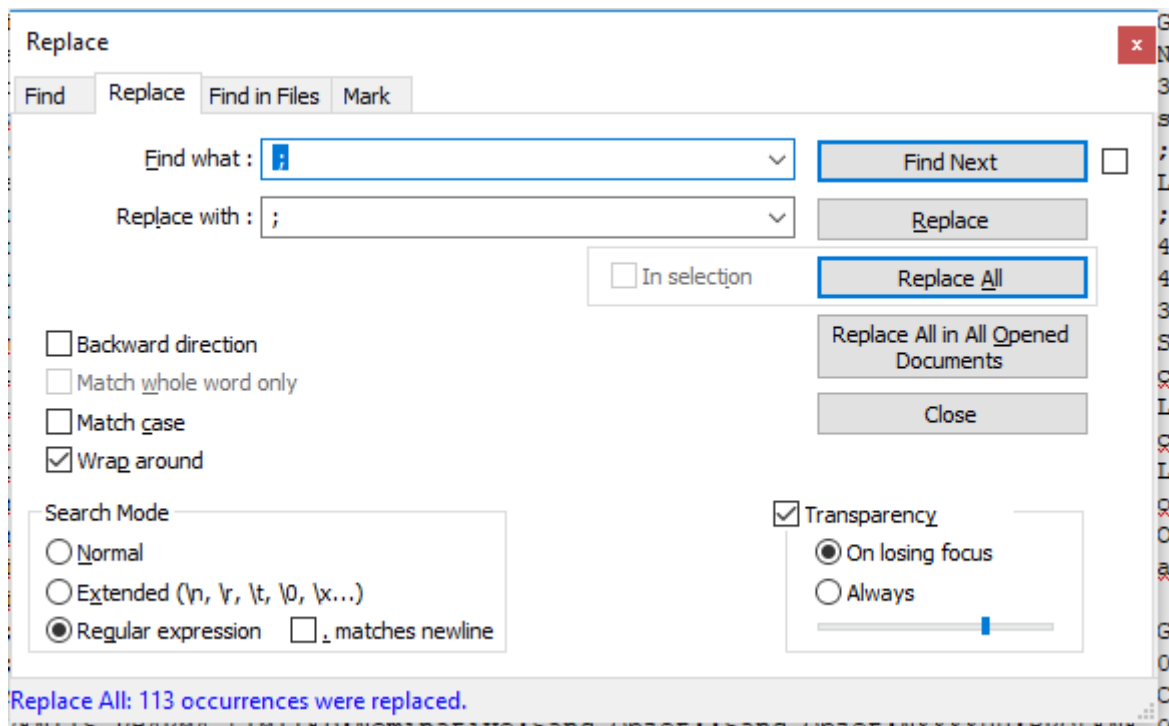
Enfin appuyez sur le bouton « Replace All ».

```
1 Date de dernière mise à jour (Source DGOS)
2 19/02/2019;ABT-414 100 mg, poudre pour sol
```

Vérifiez que l'entête tient maintenant sur une seule ligne.

\*8) On va continuer de traiter le fichier pour corriger les petits soucis de typo.

Toujours dans la fenêtre de Recherche/Remplacement (Ctrl+F si la fenêtre est fermée) :



Dans le champ « Find What » mettez « \_; » (avec \_ pour indiquer un espace).

Dans le champ « Replace with » mettez « ; » (sans espace)

\*9) Ensuite, copiez l'intégralité de la première ligne contenant les en-têtes et appuyez sur CTRL+N pour ouvrir un nouveau fichier, collez la première ligne et ouvrez la fenêtre de Recherche/Remplacement et remplacer « ; » par « \n » cela permet d'avoir une vue plus claire des en-têtes :

```

1  Date de dernière mise à jour (Source DGOS)
2  "Médicaments sous ATU (nom, dosage, forme pharmaceutique) (Source ANSM) "
3  "Substance active (Source ANSM) "
4  "Laboratoire (Source ANSM) "
5  "Statut de l'ATU octroyée (nominative/cohortes) (source: ANSM) "
6  Date de début de l'ATU (Source ANSM)
7  Date de fin de l'ATU (Source ANSM)
8  Indication de l'ATU de cohortes (Source ANSM)
9  Code indication (Source DGOS)
10 Libellé du code UCD (Source CIP)
11 "UCD 7 (Source CIP) "
12 UCD 13 (Source CIP)
13 Réserve hospitalière (oui/non) (Source ANSM)

```

Supprimez les « (Source XXX) ».

```

1 Date de dernière mise à jour
2 "Médicaments sous ATU (nom, dosage, forme pharmaceutique)"
3 "Substance active"
4 "Laboratoire"
5 "Statut de l'ATU octroyée (nominative/cohorte)"
6 Date de début de l'ATU
7 Date de fin de l'ATU
8 Indication de l'ATU de cohorte
9 Code indication
10 Libellé du code UCD
11 "UCD 7(Source CIP)"
12 UCD 13
13 Réserve hospitalière (oui/non)

```

Lors de cette étape, il est important de vérifier si les en-têtes correspondent exactement (sensible à la casse) à ceux du mapping de l'outil (cf. CSVModel\_TableNOM\_DE\_LA\_TABLE.cs) :

```

Server Explorer  Toolbox
Program.cs  CSVModel_TableATU.cs  X
C#  ToolInsertMedicamentATU  ToolInsertMedicamentATU.ToolSRC.Model.CSVModel_TableATU
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ToolInsertMedicamentATU.ToolSRC.Model
8  {
9      /// <summary>
10     /// Penser à faire une copie du csv source + supprimer les "(source DGOS / source ANSM)" dans les en-têtes de
11     /// Cette classe permet de binder/mapper les données selon les noms des colonnes du fichier CSV
12     /// </summary>
13     3 references
14     public class CSVModel_TableATU
15     {
16         [MappingCol("Date de dernière mise à jour")]
17         1 reference
18         public string DateMaJ
19         {
20             get; set;
21         }
22
23         [MappingCol("Médicaments sous ATU (nom, dosage, forme pharmaceutique)")]
24         1 reference
25         public string Medicament
26         {
27             get; set;
28         }
29     }
30 }

```

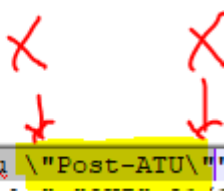
! Il est important qu'aucun guillemet / double quote (") ne soit présent dans le nom de la colonne, donc pensez à corriger le nom de l'entête de la colonne dans le fichier csv si c'est le cas)

Pour avoir ça :

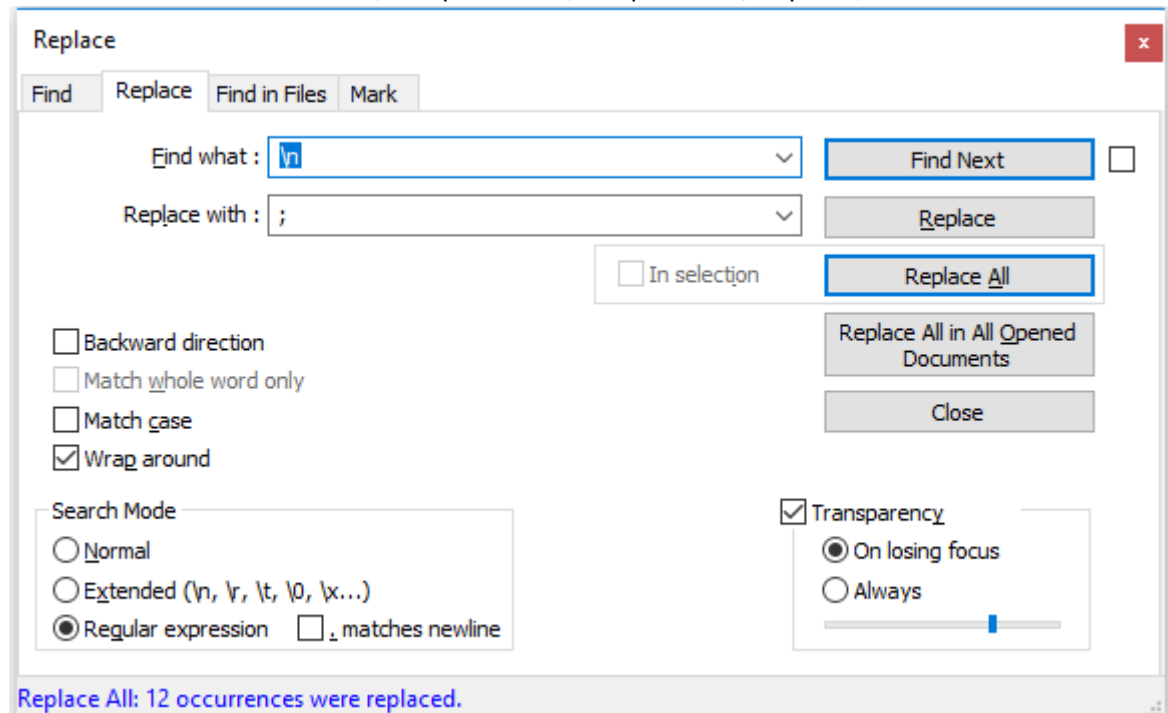
```
"Date de début de prise en charge au titre du Post-ATU";
```

Au lieu de ça :

```
"Date de début de prise en charge au titre du \"Post-ATU\"";
```

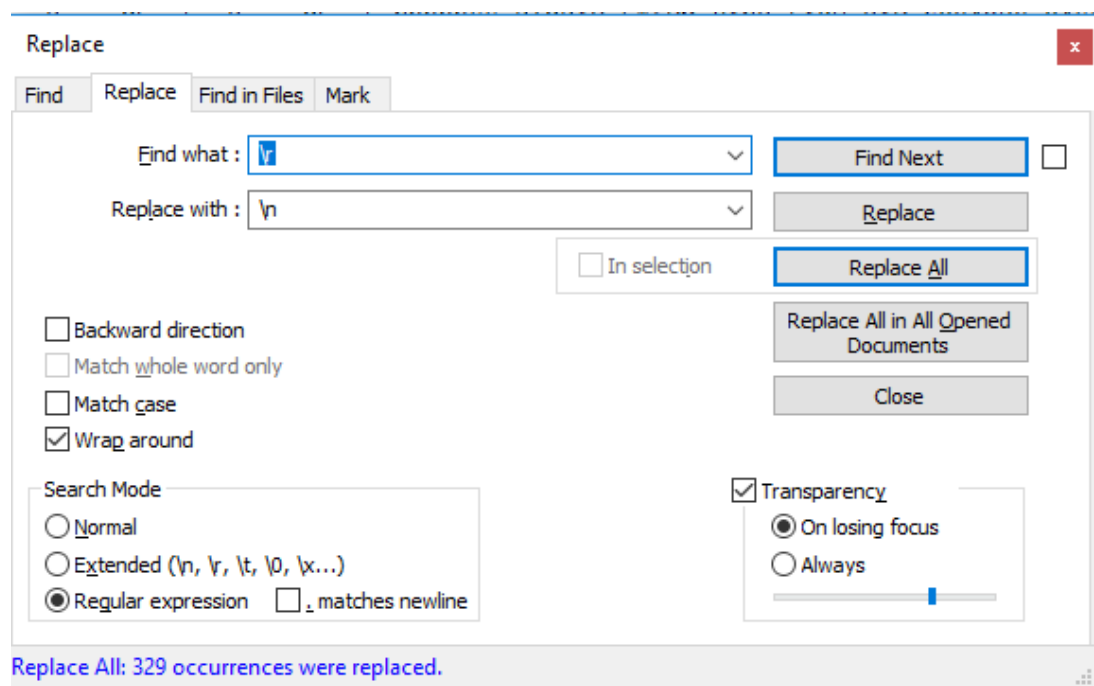


Puis dans la fenêtre Recherche/Remplacement, remplacez « \n » par « ; »



Puis copiez la ligne modifiée dans ce fichier et remplacez celle du fichier csv.

Dans le fichier csv finissez par remplacer `\r` par `\n` :

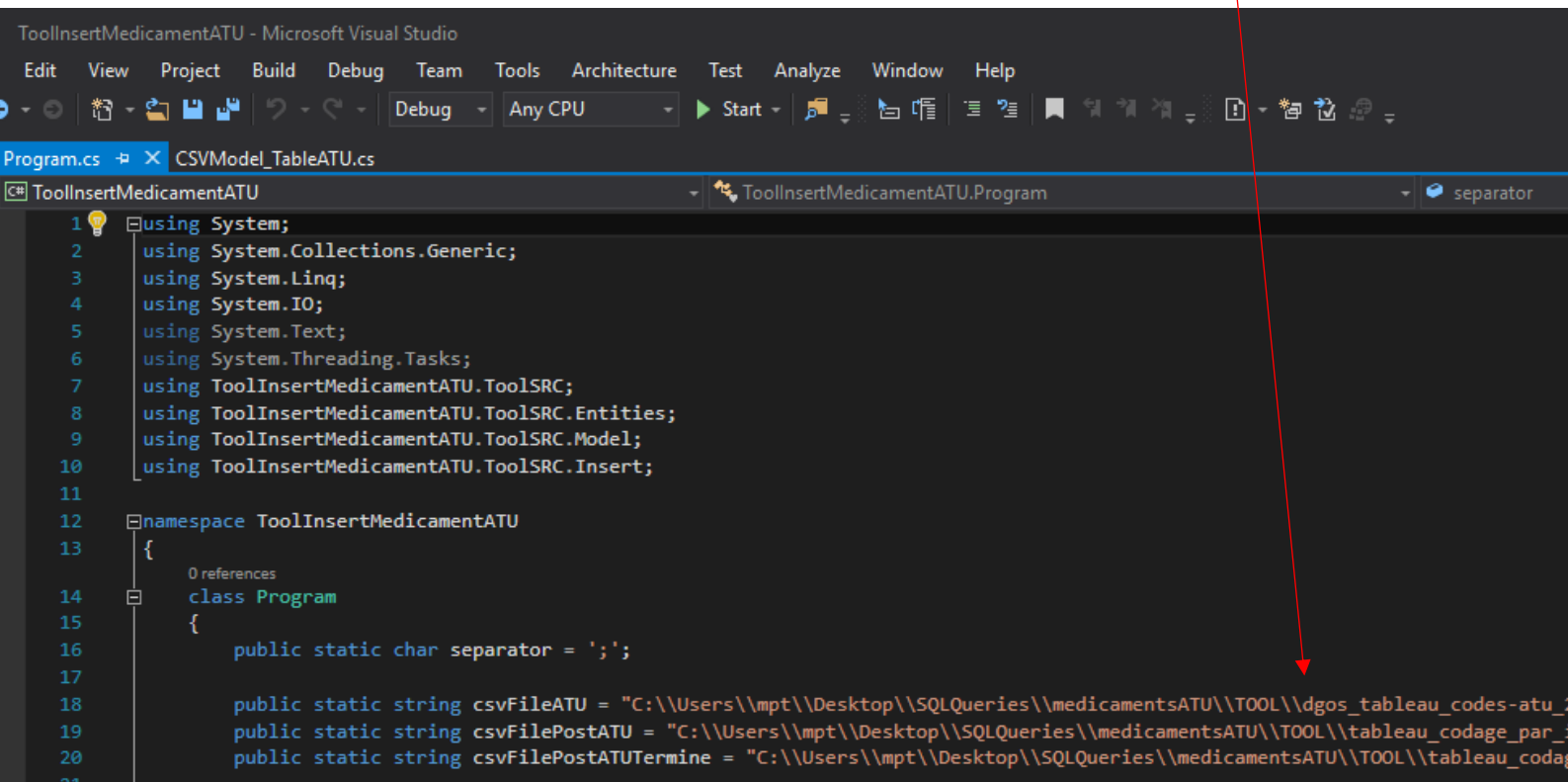


Enfin enregistrez le fichier csv.

Il devrait normalement être suffisamment préparé pour que l'outil génère correctement les tables nécessaires concernant ce fichier.

Faire de même pour tous les fichiers XLSX à convertir en CSV (A noter que si le fichier XLSX contient plusieurs feuilles, il faudra enregistrer autant de fichiers csv que de feuilles dans le fichier xlsx (et penser à bien sélectionner la feuille que l'on souhaite convertir en csv avant de commencer le processus de conversion en csv).

Enfin, vérifiez que l'outil pointe correctement sur les bon fichiers csv dans Program.cs :



```
ToolInsertMedicamentATU - Microsoft Visual Studio
Edit View Project Build Debug Team Tools Architecture Test Analyze Window Help
Program.cs CSVModel_TableATU.cs
ToolInsertMedicamentATU ToolInsertMedicamentATU.Program separator

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.IO;
5 using System.Text;
6 using System.Threading.Tasks;
7 using ToolInsertMedicamentATU.ToolsSRC;
8 using ToolInsertMedicamentATU.ToolsSRC.Entities;
9 using ToolInsertMedicamentATU.ToolsSRC.Model;
10 using ToolInsertMedicamentATU.ToolsSRC.Insert;
11
12 namespace ToolInsertMedicamentATU
13 {
14     0 references
15     class Program
16     {
17         public static char separator = ';';
18
19         public static string csvFileATU = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\dgos_tableau_codes-atu_2
20         public static string csvFilePostATU = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\tableau_codage_par_i
21         public static string csvFilePostATUTermine = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\tableau_codage
```

## II – Fonctionnement de l’outil :

L’outil est autonome et peut être lancé depuis le .exe si les liens et les noms de fichiers sont identiques à ceux pour lesquels on souhaite générer les scripts INSERT SQL.

Si l’on souhaite changer ces paramètres, il est alors nécessaire de venir modifier les paramètres dans **Program.cs** et de lancer le programme depuis Visual Studio (ou bien le recompiler pour générer le .exe qui correspond à vos liens pointant vers les fichiers):

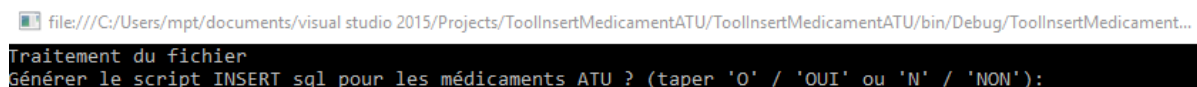
Possibilité de changer :

- Le séparateur ( ; )
- Le lien où se trouve le fichier csv de la liste de médicament ATU
- Le lien où se trouve le fichier csv de la liste de médicaments Post ATU
- Le lien où se trouve le fichier csv de la liste de médicaments Post ATU Terminé (feuille séparée du fichier xlsx et également convertie en format csv)

```
class Program
{
    public static char separator = ';';

    public static string csvFileATU = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\dgos_tableau_codes-atu_280219Exemple.csv";
    public static string csvFilePostATU = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\tableau_codage_par_indication_post-atu_1";
    public static string csvFilePostATUTermine = "C:\\Users\\mpt\\Desktop\\SQLQueries\\medicamentsATU\\TOOL\\tableau_codage_par_indication_pos
```

- Lancement de l’outil :



```
file:///C:/Users/mpt/documents/visual studio 2015/Projects/ToolInsertMedicamentATU/ToolInsertMedicamentATU/bin/Debug/ToolInsertMedicament...
Traitement du fichier
Générer le script INSERT sql pour les médicaments ATU ? (taper 'O' / 'OUI' ou 'N' / 'NON'):
```

Appuyer sur O ou bien tapez Oui pour générer les 2 premières tables (table *DonneesMedicamentsATU* et la table *ListeMedicamentsATU* qui contiennent toutes les informations nécessaires pour un médicament ATU)

```
file:///C:/Users/mpt/documents/visual studio 2015/Projects/ToolInsertMedicamentATU/ToolInsertMedicamentATU/bin/Debug/ToolInsertMedicament...
Traitement du fichier
Générer le script INSERT sql pour les médicaments ATU ? (taper 'O' / 'OUI' ou 'N' / 'NON') :
O
Path File:C:\Users\mpt\Desktop\SQLQueries\medicamentsATU\TOOL\dgos_tableau_codes-atu_280219Exemple.csv Total Lines:328
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
'Sans Objet' is not in the proper format.
Création script SQL Insert DonneesATU en cours
Générer le script INSERT sql pour les médicaments POST ATU ? (taper 'O' / 'OUI' ou 'N' / 'NON') :
```

La console affiche le nombre de lignes générées par le parser de CSV

Puis demande si l'utilisateur souhaite générer le fichier suivant (Médicaments Post ATU) : A noter que le fichier précédent d'insertion pour TableDonneesATU\_INSERT.sql se complète au fur et à mesure du programme si l'utilisateur appuie sur Oui ou O, il est donc préférable de toujours poursuivre la génération de sql par l'outil.

```
Générer le script INSERT sql pour les médicaments POST ATU ? (taper 'O' / 'OUI' ou 'N' / 'NON') :
O
Path File:C:\Users\mpt\Desktop\SQLQueries\medicamentsATU\TOOL\tableau_codage_par_indication_post-atu_1er_avril_2019corr1
V5.csv Total Lines:148
Création script SQL Insert MedicamentPOST_ATU en cours
Générer le script INSERT sql pour les médicaments POST ATU TERMINE ? (taper 'O' / 'OUI' ou 'N' / 'NON') :
O
Path File:C:\Users\mpt\Desktop\SQLQueries\medicamentsATU\TOOL\tableau_codage_par_indication_post-atu_1er_avril_2019corr1
Termine.csv Total Lines:88
Création script SQL Insert MedicamentPOST_ATU_Termine en cours
Fin du traitement, appuyez sur une touche pour quitter.
```

Idem pour le dernier fichier à générer qui complète le premier fichier de scripts SQL (TableDonneesATU\_INSERT.sql).

Les fichiers sont générés sur le bureau de l'utilisateur.

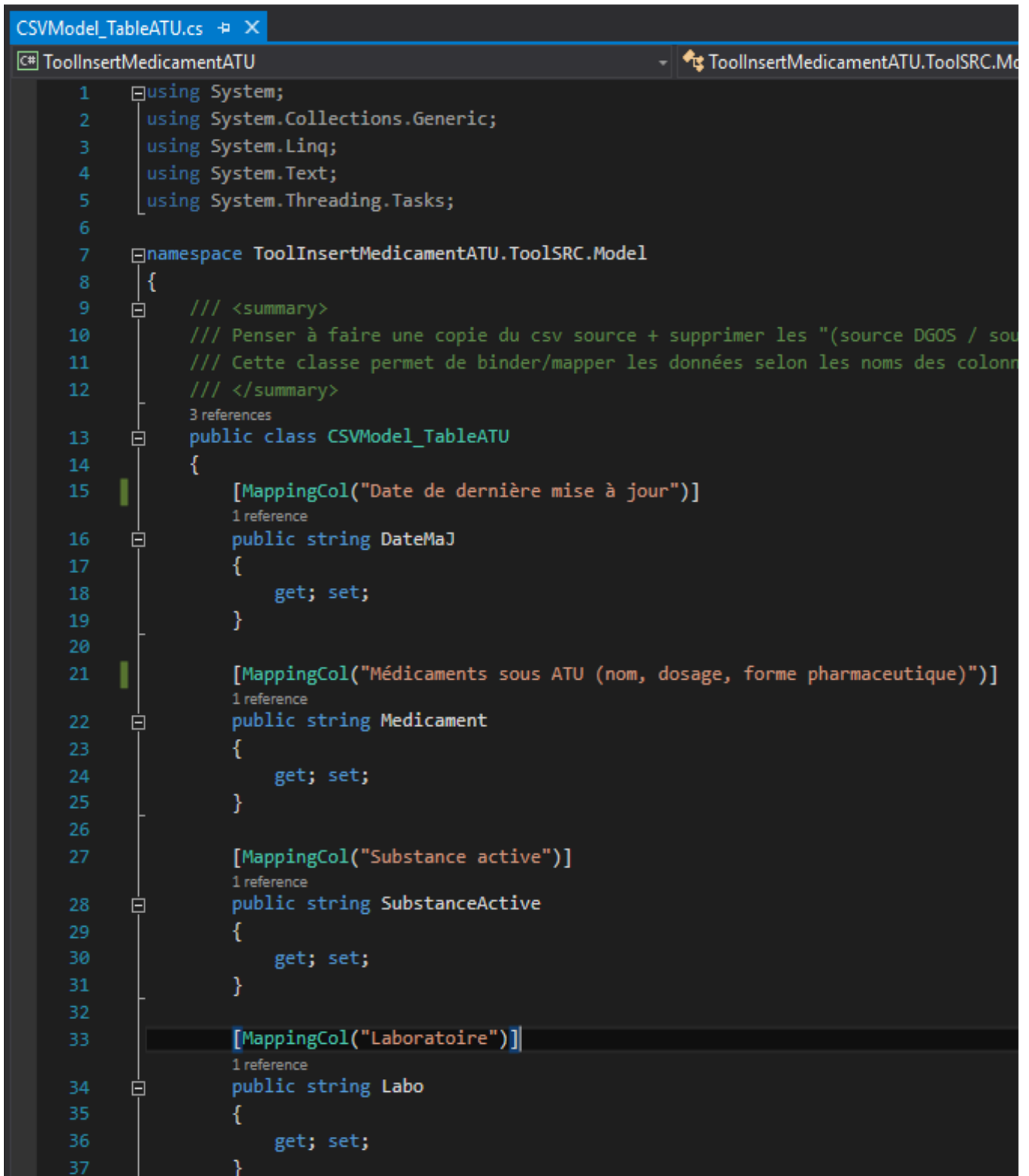
### III – Documentation Outil Génération Insert SQL

L'outil se compose d'un parser de CSV (**ParserCSV.cs**), et un parser de fichier (**FileParser.cs**) qui va venir mapper les différentes colonnes en Dictionnaires.

Pour que les colonnes se mappent correctement, l'outil utilise un Attribute « MappingColAttribute » (**MappingColAttribute.cs**) dont le rôle est de mapper les colonnes rentrées dans **Model/CSVModel\_TableNOM\_DE\_LA\_TABLE.cs** qui utilisent ce *MappingCol* comme en-tête.

Ci-dessous un exemple de mapping pour la table « Table ATU » en utilisant le MappingCol comme en-tête avec le string exact du nom de la colonne à mapper du fichier csv.

(fichier xlsx : **dgos\_tableau\_codes-atu\_280219.xlsx** )



```
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Text;
5  using System.Threading.Tasks;
6
7  namespace ToolInsertMedicamentATU.ToolSRC.Model
8  {
9      /// <summary>
10     /// Penser à faire une copie du csv source + supprimer les "(source DGOS / sou
11     /// Cette classe permet de binder/mapper les données selon les noms des colonn
12     /// </summary>
13     3 references
14     public class CSVModel_TableATU
15     {
16         [MappingCol("Date de dernière mise à jour")]
17         1 reference
18         public string DateMaJ
19         {
20             get; set;
21         }
22
23         [MappingCol("Médicaments sous ATU (nom, dosage, forme pharmaceutique)")]
24         1 reference
25         public string Medicament
26         {
27             get; set;
28         }
29
30         [MappingCol("Substance active")]
31         1 reference
32         public string SubstanceActive
33         {
34             get; set;
35         }
36
37         [MappingCol("Laboratoire")]
38         1 reference
39         public string Labo
40         {
41             get; set;
42         }
43     }
44 }
```



Dans **Program.cs**, je crée la fonction qui va venir désérialiser le fichier csv selon un type de « modèle » (le CSVModel\_TableNOM\_DE\_LA\_TABLE.cs mappé précédemment).

(exemple de fonction qui vient s'occuper de cette désérialisation : ExtractReport(string path, char separator) )

A noter qu'il faudra créer un ExtractReport différent pour chaque fichier csv ayant un modèle différent de données).

Pour générer les scripts SQL, je passe par des listes de classes nommées « Entities » dans Entities/InsertModel\_NOM\_DE\_LA\_TABLE (par exemple InsertModel\_TableATU.cs) qui vont venir contenir les informations nécessaires pour venir remplir chaque table dans la base de données.

J'utilise le constructeur de ces classes InsertModel\_XXX pour stocker les données récupérées à la lecture d'une ligne de CSV, puis j'ajoute cet objet à une liste correspondant à cet InsertModel.

1 reference

```
public InsertModel_TableListMedATU(int _idMedicament, string _ucd13, string _substanceActive, string _statutATU,
{
    IdMedicamentATU = _idMedicament;
    UCD13 = InsertTools.RemoveSpaceString(InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_ucd13)));
    SubstanceActive = InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_substanceActive));
    StatutATU = InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_statutATU));

    if (InsertTools.IsValidDate(InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_dateDebutATU)))
    {
        DateDebutATU = InsertTools.ConvertDateTimeFromString(InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_dateDebutATU)));
    }
    else
    {
        DateDebutATU = null;
    }

    if (InsertTools.IsValidDate(InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_dateFinATU)))
    {
        DateFinATU = InsertTools.ConvertDateTimeFromString(InsertTools.EscapeStringForSQLQuery(InsertTools.ConvertToUTF8String(_dateFinATU)));
    }
    else
    {
        DateFinATU = null;
    }
}
```

Les méthodes provenant de InsertTools servent à traiter les strings reçues par la lecture du csv (par exemple convertir le string en UTF8, s'assurer que les caractères soient bien « échappés » correctement pour éviter de casser le script SQL avec des ' impronptus lors de la création du script, vérifier la validité des dates, etc...

Une fois toutes les lignes du csv lues par le programme et toutes les listes correctement complétées, je fais appel à InsertSQLHandler.cs qui s'occupe de créer le fichier INSERT.sql sur le bureau de l'utilisateur et qui pour chaque objet de la liste de InsertModel va écrire le script d'insertion.

```

/// <summary>
/// Génération du string de la ligne d'Insert pour la table ListeMedicamentsATU
/// </summary>
/// <param name="line">Objet généré lors de la lecture du fichier csv depuis Program.cs</param>
/// <returns></returns>
1 reference
public string ComposeLineMedicamentsATUObj(InsertModel_TableListMedATU line)
{
    try
    {
        string insertInto = "INSERT INTO ListeMedicamentsATU (IdMedicamentATU, UCD13, SubstanceActive, StatutATU, DateDebutATU, DateFinATU)
            + "VALUES ("
            + line.IdMedicamentATU + ", "
            + line.UCD13 + ", "
            + line.SubstanceActive + ", "
            + line.StatutATU + ", "
            + line.DateDebutATU + ", "
            + line.DateFinATU
            + ")";

        return insertInto;
    }
    catch (System.Exception ex)
    {
        Console.WriteLine("Erreur lors de la création de la ligne dans le fichier d'insertion ListeMedicamentsATU : " + ex);
        return null;
    }
}

```

(Exemple de méthode générant la ligne d'insertion depuis un InsertModel)