

# Atoo-Sync GesCom - Custom ERP

## Table of Contents

### Configurer le profil

[Onglet Gestion Commerciale](#)

[Onglet Gestion Commerciale - Modifier les requêtes SQL ...](#)

[Onglet Gestion Commerciale - Editer le format](#)

### Liste des requêtes

[CUSTOMERS - CustomerExist](#)

[CUSTOMERS - CustomerFamillies](#)

[CUSTOMERS - CustomerGroups](#)

[CUSTOMERS - CustomerList](#)

[CUSTOMERS - CustomerSalesRepresentatives](#)

[CUSTOMERS - CustomersDisabled](#)

[CUSTOMERS - CustomersGDPR](#)

[CUSTOMERS - FindCustomerAccountByEmail](#)

[CUSTOMERS - FindCustomerAccountBySiret](#)

[CUSTOMERS - FindCustomerAccountByVatNumber](#)

[CUSTOMERS - ReadCustomer](#)

[CUSTOMERS - ReadCustomerContacts](#)

[CUSTOMERS - SelectCustomer](#)

[PENDINGPRODUCTS - PendingProductCustomerList](#)

[PENDINGPRODUCTS - ReadCustomerPendingProducts](#)

[PRICES - PriceGroupList](#)

[PRICES - ReadPriceGroup](#)

[PRODUCTS - CategoryList](#)

[PRODUCTS - CustomFieldList](#)

[PRODUCTS - Documents](#)

[PRODUCTS - GetCustomFieldValue](#)

[PRODUCTS - OtherProducts](#)

[PRODUCTS - ProductExist](#)

[PRODUCTS - ProductList](#)

[PRODUCTS - ProductPrices](#)

[PRODUCTS - ProductStates](#)

[PRODUCTS - ProductWarehouses](#)

[PRODUCTS - ReadCategory](#)

[PRODUCTS - ReadProduct](#)

[PRODUCTS - UpdateProductStockList](#)

[PRODUCTS - UpdateProductPriceList](#)

[SALES - AccountingJournals](#)

[SALES - Currencies](#)

[SALES - ReadOrder](#)

[SALES - InvoicePDFList](#)

[SALES - SalesNumbers](#)

[SALES - Settlements](#)

[SALES - Shippings](#)

[SALES - Warehouses](#)

[TAXES - AccountingCategories](#)

[TAXES - Taxes](#)

### **Configurer Atoo-Sync GesCom dans Shopify**

[Créer le champ de regroupement dans Shopify](#)

[Onglet Site Web : Option Shopify](#)

[Création d'une application privée dans Shopify](#)

## Configurer le profil

### Onglet Gestion Commerciale

Cet onglet permet de configurer la base de données, de paramétrer les requêtes SQL et de régler les fichiers d'export.

The screenshot shows a software window titled "Configurer le profil ODBC Custom ERP" with a close button (X) in the top right corner. On the left is a sidebar menu with icons and labels: Profil, Site Web, Gestion Commerciale (highlighted), Articles, Clients, Commandes, Import des commandes, Mise à jour des commandes, and Notification par email. The main area is titled "Gestion Commerciale" and "Configurer l'accès à la Gestion Commerciale". It contains a "Custom ERP" tab and several sections:

- Source ODBC**: "Base de données :" dropdown menu with "BIJOU SQL" selected.
- Nom d'utilisateur et mot de passe**: "Nom d'utilisateur :" and "Mot de passe :" text input fields, and a "Tester la connexion" button.
- Requêtes SQL**: A button "Modifier les requêtes SQL ..." with a green checkmark icon.
- Choisissez le dossier où enregistrer le fichier des clients**: "Fichier clients :" text box with a file path "C:\Users\Public\Fichiers Gescom export\Client.txt" and an "Editer le format" button; "Script à exécuter après la création :" text box.
- Choisissez le dossier où enregistrer le fichier des documents**: "Fichier des documents :" text box with a file path "C:\Users\Public\Fichiers Gescom export\Commande.txt" and an "Editer le format" button; "Script à exécuter après la création :" text box.
- Choisissez le dossier où enregistrer le fichier des lignes de documents**: "Fichier des lignes :" text box with a file path "C:\Users\Public\Fichiers Gescom export\Produit.txt" and an "Editer le format" button; "Script à exécuter après la création :" text box.

At the bottom, there are three buttons: "Options avancées...", "Appliquer", and "Fermer".

## Section Source ODBC

**Base de données** permet de choisir la base de donnée dans la liste des sources ODBC.

## Section Nom d'utilisateur et mot de passe

La zone texte **Nom d'utilisateur** permet de rentrer le nom de l'utilisateur pour se connecter à la base de données.

La zone texte **Mot de passe** permet de rentrer le mot de passe de l'utilisateur qui se connecte à la base de données.

Le bouton **Tester la connexion** permet de tester la connexion avec la base de données choisie en utilisant le **Nom d'utilisateur** et le **Mot de passe**.

## Section Requête SQL

Le bouton **Modifier les requête SQL** permet d'ouvrir une nouvelle fenêtre pour paramétrer les requêtes SQL de la base de données choisie et de renvoyer les champs requis pour le bon fonctionnement de l'application.

## Section Choisissez le dossier où sera enregistré le fichier client

**Fichier client** permet de choisir où sera enregistré le fichier client. Le bouton **Éditer le format** permet d'ouvrir la fenêtre de configuration du format dans lequel est enregistré le fichier exporté.

**Script à exécuter après la création** lance le script choisi **après** la création du fichier client.

## Section Choisissez le dossier où sera enregistré le fichier des documents

**Fichier des documents** permet de choisir où sera enregistré le fichier des documents. Le bouton **Éditer le format** permet d'ouvrir la fenêtre de configuration du format dans lequel est enregistré le fichier exporté.

**Script à exécuter après la création** lance le script choisi **après** la création du fichier des documents.

## Section choisissez le dossier où sera enregistré le fichier des lignes de documents

**Fichier des lignes** permet de choisir où sera enregistré le fichier des lignes de documents. Le bouton **Éditer le format** permet d'ouvrir la fenêtre de configuration de format dans lequel est enregistré le fichier exporté.

**Script à exécuter après la création** lance le script choisi **après** la création du fichier des documents.

## Onglet Gestion Commerciale - Modifier les requêtes SQL ...

Cette fenêtre sert à **définir les requêtes SQL** pour adapter les champs contenus dans la base de données aux données requises pour le bon fonctionnement de l'application.

**Modifier les requêtes SQL**

Editer la requête

Créer une requête vide | Tester la requête | Enregistrer les Requêtes | Fermer

Requête | Profil

**Requêtes**

- TAXES - GetTaxes
- TAXES - GetAccountingCategorie
- PRODUCTS - Warehouses
- PRODUCTS - ProductsStates
- PRODUCTS - Products
- PRODUCTS - ReadProduct
- PRODUCTS - ProductsNotInStock
- PRODUCTS - Documents
- PRODUCTS - StockItemList
- PRODUCTS - CustomFields
- PRODUCTS - Families
- PRODUCTS - ReadFamily**
- PRODUCTS - ProductFeaturesAsX
- CUSTOMERS - CustomerList
- CUSTOMERS - FindCustomerAcco
- CUSTOMERS - FindCustomerAcco
- CUSTOMERS - Customers
- CUSTOMERS - CustomersDisabl

**Requête: PRODUCTS - ReadFamily**

```

1  SELECT
2      CL_No as 'FamilyKey',
3      CL_NoParent as 'FamilyParentKey',
4      CL_Intitule as 'FamilyName',
5      " as FamilyDescription
6  FROM F_CATALOGUE
7  WHERE CL_No = '{0}'
    
```

**Champs obligatoires**

- FamilyKey : String
- FamilyParentKey : String
- FamilyName : String
- FamilyDescription : String

**Informations**

Requête permettant de retourner les données d'une famille d'article.

**Résultats**

Family Key	Family Parent K...	Family Name	Family Descripti...
2	1	Appareils	

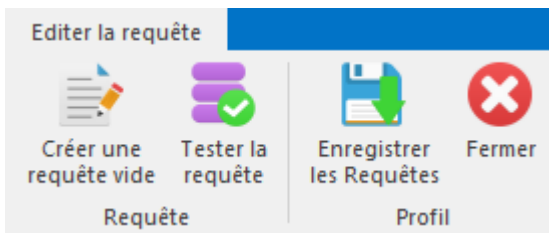
Résultats | Requête SQL

**Paramètres**

Nom	Valeur de test
FamilyKey	2

{0} = FamilyKey (String)

## Bandeau Éditer la requête



## Section Requête

Ces deux boutons sont utilisés sur la requête sélectionnée dans le panneau **Requêtes**.

**Créer une requête vide** remplace la requête en cours par un aperçu de la requête avec les paramètres et alias des champs attendus.

Exemple pour la requête "vide" **PRODUCT - Products** qui sert à retourner la liste des articles pour la sélection et l'exportation dans la boutique :

```
Requête: PRODUCTS - Products
1  SELECT
2  ProductKey as 'ProductKey' /* Type: String */,
3  ProductName as 'ProductName' /* Type: String */
4  FROM TABLE
5  WHERE
6  FilterDateModification = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

Le bouton **Tester la requête** permet de lancer l'exécution de la requête pour s'assurer de son bon fonctionnement. Le résultat de la requête s'affiche dans le panneau **Résultats**.

## Section Profil

Le bouton **Enregistrer les requêtes** enregistre la totalité des requêtes disponibles.

Le bouton **Fermer** ferme la fenêtre **Modifier les requêtes** sans sauvegarder les requêtes qui ont été modifiées.

## Panneau Requêtes



Ce panneau **Requêtes** affiche toutes les requêtes dont l'application a besoin pour son fonctionnement.

Le fait de sélectionner une requête dans la liste, affiche cette requête dans la partie **zone de texte Requête** et permet de la modifier si besoin.

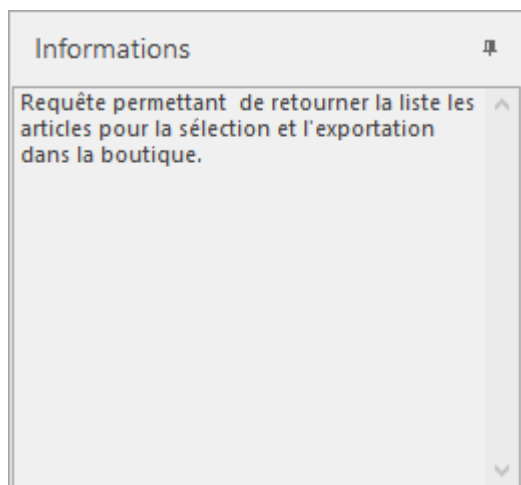
Les requêtes sont classées par catégorie, en tête de liste se trouvent les **taxes**, puis les **articles** (products), les **clients** (customers) et en fin les **ventes** (sales).

Quand la couleur de la requête dans la liste est **bleu** cela indique que tous les champs et paramètres obligatoires sont contenus dans la requête.

**Note** : La couleur bleu n'indique pas que la requête est bonne ou qu'elle renvoie un résultat correct. Le test des requêtes avec le bouton **Tester la requête** est vivement conseillé.

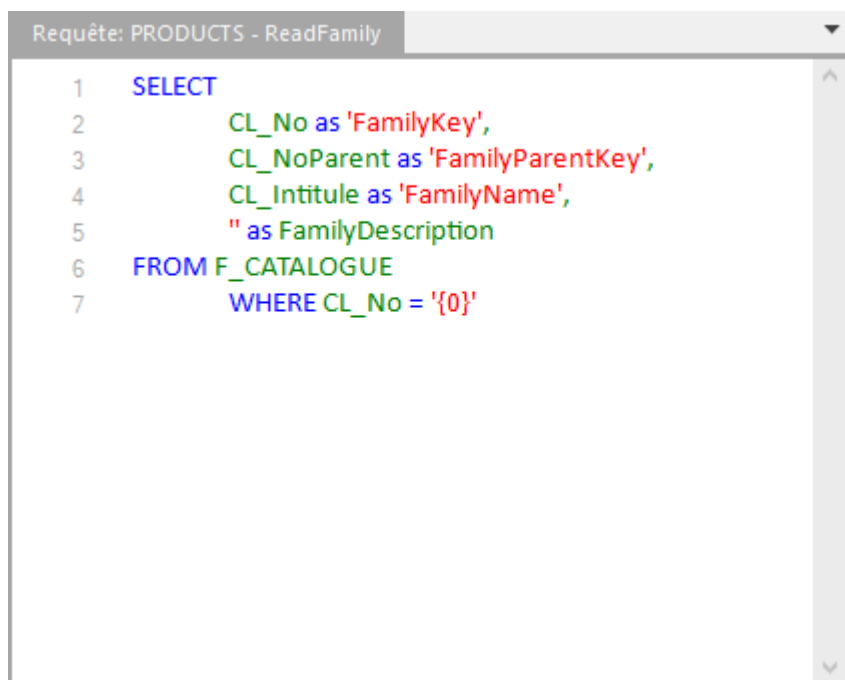
Quand la couleur de la requête est **rouge**, cela indique que des champs ou paramètres obligatoires manquent dans la requête.

## Panneau Informations



Le panneau **Informations** fournit des informations sur la requête sélectionnée dans le panneau **Requête**. Ces informations permettent de comprendre l'utilité de la requête et de ce qu'elle doit renvoyer.

## Zone de texte Requête



Cette zone affiche la requête en cours et permet de la modifier (*qui est sélectionné dans le panneau **Requêtes***).

## Panneau Champs obligatoires



Ce panneau indique les champs obligatoires avec leurs noms et types qui doivent être inclus dans la requête sélectionnée.

Les champs en **bleu** indiquent que les champs figurent dans la requête.

Un champ **rouge** signifie que le champ est manquant dans la requête.

## Panneau Paramètres



Ce panneau indique les paramètres à insérer dans la requête avec la possibilité sur certaines de rentrer des valeurs tests en vue de vérifier son bon fonctionnement grâce au bouton **Tester la requête**.

La section basse indique les représentations des paramètres à insérer (`{0}`, `{1}`, `{2}`, etc...) et les types de données qui sont transmis dans la requête.

Exemple avec la requête **PRODUCT - ReadFamily** (*retourne les données d'une famille d'articles*) :

<pre> 1 SELECT 2     CL_No as 'FamilyKey', 3     CL_NoParent as 'FamilyParentKey', 4     CL_Intitule as 'FamilyName', 5     " as FamilyDescription 6 FROM F_CATALOGUE 7 WHERE CL_No = '{0}'                 </pre>	<pre> FamilyKey : String FamilyParentKey : String FamilyName : String FamilyDescription : String                 </pre>
--	---

Paramètres	
Nom	Valeur de test
FamilyKey	
{0} = FamilyKey (String)	

Autre exemple avec la requête **PRODUCT - Products** (retourne la liste des articles pour la sélection

<p>Requête: PRODUCTS - Products</p> <pre> 1 SELECT 2     AR_Ref as 'ProductKey', 3     AR_Design as 'ProductName' 4 FROM F_Article 5 WHERE cbModification &gt;= '{0}'                 </pre>	<p>Champs obligatoires</p> <pre> ProductKey : String ProductName : String                 </pre>
--	--

Paramètres	
Nom	Valeur de test
FilterDateModification	29/05/2020
{0} = FilterDateModification (Date) (format: 'yyyy-MM-ddTHH:mm:ss')	

### Panneau Résultat

Ce panneau comprend deux onglets.

L'onglet **Résultats** renvoie le résultat de la requête en cours après avoir cliqué sur le bouton **Tester la requête**.

Résultats			
Family Key	Family Name	Family Parent Key	
31	Entretien	30	^
32	Formation	30	
25	Lingots	24	
33	Location	30	
36	Maintenance	30	v

Résultats Requête SQL

L'onglet **Requête SQL** permet de visualiser et si besoin, de copier la vraie requête exécutée pendant le test.

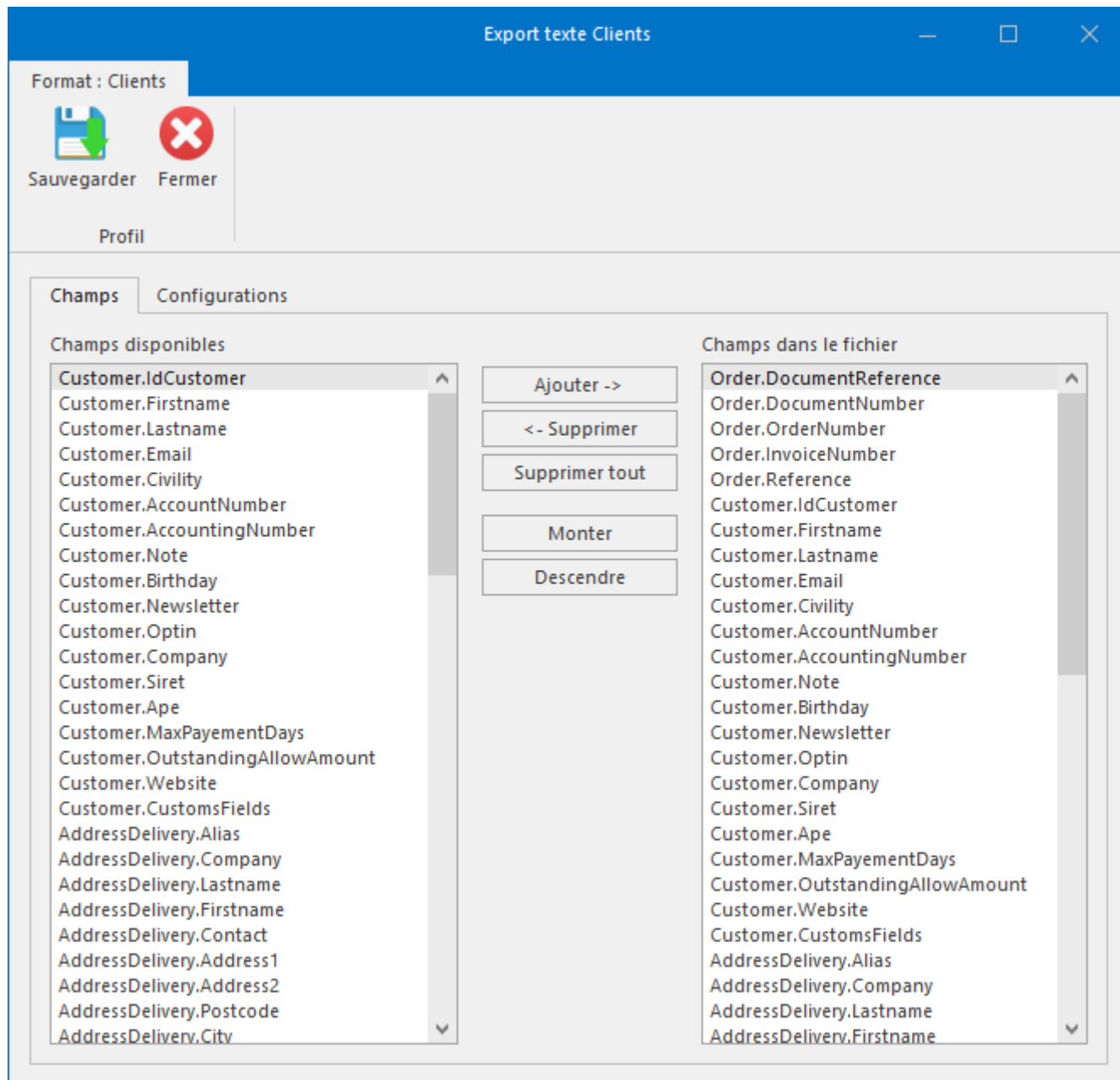
```

Requête SQL
SELECT
    AR_Ref as 'ProductKey',
    AR_Design as 'ProductName'
FROM F_Article
WHERE cbModification >= '2020-05-29T17:19:11'
    
```

Résultats Requête SQL

## Onglet Gestion Commerciale - Editer le format

Cette fenêtre permet de configurer les formats d'export textes des **clients**, des **documents** et des **lignes de documents**. Les possibilités de format sont identiques sur les trois types de formats (*clients, documents, lignes de documents*). Le type de format d'export est indiqué sur le titre et sur l'onglet **Format** du bandeau de la fenêtre.



Le bouton **Sauvegarder** enregistre le format.

Le bouton **Fermer** ferme la fenêtre sans sauvegarder.

## Onglet Champs



La liste de gauche **Champs disponibles** répertorie toute la liste de champs que l'on peut ajouter au format.

La liste de droite **Champs dans le fichier** indique tous les champs qui sont inclus dans le format d'export.

Entre ces deux listes se trouvent les boutons de manipulation des champs du format.

Le bouton **Ajouter** permet d'ajouter le champ sélectionné dans la liste **Champs disponibles** à la suite du champ sélectionné dans la liste **Champs dans le fichier**.

Le bouton **Supprimer** retire le champ sélectionné de la liste **Champs dans le fichier**.

Le bouton **Supprimer tout** enlève tous les champs contenus dans la liste **Champs dans le fichier**.

Le bouton **Monter** monte d'un niveau le champ sélectionné dans la liste **Champs dans le fichier** et permet ainsi d'organiser l'ordre des champs dans le fichier d'export.

Le bouton **Descendre** descend d'un niveau le champ sélectionné dans la liste **Champs dans le fichier**.

## Onglet Configuration

Champs **Configurations**

**Fichier**  
Séparateur d'enregistrement : Return/LineFeed  
Séparateur des champs : ;  
Délimiteur de texte :  

---

**Format des dates et des nombres**  
Format de la date : dd/MM/yyyy  
mercredi 29 juillet 2020 -> 29/07/2020  
Format des nombres : #0.00  
Séparateur décimal : .  
Séparateur de millier :  
12345,56789 -> 12345.57  

---

**Options**  
 Insérer des en-têtes

### Section Fichier

La liste déroulante **Séparateur d'enregistrement** sélectionne quel séparateur délimite chaque enregistrement.

La liste déroulante **Séparateur de champs** permet de choisir ou de spécifier le séparateur de chaque champ se trouvant dans les enregistrements.

**Délimiteur de texte** définit les caractères qui sont insérés en début et en fin de champs.

### Section Format des dates et des nombres

La liste déroulante **Format de la date** permet de choisir parmi une liste de formats les plus courants, comment les dates sont formatées ou d'en définir un personnalisé.

La liste déroulante **Format des nombres** permet de choisir ou de spécifier le format des nombres.

Les zones de texte **Séparateur décimal** et **Séparateur de milliers** remplace les paramètres linguistiques de séparateur décimal et millier défini dans Windows pour les nombres contenus dans le fichier.

**Note** : Le séparateur décimal est représenté par un point dans les systèmes anglo-saxons et par une virgule dans les autres systèmes

## Section Options

**Insérer des en-têtes** active l'insertion en début de fichier, des noms des champs qui sont contenus dans le format du fichier.

## Liste des requêtes

### CUSTOMERS - CustomerExist

Cette requête permet de rechercher si le code client existe déjà dans l'ERP.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** String le code unique du client l'ERP (exemple: C0001)

### Argument obligatoire

- **{0}** Le code client du client trouvé dans la boutique ou généré par l'application Atoo-Sync (exemple: C0001)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
WHERE
  CustomerKey = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - CustomerFamillies

Cette requête permet de retourner la liste des familles clients de l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

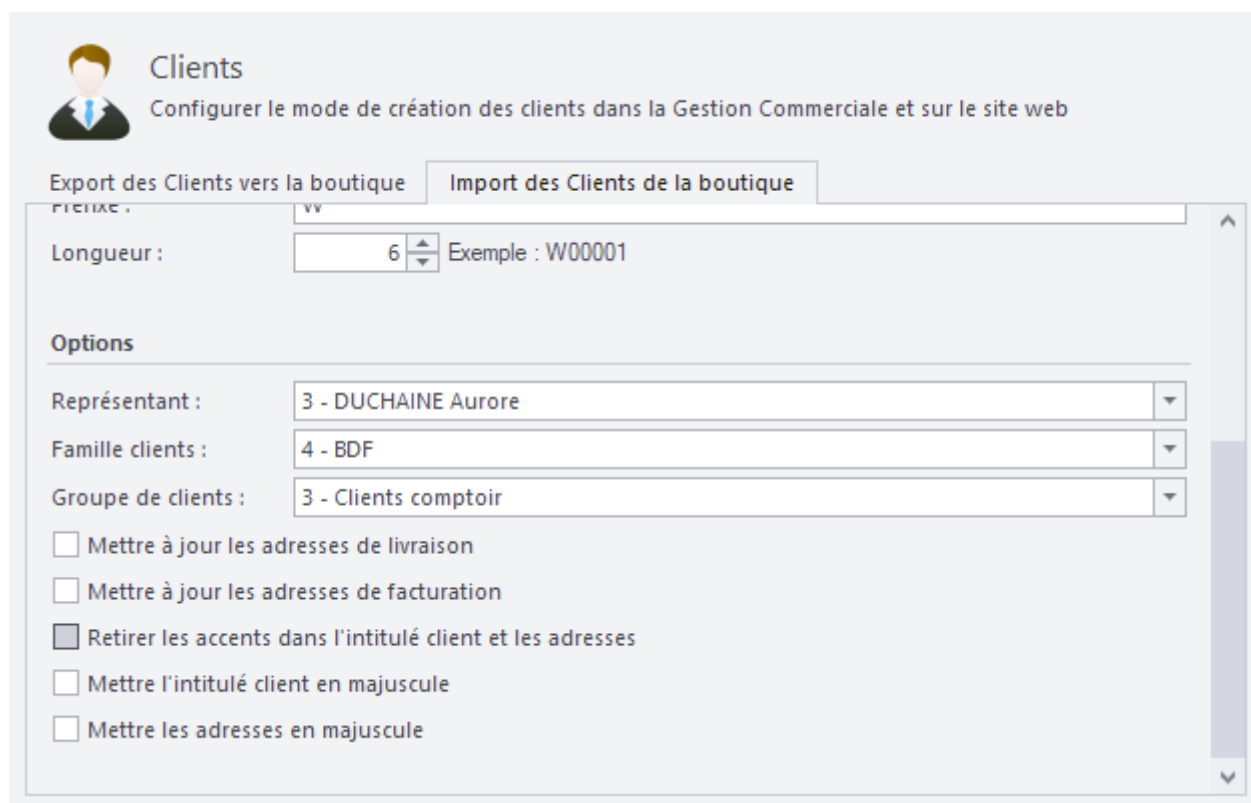
- **CustomerFamilyKey** String le code unique de la famille client dans l'ERP (exemple: FC01)
- **CustomerFamilyName** String le nom de la famille client dans l'ERP (exemple: Revendeur )

### Exemple de requête

```
SELECT
  CustomerFamilyKey as 'CustomerFamilyKey' /* Type: String */,
  CustomerFamilyName as 'CustomerFamilyName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration de la création des clients dans l'ERP.



**Clients**  
Configurer le mode de création des clients dans la Gestion Commerciale et sur le site web

Export des Clients vers la boutique | Import des Clients de la boutique

Longueur : 6 Exemple : W00001

**Options**

Représentant : 3 - DUCHAINE Aurore

Famille clients : 4 - BDF

Groupe de clients : 3 - Clients comptoir

Mettre à jour les adresses de livraison

Mettre à jour les adresses de facturation

Retirer les accents dans l'intitulé client et les adresses

Mettre l'intitulé client en majuscule

Mettre les adresses en majuscule

## CUSTOMERS - CustomerGroups

Cette requête permet de retourner la liste des groupes de clients de l'ERP.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CustomerGroupKey** **String** le code unique du groupe de clients dans l'ERP (exemple: G01)
- **CustomerGroupName** **String** le nom du groupe de clients dans l'ERP (exemple: Province )

## Exemple de requête

```
SELECT
  CustomerGroupKey as 'CustomerGroupKey' /* Type: String */,
  CustomerGroupName as 'CustomerGroupName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration de la création des clients dans l'ERP.



**Clients**  
Configurer le mode de création des clients dans la Gestion Commerciale et sur le site web

Export des Clients vers la boutique | Import des Clients de la boutique

Longueur : 6 Exemple : W00001

**Options**

Représentant : 3 - DUCHAINE Aurore

Famille clients : 4 - BDF

Groupe de clients : 3 - Clients comptoir

Mettre à jour les adresses de livraison

Mettre à jour les adresses de facturation

Retirer les accents dans l'intitulé client et les adresses

Mettre l'intitulé client en majuscule

Mettre les adresses en majuscule

## CUSTOMERS - CustomerList

Cette requête permet de retourner la liste des clients à exporter dans la boutique.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation pour filtrer les articles

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
WHERE
  FilterDateModification = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## CUSTOMERS - CustomerSalesRepresentatives

Cette requête permet de retourner la liste des commerciaux des clients de l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :


- **CustomerSalesRepresentativeKey** String le code unique du commercial dans l'ERP (exemple: R01)
- **CustomerSalesRepresentativeFirstName** String le prénom du commercial dans l'ERP (exemple: Sophie)
- **CustomerSalesRepresentativeLastName** String le nom du commercial dans l'ERP (exemple: LATOUR)

### Exemple de requête

```
SELECT
  CustomerSalesRepresentativeKey as 'CustomerSalesRepresentativeKey' /* Type: String */,
  CustomerSalesRepresentativeFirstName as 'CustomerSalesRepresentativeFirstName' /*
Type: String */,
  CustomerSalesRepresentativeLastName as 'CustomerSalesRepresentativeLastName' /*
Type: String */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration de la création des clients dans l'ERP.



### Clients

Configurer le mode de création des clients dans la Gestion Commerciale et sur le site web

Export des Clients vers la boutique
Import des Clients de la boutique

Longueur :  Exemple : W00001

**Options**

Représentant :

Famille clients :

Groupe de clients :

Mettre à jour les adresses de livraison

Mettre à jour les adresses de facturation

Retirer les accents dans l'intitulé client et les adresses

Mettre l'intitulé client en majuscule

Mettre les adresses en majuscule

## CUSTOMERS - CustomersDisabled

Cette requête permet de retourner une liste de clients issus de l'ERP pour les désactiver dans la boutique web.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
```

## CUSTOMERS - CustomersGDPR

Cette requête permet de retourner une liste de clients issus de l'ERP pour activer l'option RGPD et supprimer les données dans la boutique.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
```

## CUSTOMERS - FindCustomerAccountByEmail

Cette requête permet de rechercher si le client n'existe pas déjà dans l'ERP. La recherche est effectuée sur l'adresse email du client associé à la commande.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Argument obligatoire

- `{0}` L'adresse email du client de la commande de la boutique (exemple: john@doe.com)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
WHERE
  CustomerEmail = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - FindCustomerAccountBySiret

Cette requête permet de rechercher si le client n'existe pas déjà dans l'ERP. La recherche est effectuée sur le numéro de Siret du client associé à la commande.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Argument obligatoire

- `{0}` Le numéro de Siret du client de la commande de la boutique (exemple: 425499428)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
WHERE
  CustomerSiret = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - FindCustomerAccountByVatNumber

Cette requête permet de rechercher si le client n'existe pas déjà dans l'ERP. La recherche est effectuée sur le numéro de TVA du client associé à la commande.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

### Argument obligatoire

- `{0}` Le numéro de TVA du client de la commande de la boutique (exemple: FR41458745962)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
WHERE
  CustomerVatNumber = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - ReadCustomer

Cette requête permet de lire les données d'un client pour le créer ou le modifier dans la boutique. Pour chaque code client retourné par la requête **Ccustomers** Atoo-Sync crée une requête. C'est cette requête qui permet d'envoyer les données du client dans la boutique.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CustomerKey** **String** le code unique du client dans l'ERP (exemple: C0001)
- **PriceGroupKey** **String** le code unique du groupe de tarifs dans l'ERP (exemple: TARIF1)
- **CustomerCompany** **String** la société du client dans l'ERP (exemple: ATOO NEXT)
- **Customerfirstname** **String** le prénom du client dans l'ERP (exemple: Jean)
- **CustomerLastname** **String** le nom du client dans l'ERP (exemple: DUPONT)
- **CustomerEmail** **String** l'adresse email client dans l'ERP (exemple: jean.dupont@atoo-next.com)
- **CustomerSiret** **String** le numéro de SIRET du client dans l'ERP (exemple: 425499428)
- **CustomerApe** **String** le numéro d'APE du client dans l'ERP (exemple: 6021Z)
- **CustomerVatNumber** **String** le numéro de TVA du client dans l'ERP (exemple: FR41458745962)
- **CustomerSiret** **String** le numéro de SIRET du client dans l'ERP (exemple: 425499428)
- **CustomerWebsite** **String** l'URL du site web du client dans l'ERP (exemple: <https://www.atoo-next.net>)
- **CustomerNote** **String** les notes du client dans l'ERP
- **CustomerInvoiceName** **String** le nom de l'adresse de facturation dans l'ERP (exemple: Facturation)
- **CustomerInvoiceAddress1** **String** l'adresse 1 de l'adresse de facturation
- **CustomerInvoiceAddress2** **String** l'adresse 2 de l'adresse de facturation
- **CustomerInvoicePostcode** **String** le code postal de l'adresse de facturation
- **CustomerInvoiceCity** **String** la ville de l'adresse de facturation
- **CustomerInvoicePhone** **String** le numéro de téléphone de l'adresse de facturation
- **CustomerInvoiceState** **String** le département de l'adresse de facturation
- **CustomerInvoiceCountry** **String** le pays de l'adresse de facturation
- **CustomerDeliveryName** **String** le nom de l'adresse de livraison dans l'ERP (exemple: Livraison)
- **CustomerDeliveryAddress1** **String** l'adresse 1 de l'adresse de livraison
- **CustomerDeliveryAddress2** **String** l'adresse 2 de l'adresse de livraison
- **CustomerDeliveryPostcode** **String** le code postal de l'adresse de livraison
- **CustomerDeliveryCity** **String** la ville de l'adresse de livraison
- **CustomerDeliveryPhone** **String** le numéro de téléphone de l'adresse de livraison
- **CustomerDeliveryState** **String** le département de l'adresse de livraison
- **CustomerDeliveryCountry** **String** le pays de l'adresse de livraison

## Argument obligatoire

- {0} spécifie le code unique du client (exemple: C0001)

## Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */,
  PriceGroupKey as 'PriceGroupKey' /* Type: String */,
  CustomerCompany as 'CustomerCompany' /* Type: String */,
  Customerfirstname as 'Customerfirstname' /* Type: String */,
  CustomerLastname as 'CustomerLastname' /* Type: String */,
  CustomerEmail as 'CustomerEmail' /* Type: String */,
  CustomerSiret as 'CustomerSiret' /* Type: String */,
  CustomerApe as 'CustomerApe' /* Type: String */,
  CustomerVatNumber as 'CustomerVatNumber' /* Type: String */,
  CustomerWebsite as 'CustomerWebsite' /* Type: String */,
  CustomerNote as 'CustomerNote' /* Type: String */,
  CustomerInvoiceName as 'CustomerInvoiceName' /* Type: String */,
  CustomerInvoiceAddress1 as 'CustomerInvoiceAddress1' /* Type: String */,
  CustomerInvoiceAddress2 as 'CustomerInvoiceAddress2' /* Type: String */,
  CustomerInvoicePostcode as 'CustomerInvoicePostcode' /* Type: String */,
  CustomerInvoiceCity as 'CustomerInvoiceCity' /* Type: String */,
  CustomerInvoicePhone as 'CustomerInvoicePhone' /* Type: String */,
  CustomerInvoiceState as 'CustomerInvoiceState' /* Type: String */,
  CustomerInvoiceCountry as 'CustomerInvoiceCountry' /* Type: String */,
  CustomerDeliveryName as 'CustomerDeliveryName' /* Type: String */,
  CustomerDeliveryAddress1 as 'CustomerDeliveryAddress1' /* Type: String */,
  CustomerDeliveryAddress2 as 'CustomerDeliveryAddress2' /* Type: String */,
  CustomerDeliveryPostcode as 'CustomerDeliveryPostcode' /* Type: String */,
  CustomerDeliveryCity as 'CustomerDeliveryCity' /* Type: String */,
  CustomerDeliveryPhone as 'CustomerDeliveryPhone' /* Type: String */,
  CustomerDeliveryState as 'CustomerDeliveryState' /* Type: String */,
  CustomerDeliveryCountry as 'CustomerDeliveryCountry' /* Type: String */
FROM TABLE
WHERE
  CustomerKey = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - ReadCustomerContacts

Cette requête permet de lire les contacts d'un client pour les créer ou les modifier dans la boutique. Pour chaque code client retourné par la requête **Customers** Atoo-Sync crée une requête. C'est cette requête qui permet d'envoyer les contacts du client dans la boutique.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ContactKey** `String` le code unique du contact dans l'ERP (exemple: CC0001)
- **ContactCompany** `String` la société du contact dans l'ERP (exemple: ATOO NEXT)
- **ContactFirstname** `String` le prénom du contact dans l'ERP (exemple: Jean)
- **ContactLastname** `String` le nom du contact dans l'ERP (exemple: DUPONT)
- **ContactEmail** `String` l'adresse email contact dans l'ERP (exemple: jean.dupont@atoo-next.com)
- **ContactPhone** `String` le numéro de téléphone du contact dans l'ERP
- **ContactPhoneMobile** `String` le numéro de téléphone portable du contact dans l'ERP
- **ContactInvoiceName** `String` le nom de l'adresse de facturation dans l'ERP (exemple: Facturation)
- **ContactInvoiceAddress1** `String` l'adresse 1 de l'adresse de facturation
- **ContactInvoiceAddress2** `String` l'adresse 2 de l'adresse de facturation
- **ContactInvoicePostcode** `String` le code postal de l'adresse de facturation
- **ContactInvoiceCity** `String` la ville de l'adresse de facturation
- **ContactInvoicePhone** `String` le numéro de téléphone de l'adresse de facturation
- **ContactInvoiceState** `String` le département de l'adresse de facturation
- **ContactInvoiceCountry** `String` le pays de l'adresse de facturation
- **ContactDeliveryName** `String` le nom de l'adresse de livraison dans l'ERP (exemple: Livraison)
- **ContactDeliveryAddress1** `String` l'adresse 1 de l'adresse de livraison
- **ContactDeliveryAddress2** `String` l'adresse 2 de l'adresse de livraison
- **ContactDeliveryPostcode** `String` le code postal de l'adresse de livraison
- **ContactDeliveryCity** `String` la ville de l'adresse de livraison
- **ContactDeliveryPhone** `String` le numéro de téléphone de l'adresse de livraison
- **ContactDeliveryState** `String` le département de l'adresse de livraison
- **ContactDeliveryCountry** `String` le pays de l'adresse de livraison

## Argument obligatoire

- `{0}` spécifie le code unique du client (exemple: C0001)

## Exemple de requête

```
SELECT
  ContactKey as 'ContactKey' /* Type: String */,
  ContactCompany as 'ContactCompany' /* Type: String */,
```

```
ContactFirstname as 'ContactFirstname' /* Type: String */,
ContactLastname as 'ContactLastname' /* Type: String */,
ContactEmail as 'ContactEmail' /* Type: String */,
ContactPhone as 'ContactPhone' /* Type: String */,
ContactPhoneMobile as 'ContactPhoneMobile' /* Type: String */,
ContactInvoiceName as 'ContactInvoiceName' /* Type: String */,
ContactInvoiceAddress1 as 'ContactInvoiceAddress1' /* Type: String */,
ContactInvoiceAddress2 as 'ContactInvoiceAddress2' /* Type: String */,
ContactInvoicePostcode as 'ContactInvoicePostcode' /* Type: String */,
ContactInvoiceCity as 'ContactInvoiceCity' /* Type: String */,
ContactInvoicePhone as 'ContactInvoicePhone' /* Type: String */,
ContactInvoiceState as 'ContactInvoiceState' /* Type: String */,
ContactInvoiceCountry as 'ContactInvoiceCountry' /* Type: String */,
ContactDeliveryName as 'ContactDeliveryName' /* Type: String */,
ContactDeliveryAddress1 as 'ContactDeliveryAddress1' /* Type: String */,
ContactDeliveryAddress2 as 'ContactDeliveryAddress2' /* Type: String */,
ContactDeliveryPostcode as 'ContactDeliveryPostcode' /* Type: String */,
ContactDeliveryCity as 'ContactDeliveryCity' /* Type: String */,
ContactDeliveryPhone as 'ContactDeliveryPhone' /* Type: String */,
ContactDeliveryState as 'ContactDeliveryState' /* Type: String */,
ContactDeliveryCountry as 'ContactDeliveryCountry' /* Type: String */
FROM TABLE
WHERE
    CustomerKey = '{0}' /* Type: String, Format: */
```

## CUSTOMERS - SelectCustomer

Cette requête permet de retourner une liste des clients de l'ERP pour la sélection dans l'application Atoo-Sync.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)
- **CustomerName** `String` le nom du client dans l'ERP (exemple: ATOO NEXT)
- **AccountingAccountNumber** `String` le compte comptable du client dans l'ERP (exemple: 411000)

### Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */,
  CustomerName as 'CustomerName' /* Type: String */,
  AccountingAccountNumber as 'AccountingAccountNumber' /* Type: String */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration pour la sélection d'un client.

Sélectionner un client

Entrez le texte à rechercher... Trouver Effacer

Numéro de compte ▲	Nom du client
MANDR	Manderscheid Gmbh
MARTINS	Martins SA
NETTO	Nettoyage et Cie
OPALE	Opale
PANDAN	Pandan Tiffany
PERLE	Perles parisiennes
PLATI	Platine & fils
QUART	La Montre du Quartier
RANDOR	RANDOR Rémi
RUBIS	Rubis sur Longleux
SAPHIRA	Saphira
SECUR	Sécurité services
TATOR	Tat'OR
TELEC	Télécoms
TOPAZ	La Topaze Lyonnaise
WB0000001	DOMAT Michel

## PENDINGPRODUCTS - PendingProductCustomerList

Cette requête permet de retourner la liste des clients avec des articles en attente à exporter dans la boutique.

**Notes** : Les articles en attente sont les articles commandés mais pas encore livrés au client, en général dans les ERP, ils sont dans les bons de commandes des clients (gestion de reliquat).

## Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomerKey** `String` le code unique du client l'ERP (exemple: C0001)

## Exemple de requête

```
SELECT
  CustomerKey as 'CustomerKey' /* Type: String */
FROM TABLE
```

## PENDINGPRODUCTS - ReadCustomerPendingProducts

Cette requête envoie pour un client les articles en attente de livraison dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **OrderKey** **String**, le numéro unique de la commande dans la boutique (exemple: 1245)
- **DocumentNumber** **String**, le numéro du document dans l'ERP (exemple: BC1547)
- **DocumentDate** **Date**, la date du document dans l'ERP (exemple: 2020-09-28)
- **DocumentDeliveryDate** **Date**, la date de livraison du document dans l'ERP (exemple: 2020-10-06)
- **ProductKey** **String** le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** **String** le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)
- **ProductQuantity** **Numeric** la quantité commandée de l'article dans le document (exemple: 12)
- **ProductUnitPriceTaxExclude** **Numeric** le prix unitaire HT de la ligne d'article (exemple: 186.66)
- **ProductUnitPriceTaxInclude** **Numeric** le prix unitaire TTC de la ligne d'article (exemple: 223.99)
- **ProductTotalPriceTaxExclude** **Numeric** le prix total HT de la ligne d'article (exemple: 2 239.90)
- **ProductTotalPriceTaxInclude** **Numeric** le prix total TTC de la ligne d'article (exemple: 2 687.88)
- **ProductTotalTax** **Numeric** le montant total des taxes de la ligne d'article (exemple: 447.98)
- **ProductDiscount** **String** la remise de la ligne d'article (exemple: 10 %)
- **ProductPackaging** **Numeric** le conditionnement de l'article (exemple: 6)
- **ProductUnity** **String** l'unité de vente de l'article (exemple: La caisse)
- **DocumentDeliveryDate** **Date**, la date de livraison de la ligne d'article (exemple: 2020-10-06)

### Argument obligatoire

- **{0}** spécifie le code unique du client (exemple: C0001)

### Exemple de requête

```
SELECT
  OrderKey as 'OrderKey' /* Type: String */,
  DocumentNumber as 'DocumentNumber' /* Type: String */,
  DocumentDate as 'DocumentDate' /* Type: Date */,
  DocumentDeliveryDate as 'DocumentDeliveryDate' /* Type: Date */,
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */,
  ProductQuantity as 'ProductQuantity' /* Type: Numeric */,
  ProductUnitPriceTaxExclude as 'ProductUnitPriceTaxExclude' /* Type: Numeric */,
```

```
ProductUnitPriceTaxInclude as 'ProductUnitPriceTaxInclude' /* Type: Numeric */,
ProductTotalPriceTaxExclude as 'ProductTotalPriceTaxExclude' /* Type: Numeric */,
ProductTotalPriceTaxInclude as 'ProductTotalPriceTaxInclude' /* Type: Numeric */,
ProductTotalTax as 'ProductTotalTax' /* Type: Numeric */,
ProductDiscount as 'ProductDiscount' /* Type: String */,
ProductPackaging as 'ProductPackaging' /* Type: Numeric */,
ProductUnity as 'ProductUnity' /* Type: String */,
ProductDeliveryDate as 'ProductDeliveryDate' /* Type: Date */
FROM TABLE
WHERE
  CustomerKey = '{0}' /* Type: String, Format: */
```

## PRICES - PriceGroupList

Cette requête permet de retourner la liste des groupes de tarifs de l'ERP pour les créer dans la boutique comme groupe de clients.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **PriceGroupKey** `String` le code unique du groupe de clients dans l'ERP (exemple: TARIF1)
- **PriceGroupName** `String` le nom du groupe de clients dans l'ERP (exemple: REVENDEUR)

### Exemple de requête

```
SELECT
  PriceGroupKey as 'PriceGroupKey' /* Type: String */,
  PriceGroupName as 'PriceGroupName' /* Type: String */
FROM TABLE
```

## PRICES - ReadPriceGroup

Cette requête permet de retourner les données d'un groupe de tarifs pour le créer dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **PriceGroupKey** **String** le code unique du groupe de clients dans l'ERP (exemple: TARIF1)
- **PriceGroupName** **String** le nom du groupe de clients dans l'ERP (exemple: REVENDEUR)
- **PriceGroupTaxInclude** **Boolean** indique si le groupe est HT (0) ou TTC (1) (exemple: 1)

### Argument obligatoire

- **{0}** spécifie le code unique du groupe de client (exemple: TARIF1)

### Exemple de requête

```
SELECT
  PriceGroupKey as 'PriceGroupKey' /* Type: String */,
  PriceGroupName as 'PriceGroupName' /* Type: String */,
  PriceGroupTaxInclude as 'PriceGroupTaxInclude' /* Type: Boolean (0 or 1) */
FROM TABLE
WHERE
  PriceGroupKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - CategoryList

Cette requête permet de retourner la liste des catégories des articles pour les créer dans la boutique. Il est possible de spécifier une catégorie parente pour créer une arborescence de catégories dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CategoryKey** `String`, le code unique de la catégorie d'articles dans l'ERP (exemple: FA03)
- **CategoryName** `String`, le nom de la catégorie d'articles dans l'ERP (exemple: Pinceaux)
- **CategoryParentKey** `String`, le code unique de la catégorie parente dans l'ERP (exemple: FA01)

### Exemple de requête

```
SELECT
  CategoryKey as 'CategoryKey' /* Type: String */,
  CategoryName as 'CategoryName' /* Type: String */,
  CategoryParentKey as 'CategoryParentKey' /* Type: String */
FROM TABLE
```

## PRODUCTS - CustomFieldList

Cette requête retourne la liste des champs personnalisables configurés sur les articles pour pouvoir les utiliser dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CustomFieldKey** **String**, le code unique du champ personnalisé de l'article dans l'ERP (exemple: ATTR001)
- **CustomFieldName** **String**, le nom du champ personnalisé de l'article dans l'ERP (exemple: Définition)
- **IsText** **Boolean**, Indique si le champ est de type texte 0 = Non, 1 = Oui (exemple: 1)
- **IsRTF** **Boolean**, Indique si le champ est de type texte enrichi 0 = Non, 1 = Oui (exemple: 1)
- **IsNumeric** **Boolean**, Indique si le champ est de type numérique 0 = Non, 1 = Oui (exemple: 0)

**Notes :** Les champs **IsText**, **IsRTF** et **IsNumeric** servent à connaître le type de champ.

### Exemple de requête

```
SELECT
  'color_field' as 'CustomFieldKey' /* Type: String */,
  'Couleur' as 'CustomFieldName' /* Type: String */,
  1 as 'IsText' /* Type: Boolean (1 or 0) */,
  0 as 'IsRTF' /* Type: Boolean (1 or 0) */,
  0 as 'IsNumeric' /* Type: Boolean (1 or 0) */
FROM TABLE
```

### Utilisation dans la requête ReadProduct

Pour utiliser ces champs dans la requête **ReadProduct**, il faut la compléter avec la valeur du champ et clé du champ comme nom de colonne.

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */,
  ...
  'Rouge' as 'color_field'
FROM TABLE
WHERE
  ProductKey = '{0}' /* Type: String, Format: */
```



## PRODUCTS - Documents

Cette requête retourne la liste des documents pour un article, le nom du fichier doit être le chemin complet du document.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **DocumentFilename** `String`, le nom du fichier dans l'ERP (exemple: documentation.pdf)
- **DocumentName** `String`, le chemin du fichier (exemple: t:\dossier\documentation.pdf)

### Argument obligatoire

- **{0}** spécifie le code unique de l'article (exemple: ART0001)

### Exemple de requête

```
SELECT
  DocumentFilename as 'DocumentFilename' /* Type: String */,
  DocumentName as 'DocumentName' /* Type: String */
FROM TABLE
WHERE
  ProductKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - GetCustomFieldValue

Cette requête permet de lire un champ personnalisé de la fiche article pour l'envoyer dans la boutique.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CustomFieldValue** `String`, la valeur du champ personnalisé de l'article dans l'ERP (exemple: Full HD)

### Arguments obligatoires

- **{0}** spécifie le code de l'article (exemple: ART0001)
- **{1}** spécifie le code du champ personnalisé (exemple: ATTR001)

### Exemple de requête

```
SELECT
  CustomFieldValue as 'CustomFieldValue' /* Type: String */
FROM TABLE
WHERE
  ProductKey = '{0}' /* Type: String, Format: */
AND
  CustomFieldKey = '{1}' /* Type: String, Format: */,
```

## PRODUCTS - OtherProducts

Cette requête retourne la liste les articles pour les utiliser dans les commandes (article divers, article frais de port, article remise ...).

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** String, le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** String, le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)

## Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration des autres article.

Commandes  
Configurer le mode de création des commandes dans la Gestion Commerciale

Création des commandes Articles Règlements

Autres articles

Article Divers :	ZDIVERS	...
Article Emballage :	ZPORT	...
Article Frais de port :	ZPORT	...
Article Remise :	Mappage des articles remises....	✓

## PRODUCTS - ProductExist

Cette requête permet de rechercher si un article existe dans l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** `String`, le code de l'article dans l'ERP (exemple: ART0001)
- **ProductName** `String` le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)

### Argument obligatoire

- `{0}` spécifie le code unique de l'article (exemple: ART0001)

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */
FROM TABLE
WHERE
  ProductKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - ProductList

Cette requête retourne la liste des articles à créer ou à mettre à jour dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** `String`, le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** `String`, le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation pour filtrer les articles

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */
FROM TABLE
WHERE
  ProductModificationDate = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## PRODUCTS - ProductPrices

Cette requête retourne la liste des prix spécifiques d'un article.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** String, le code unique de l'article dans l'ERP (exemple: ART0001)
- **PriceGroupKey** String, le code du groupe de tarifs dans l'ERP (exemple: TARIF1)
- **CustomerKey** String, le code unique du client l'ERP (exemple: C0001)
- **ShopKey** String, le code unique de la boutique (exemple: 1)
- **Price** Numeric, le prix HT de l'article (exemple: 45.78)
- **FromQuantity** Numeric, la quantité à partir de laquelle le prix s'applique (exemple: 4)
- **ReductionAmount** Numeric, le montant de la remise HT (exemple: 7.50)
- **ReductionPercent** Numeric, le montant de la remise en pourcentage (exemple: 5)
- **From** Date, la date de début du prix spécifique (exemple: 2020-09-27)
- **To** Date, la date de fin du prix spécifique (exemple: 2020-10-05)

### Argument obligatoire

- **{0}** spécifie le code unique de l'article (exemple: ART0001)

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  PriceGroupKey as 'PriceGroupKey' /* Type: String */,
  CustomerKey as 'CustomerKey' /* Type: String */,
  ShopKey as 'ShopKey' /* Type: String */,
  Price as 'Price' /* Type: Numeric */,
  FromQuantity as 'FromQuantity' /* Type: Numeric */,
  ReductionAmount as 'ReductionAmount' /* Type: Numeric */,
  ReductionPercent as 'ReductionPercent' /* Type: Numeric */,
  From as 'From' /* Type: Date */,
  To as 'To' /* Type: Date */
FROM TABLE
WHERE
  ProductKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - ProductStates

Cette requête retourne la liste des articles et le statut actif/inactif pour activer ou désactiver la fiche produit dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** `String`, le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** `String`, le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)
- **ProductState** `Boolean`, l'état (1 = actif / 0 = inactif) de l'article dans l'ERP (exemple: 1)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation pour filtrer les articles

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */,
  ProductState as 'ProductState' /* Type: Boolean (1 or 0) */
FROM TABLE
WHERE
  ProductModificationDate = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## PRODUCTS - ProductWarehouses

Cette requête retourne la liste des quantités en stock des articles dans les différents dépôts de l'ERP.

**Note** : Souvent cette information n'est pas gérée nativement dans les CMS ecommerce, il faut pour cela "adapter" le **WebService Atoo-Sync** si vous voulez utiliser cette information dans la boutique.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** **String** le code unique de l'article dans l'ERP (exemple: ART0001)
- **WarehouseKey** **String** le code du dépôt dans l'ERP (exemple: DE01)
- **WarehouseName** **String** le nom du dépôt dans l'ERP (exemple: Principal)
- **Quantity** **Numeric**, la quantité en stock de l'article dans le dépôt de l'ERP (exemple: 99)
- **StockReal** **Numeric**, la quantité en stock réel de l'article dans le dépôt de l'ERP (exemple: 47)
- **StockAvailable** **Numeric**, la quantité en stock disponible de l'article dans le dépôt de l'ERP (exemple: 23)
- **StockTarget** **Numeric**, la quantité en stock à terme de l'article dans le dépôt de l'ERP (exemple: 452)
- **StockRealMinusOrders** **Numeric**, la quantité en stock réel moins les commandes clients de l'article dans le dépôt de l'ERP (exemple: 32)
- **StockTargetMinusPurchaseOrders** **Numeric**, la quantité en stock à terme moins les commandes fournisseurs de l'article dans le dépôt de l'ERP (exemple: 328)
- **Location** **String** l'emplacement de l'article dans le dépôt de l'ERP (exemple: A4T3P2)

## Argument obligatoire

- **{0}** spécifie le code unique de l'article (exemple: ART0001)

## Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  WarehouseKey as 'WarehouseKey' /* Type: String */,
  WarehouseName as 'WarehouseName' /* Type: String */,
  Quantity as 'Quantity' /* Type: Numeric */,
  StockReal as 'StockReal' /* Type: Numeric */,
  StockAvailable as 'StockAvailable' /* Type: Numeric */,
  StockTarget as 'StockTarget' /* Type: Numeric */,
  StockRealMinusOrders as 'StockRealMinusOrders' /* Type: Numeric */,
  StockTargetMinusPurchaseOrders as 'StockTargetMinusPurchaseOrders' /* Type: Numeric
*/,
```

```
Location as 'Location' /* Type: string */  
FROM TABLE  
WHERE  
ProductKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - ReadCategory

Cette requête permet de lire les données d'une catégorie d'articles pour la créer comme catégorie dans la boutique. Pour chaque catégorie retournée par la requête **CategoryList**.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **CategoryKey** `String` le code unique de la catégorie d'articles dans l'ERP (exemple: FA03)
- **CategoryParentKey** `String` le code unique de la catégorie parente de la famille dans l'ERP (exemple: FA01)
- **CategoryName** `String` le nom de la catégorie d'article dans l'ERP (exemple: Pinceaux)
- **CategoryDescription** `String` la description de la catégorie d'articles dans l'ERP

### Argument obligatoire

- `{0}` spécifie le code unique de la famille d'articles (exemple: FA03)

### Exemple de requête

```
SELECT
  CategoryKey as 'CategoryKey' /* Type: String */,
  CategoryParentKey as 'CategoryParentKey' /* Type: String */,
  CategoryName as 'CategoryName' /* Type: String */,
  CategoryDescription as 'CategoryDescription' /* Type: String */
FROM TABLE
WHERE
  CategoryKey = '{0}' /* Type: String, Format: */
```

## PRODUCTS - ReadProduct

Cette requête permet de lire les données d'un article pour le créer ou le modifier dans la boutique. Pour chaque code produit retourné par la requête **PRODUCTS - ProductList** Atoo-Sync crée une requête. C'est cette requête qui permet d'envoyer les données de l'article dans la boutique.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** **String** le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** **String** le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)
- **ProductNameLanguage2** **String** le nom de l'article dans la langue 2
- **ProductNameLanguage3** **String** le nom de l'article dans la langue 3
- **ProductNameLanguage4** **String** le nom de l'article dans la langue 4
- **ProductNameLanguage5** **String** le nom de l'article dans la langue 5
- **ProductDescription** **String** la description de l'article dans l'ERP
- **ProductDescriptionLanguage2** **String** la description de l'article dans la langue 2
- **ProductDescriptionLanguage3** **String** la description de l'article dans la langue 3
- **ProductDescriptionLanguage4** **String** la description de l'article dans la langue 4
- **ProductDescriptionLanguage5** **String** la description de l'article dans la langue 5
- **ProductShortDescription** **String** le résumé de l'article dans l'ERP
- **ProductShortDescriptionLanguage2** **String** le résumé de l'article dans la langue 2
- **ProductShortDescriptionLanguage3** **String** le résumé de l'article dans la langue 3
- **ProductShortDescriptionLanguage4** **String** le résumé de l'article dans la langue 4
- **ProductShortDescriptionLanguage5** **String** le résumé de l'article dans la langue 5
- **ProductMetaTitle** **String** la balise titre de l'article dans l'ERP
- **ProductMetaTitleLanguage2** **String** la balise titre de l'article dans la langue 2
- **ProductMetaTitleLanguage3** **String** la balise titre de l'article dans la langue 3
- **ProductMetaTitleLanguage4** **String** la balise titre de l'article dans la langue 4
- **ProductMetaTitleLanguage5** **String** la balise titre de l'article dans la langue 5
- **ProductMetaDescription** **String** la description de l'article pour les moteurs de recherche dans l'ERP
- **ProductMetaDescriptionLanguage2** **String** la méta-description dans la langue 2
- **ProductMetaDescriptionLanguage3** **String** la méta-description dans la langue 3
- **ProductMetaDescriptionLanguage4** **String** la méta-description dans la langue 4
- **ProductMetaDescriptionLanguage5** **String** la méta-description dans la langue 5
- **ProductKeyWords** **String** les mots clés de l'article dans l'ERP
- **ProductKeyWordsLanguage2** **String** les mots clés de l'article dans la langue 2
- **ProductKeyWordsLanguage3** **String** les mots clés de l'article dans la langue 3
- **ProductKeyWordsLanguage4** **String** les mots clés de l'article dans la langue 4
- **ProductKeyWordsLanguage5** **String** les mots clés de l'article dans la langue 5
- **ProductLinkRewrite** **String** l'url simplifiée de l'article dans l'ERP
- **ProductLinkRewriteLanguage2** **String** l'url simplifiée de l'article dans la langue 2
- **ProductLinkRewriteLanguage3** **String** l'url simplifiée de l'article dans la langue 3
- **ProductLinkRewriteLanguage4** **String** l'url simplifiée de l'article dans la langue 4

- **ProductLinkRewriteLanguage5** **String** l'url simplifiée de l'article dans la langue 5
- **TaxKey** **String** le code de la taxe de l'article (exemple: C20)
- **RegularPriceTaxExclude** **Numeric** prix de vente régulier en HT de l'article (exemple: 165.83)
- **RegularPriceTaxInclude** **Numeric** prix de vente régulier en TTC de l'article (exemple: 199.00)
- **ProductPriceTaxExclude** **Numeric** prix de vente HT de l'article (exemple: 124.56)
- **ProductPriceTaxInclude** **Numeric** prix de vente TTC de l'article (exemple: 149.47)
- **ProductTaxRate** **Numeric** Le taux de TVA de l'article (exemple: 20)
- **ProductEcotax** **Numeric** montant de l'écotaxe HT de l'article (exemple: 0.88)
- **ProductWholesalePrice** **Numeric** prix d'achat HT de l'article (exemple: 87.12)
- **ProductSaleUnity** **String** l'unité de vente de l'article (exemple: Par pièce)
- **ProductSupplierReference** **String** la référence fournisseur de l'article (exemple: RT0121)
- **ProductWidth** **Numeric** la largeur de l'article (exemple: 8.50)
- **ProductHeight** **Numeric** la hauteur de l'article (exemple: 24.70)
- **ProductDepth** **Numeric** la profondeur de l'article (exemple: 12.60)
- **ProductWeight** **Numeric** le poids de l'article (exemple: 1.250)
- **ProductEAN** **String** le code barre EAN de l'article (exemple: 1247869534712)
- **ProductUPC** **String** le code UPC de l'article (exemple: 72527273070)
- **ProductSupplierName** **String** le nom du fournisseur de l'article (exemple: RABONY)
- **ProductManufacturerName** **String** la marque de l'article (exemple: APPLE)
- **FamilyKey** **String**, le code de la famille de l'article (exemple: FA03)
- **SubFamilyKey** **String**, le code de la sous-famille de l'article (exemple: SF01)
- **Quantity** **Numeric** la quantité en stock de l'article dans l'ERP (exemple: 99)
- **StockReal** **Numeric** la quantité en stock réel de l'article dans l'ERP (exemple: 47)
- **StockAvailable** **Numeric** la quantité en stock disponible de l'article dans l'ERP (exemple: 23)
- **StockTarget** **Numeric** la quantité en stock à terme de l'article dans l'ERP (exemple: 452)
- **StockRealMinusOrders** **Numeric** la quantité en stock réel moins les commandes clients de l'article dans l'ERP (exemple: 32)
- **StockTargetMinusPurchaseOrders** **Numeric** la quantité en stock à terme moins les commandes fournisseurs de l'article dans l'ERP (exemple: 328)
- **Location** **String** l'emplacement de l'article dans le dépôt (exemple: A4T3P2)
- **ProductNextDeliveryDate** **Date** la date de prochaine livraison de l'article dans l'ERP (exemple: 2020-10-23)
- **ProductNextDeliveryQuantity** **Numeric** la prochaine quantité livrée de l'article dans l'ERP (exemple: 965)
- **ProductVariationReference** **String** le code de regroupement des articles dans le CMS (exemple: PANTALON)
- **ProductVariation1Name** **String** le nom de la variation 1 dans le CMS (exemple: Couleur)
- **ProductVariation1Value** **String** la valeur de la variation 1 dans le CMS (exemple: Jaune)
- **ProductVariation2Name** **String**, le nom de la variation 2 dans le CMS (exemple: Taille)
- **ProductVariation2Value** **String**, la valeur de la variation 2 dans le CMS (exemple: L)
- **ProductVariation3Name** **String**, le nom de la variation 3 dans le CMS (exemple: Matière)
- **ProductVariation3Value** **String**, la valeur de la variation 3 dans le CMS (exemple: Cuir)
- **ProductVariation4Name** **String**, le nom de la variation 4 dans le CMS (exemple: Pointure)
- **ProductVariation4Value** **String**, la valeur de la variation 4 dans le CMS (exemple: 42)
- **ProductVariation5Name** **String**, le nom de la variation 5 dans le CMS (exemple: Diamètre)
- **ProductVariation5Value** **String**, la valeur de la variation 5 dans le CMS (exemple: 20 cm)
- **ProductCategory1** **String**, la premier niveau de la catégorie de l'article dans le CMS (exemple: Outils)

- **ProductCategory2** **String**, la second niveau de la catégorie de l'article dans le CMS (exemple: Peintre)
- **ProductCategory3** **String**, la troisième niveau de la catégorie de l'article dans le CMS (exemple: Brosses)
- **ProductCategory4** **String**, la quatrième niveau de la catégorie de l'article dans le CMS (exemple: Plate)
- **ProductCategory5** **String**, la cinquième de la catégorie de l'article dans le CMS (exemple: Nylon)
- **ProductCategory6** **String**, la dernière niveau de la catégorie de l'article dans le CMS (exemple: Souple)
- **AdditionalCategories** **String**, les catégories supplémentaires de l'article dans le CMS (exemple: Mur, Plafond, Extérieure)
- **ProductPhoto** **String**, le chemin de la photo principale de l'article dans l'ERP (exemple: T:\dossier\ART0001.jpg)
- **ProductAdditionalPhoto1** **String**, le chemin de la photo secondaire 1 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_1.jpg)
- **ProductAdditionalPhoto2** **String**, le chemin de la photo secondaire 2 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_2.jpg)
- **ProductAdditionalPhoto3** **String**, le chemin de la photo secondaire 3 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_3.jpg)
- **ProductAdditionalPhoto4** **String**, le chemin de la photo secondaire 4 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_4.jpg)
- **ProductAdditionalPhoto5** **String**, le chemin de la photo secondaire 5 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_5.jpg)
- **ProductAdditionalPhoto6** **String**, le chemin de la photo secondaire 6 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_6.jpg)
- **ProductAdditionalPhoto7** **String**, le chemin de la photo secondaire 7 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_7.jpg)
- **ProductAdditionalPhoto8** **String**, le chemin de la photo secondaire 8 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_8.jpg)
- **ProductAdditionalPhoto9** **String**, le chemin de la photo secondaire 9 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_9.jpg)
- **ProductAdditionalPhoto10** **String**, le chemin de la photo secondaire 10 de l'article dans l'ERP (exemple: T:\dossier\ART0001\_10.jpg)

**Note:** Cette requête est la plus complète, elle permet de renseigner les informations de texte de l'article, le prix, le stock ainsi que les catégories de l'article si renseignées. Si certains champs ne sont pas utilisés, il faut les remplacer par des valeurs vides.

## Argument obligatoire

- {0} spécifie le code unique de l'article (exemple: ART0001)

## Exemple de requête

```
SELECT
```

```
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */,
  ProductNameLanguage2 as 'ProductNameLanguage2' /* Type: String */,
  ProductNameLanguage3 as 'ProductNameLanguage3' /* Type: String */,
  ProductNameLanguage4 as 'ProductNameLanguage4' /* Type: String */,
  ProductNameLanguage5 as 'ProductNameLanguage5' /* Type: String */,
  ProductDescription as 'ProductDescription' /* Type: String */,
  ProductDescriptionLanguage2 as 'ProductDescriptionLanguage2' /* Type: String */,
  ProductDescriptionLanguage3 as 'ProductDescriptionLanguage3' /* Type: String */,
  ProductDescriptionLanguage4 as 'ProductDescriptionLanguage4' /* Type: String */,
  ProductDescriptionLanguage5 as 'ProductDescriptionLanguage5' /* Type: String */,
  ProductShortDescription as 'ProductShortDescription' /* Type: String */,
  ProductShortDescriptionLanguage2 as 'ProductShortDescriptionLanguage2' /* Type: String
*/,
  ProductShortDescriptionLanguage3 as 'ProductShortDescriptionLanguage3' /* Type: String
*/,
  ProductShortDescriptionLanguage4 as 'ProductShortDescriptionLanguage4' /* Type: String
*/,
  ProductShortDescriptionLanguage5 as 'ProductShortDescriptionLanguage5' /* Type: String
*/,
  ProductMetaTitle as 'ProductMetaTitle' /* Type: String */,
  ProductMetaTitleLanguage2 as 'ProductMetaTitleLanguage2' /* Type: String */,
  ProductMetaTitleLanguage3 as 'ProductMetaTitleLanguage3' /* Type: String */,
  ProductMetaTitleLanguage4 as 'ProductMetaTitleLanguage4' /* Type: String */,
  ProductMetaTitleLanguage5 as 'ProductMetaTitleLanguage5' /* Type: String */,
  ProductMetaDescription as 'ProductMetaDescription' /* Type: String */,
  ProductMetaDescriptionLanguage2 as 'ProductMetaDescriptionLanguage2' /* Type: String
*/,
  ProductMetaDescriptionLanguage3 as 'ProductMetaDescriptionLanguage3' /* Type: String
*/,
  ProductMetaDescriptionLanguage4 as 'ProductMetaDescriptionLanguage4' /* Type: String
*/,
  ProductMetaDescriptionLanguage5 as 'ProductMetaDescriptionLanguage5' /* Type: String
*/,
  ProductKeyWords as 'ProductKeyWords' /* Type: String */,
  ProductKeyWordsLanguage2 as 'ProductKeyWordsLanguage2' /* Type: String */,
  ProductKeyWordsLanguage3 as 'ProductKeyWordsLanguage3' /* Type: String */,
  ProductKeyWordsLanguage4 as 'ProductKeyWordsLanguage4' /* Type: String */,
  ProductKeyWordsLanguage5 as 'ProductKeyWordsLanguage5' /* Type: String */,
  ProductLinkRewrite as 'ProductLinkRewrite' /* Type: String */,
  ProductLinkRewriteLanguage2 as 'ProductLinkRewriteLanguage2' /* Type: String */,
  ProductLinkRewriteLanguage3 as 'ProductLinkRewriteLanguage3' /* Type: String */,
  ProductLinkRewriteLanguage4 as 'ProductLinkRewriteLanguage4' /* Type: String */,
  ProductLinkRewriteLanguage5 as 'ProductLinkRewriteLanguage5' /* Type: String */,
  TaxeKey as 'TaxeKey' /* Type: String */,
  RegularPriceTaxExclude as 'RegularPriceTaxExclude' /* Type: Numeric */,
  RegularPriceTaxInclude as 'RegularPriceTaxInclude' /* Type: Numeric */,
```

```
ProductPriceTaxExclude as 'ProductPriceTaxExclude' /* Type: Numeric */,
ProductPriceTaxInclude as 'ProductPriceTaxInclude' /* Type: Numeric */,
ProductTaxRate as 'ProductTaxRate' /* Type: Numeric */,
ProductEcotax as 'ProductEcotax' /* Type: Numeric */,
ProductWholesalePrice as 'ProductWholesalePrice' /* Type: Numeric */,
ProductSaleUnity as 'ProductSaleUnity' /* Type: String */,
ProductSupplierReference as 'ProductSupplierReference' /* Type: String */,
ProductWidth as 'ProductWidth' /* Type: Numeric */,
ProductHeight as 'ProductHeight' /* Type: Numeric */,
ProductDepth as 'ProductDepth' /* Type: Numeric */,
ProductWeight as 'ProductWeight' /* Type: Numeric */,
ProductEAN as 'ProductEAN' /* Type: String */,
ProductUPC as 'ProductUPC' /* Type: String */,
ProductSupplierName as 'ProductSupplierName' /* Type: String */,
ProductManufacturerName as 'ProductManufacturerName' /* Type: String */,
FamilyKey as 'FamilyKey' /* Type: String */,
SubFamilyKey as 'SubFamilyKey' /* Type: String */,
Quantity as 'Quantity' /* Type: Numeric */,
StockReal as 'StockReal' /* Type: Numeric */,
StockAvailable as 'StockAvailable' /* Type: Numeric */,
StockTarget as 'StockTarget' /* Type: Numeric */,
StockRealMinusOrders as 'StockRealMinusOrders' /* Type: Numeric */,
StockTargetMinusPurchaseOrders as 'StockTargetMinusPurchaseOrders' /* Type: Numeric
*/,
Location as 'Location' /* Type: string */,
ProductNextDeliveryDate as 'ProductNextDeliveryDate' /* Type: Date */,
ProductNextDeliveryQuantity as 'ProductNextDeliveryQuantity' /* Type: Numeric */,
ProductVariationReference as 'ProductVariationReference' /* Type: String */,
ProductVariation1Name as 'ProductVariation1Name' /* Type: String */,
ProductVariation1Value as 'ProductVariation1Value' /* Type: String */,
ProductVariation2Name as 'ProductVariation2Name' /* Type: String */,
ProductVariation2Value as 'ProductVariation2Value' /* Type: String */,
ProductVariation3Name as 'ProductVariation3Name' /* Type: String */,
ProductVariation3Value as 'ProductVariation3Value' /* Type: String */,
ProductVariation4Name as 'ProductVariation4Name' /* Type: String */,
ProductVariation4Value as 'ProductVariation4Value' /* Type: String */,
ProductVariation5Name as 'ProductVariation5Name' /* Type: String */,
ProductVariation5Value as 'ProductVariation5Value' /* Type: String */,
ProductCategory1 as 'ProductCategory1' /* Type: String */,
ProductCategory2 as 'ProductCategory2' /* Type: String */,
ProductCategory3 as 'ProductCategory3' /* Type: String */,
ProductCategory4 as 'ProductCategory4' /* Type: String */,
ProductCategory5 as 'ProductCategory5' /* Type: String */,
ProductCategory6 as 'ProductCategory6' /* Type: String */,
AdditionnalCategories as 'AdditionnalCategories' /* Type: String */,
ProductPhoto as 'ProductPhoto' /* Type: String */
FROM TABLE
WHERE
ProductKey = '{0}' /* Type: String, Format: */
```



## PRODUCTS - UpdateProductStockList

Cette requête retourne la liste des articles pour la mise à jour du stock dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** `String`, le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** `String`, le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation de stock pour filtrer les articles

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */
FROM TABLE
WHERE
  FilterDateModification = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## PRODUCTS - UpdateProductPriceList

Cette requête retourne la liste des articles pour la mise à jour du prix dans la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ProductKey** `String`, le code unique de l'article dans l'ERP (exemple: ART0001)
- **ProductName** `String`, le nom de l'article dans l'ERP (exemple: Sac besace cuir femme)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation des prix pour filtrer les articles

### Exemple de requête

```
SELECT
  ProductKey as 'ProductKey' /* Type: String */,
  ProductName as 'ProductName' /* Type: String */
FROM TABLE
WHERE
  FilterDateModification = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## SALES - AccountingJournals

Cette requête retourne la liste des journaux comptables de l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **JournalKey** String, le code unique du journal comptable de l'ERP (exemple: BQ)
- **JournalName** String, le nom du journal comptable de l'ERP (exemple: Banque)

### Exemple de requête

```
SELECT
  JournalKey as 'JournalKey' /* Type: String */,
  JournalName as 'JournalName' /* Type: String */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration des mappages des règlements/paiements entre la boutique et l'ERP, pour indiquer le mode de règlement à renseigner sur le document de vente.

**Atoo Sync** Mappage des Règlements/Paiements

Règlement par défaut :

Journal par défaut :

Déposer la colonne de votre choix ICI pour grouper par celle-ci.

Site web	Devise	Mode de Règlement	Journal	Co
<input type="checkbox"/>	<input type="checkbox"/>	=	=	<input type="checkbox"/>
Chèque	Euro	Chèque	Banque Européenne...	
opartdevis	Euro	Chèque	Caisse	
ps_checkout	Euro	Chèque	Caisse	
ps_checkpayment	Euro	Chèque	Banque Rivas & Duras	
ps_wirepayment	Euro	Chèque	Banque Rivas & D...	

## SALES - Currencies

Cette requête retourne la liste des devises configurées dans l'ERP.

### Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **CurrencyName** `String`, le nom unique de la devise dans l'ERP (exemple: Dollar Américain)

### Exemple de requête

```
SELECT
  CurrencyName as 'CurrencyName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration des devises entre la boutique et l'ERP pour que le logiciel Atoo-Sync sache si la commande de la boutique est en devise et si oui quelle devise renseigner sur le document de vente.

The screenshot shows a dialog box titled "Mappage des devises" with a close button (X) in the top right corner. Below the title bar, there is a label "Devise de la boutique :" followed by a dropdown menu currently displaying "Euro". Underneath is a table with two columns: "Boutique" and "Gestion Commerciale". The table body is currently empty. At the bottom of the dialog, there are two buttons: "Appliquer" and "Annuler".

## SALES - ReadOrder

Cette requête permet de lire les informations du document dans l'ERP correspondant à la commande de la boutique. Elle sert à envoyer le numéro de suivi du transporteur et à modifier le statut de la commande dans la boutique.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **DocumentNumber** `String`, le numéro de document dans l'ERP (exemple: BC14571)
- **DocumentType** `Integer`, le type de document dans l'ERP (exemple: 4)
  - Valeurs attendues pour **DocumentType**
  - 1 = Devis dans l'ERP
  - 2 = Bon de commande dans l'ERP
  - 3 = Préparation de livraison dans l'ERP
  - 4 = Bon de livraison dans l'ERP
  - 5 = Facture dans l'ERP
  - 6 = Facture validée ou comptabilisée dans l'ERP
- **DeliveryDate** `Date`, la date de livraison du document dans l'ERP (exemple 2021-02-14).
- **TrackingNumber** `String`, le numéro de suivi de la commande dans l'ERP (exemple: 6C4551247).
- **ShippingKey** `String`, le code du transporteur de la commande dans l'ERP (exemple: 01).

## Arguments obligatoires

- `{0}` spécifie le code client dans l'ERP (exemple: C0001)
- `{1}` spécifie le code unique de la commande de la boutique (exemple: 1245)

## Exemple de requête

```
SELECT
  DocumentNumber as 'DocumentNumber' /* Type: String */,
  DocumentType as 'DocumentType' /* Type: Integer (1=Quote, 2=Order,
3=PreparationDelivery, 4=Delivery, 5=Invoice, 6=LockedInvoice) */,
  DeliveryDate as 'DeliveryDate' /* Type: String */,
  TrackingNumber as 'TrackingNumber' /* Type: String */,
  ShippingKey as 'ShippingKey ' /* Type: String */
FROM TABLE
WHERE
  CustomerKey = '{0}' /* Type: String, Format: */,
  OrderKey = '{1}' /* Type: String, Format: */
```

## Important

Le **ShippingKey** doit être le même que celui retourné dans la requête **SALES - Shippings**.

## SALES - InvoicePDFList

Cette requête permet de lister les factures PDF à envoyer dans la boutique. La requête peut retourner plusieurs lignes pour chaque commande issue de la boutique.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **OrderKey** `String`, le numéro unique de la commande dans la boutique (exemple: 1245)
- **CustomerKey** `String`, le code unique du client l'ERP (exemple: C0001)
- **DocumentNumber** `String`, le numéro de document dans l'ERP (exemple: FA20004)
- **DocumentReference** `String`, la référence du document dans l'ERP (exemple: WEB 1245)
- **DocumentName** `String`, le nom du fichier PDF du document dans l'ERP (exemple: facture FA20004.pdf)
- **PDFFilename** `String`, le chemin complet du fichier pdf (exemple: t:\pdf\facture FA20004.pdf)
- **DocumentDate** `Date`, la date de la facture dans l'ERP (exemple: 2020-09-28)
- **DocumentTotalTaxExclude** `Numeric`, le montant total HT de la facture dans l'ERP (exemple: 1328.45)
- **DocumentTotalTaxInclude** `Numeric`, le montant total TTC de la facture dans l'ERP (exemple: 1594.14)

### Argument obligatoire

- `{0}` spécifie la date de dernière synchronisation pour filtrer les documents déjà envoyés.

### Exemple de requête

```
SELECT
  OrderKey as 'OrderKey' /* Type: String */,
  CustomerKey as 'CustomerKey' /* Type: String */,
  DocumentNumber as 'DocumentNumber' /* Type: String */,
  DocumentReference as 'DocumentReference' /* Type: String */,
  DocumentName as 'DocumentName' /* Type: String */,
  PDFFilename as 'PDFFilename' /* Type: String (Full path) */,
  DocumentDate as 'DocumentDate' /* Type: Date */,
  DocumentTotalTaxExclude as 'DocumentTotalTaxExclude' /* Type: Numeric */,
  DocumentTotalTaxInclude as 'DocumentTotalTaxInclude' /* Type: Numeric */
FROM TABLE
WHERE
  FilterDateModification = '{0}' /* Type: Date, Format: yyyy-MM-ddTHH:mm:ss */
```

## SALES - SalesNumbers

Cette requête retourne la liste des souche/séries des documents de vente configurées dans l'ERP.

## Champ obligatoire

La requête doit retourner obligatoirement le champ suivant :

- **SaleNumber** String, le nom unique de la souche/série de document l'ERP (exemple: Vente Web)

## Exemple de requête

```
SELECT
  SaleNumber as 'SaleNumber' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration de la création des commandes pour que le logiciel Atoo-Sync sache quelle souche/série de document renseigner sur le document de vente.



The screenshot displays the 'Commandes' configuration interface. On the left is a navigation menu with icons for Profil, Site Web, Gestion Commerciale, Articles, Clients, and Commandes. The main area is titled 'Commandes' and includes the subtitle 'Configurer le mode de création des commandes dans la Gestion Commerciale'. Below this, there are tabs for 'Création des commandes', 'Articles', and 'Règlements'. The 'Création de la commande' section contains several fields: 'Date de la commande' (Date de la commande), 'Précision des prix' (2 chiffres), 'Souche de vente' (WEBGLO, highlighted with a red box), 'Dépôt à utiliser' (Bijou SA), and 'Référence de la commande' (Web %OI - %CF %CL). An example reference is shown as 'Exemple : Web 187 - John DOE'.

## SALES - Settlements

Cette requête retourne la liste des modes de règlements de l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **SettlementKey** String, le code unique du mode de règlement dans l'ERP (exemple: CB)
- **SettlementName** String, le nom du mode de règlement dans l'ERP (exemple: Carte Bancaire)

### Exemple de requête

```
SELECT
  SettlementKey as 'SettlementKey' /* Type: String */,
  SettlementName as 'SettlementName' /* Type: String */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration des mappages des règlements/paiements entre la boutique et l'ERP, pour identifier le mode de règlement à renseigner sur le document de vente.

Mappage des Règlements/Paiements

Règlement par défaut :

Journal par défaut :

Déposer la colonne de votre choix ICI pour grouper par celle-ci.

Site web ▲	Devise	Mode de Règlement	Journal	Co
<input type="text" value="nbc"/>	<input type="text" value="nbc"/>	=	=	<input type="text" value="nbc"/>
Chèque	Euro	Chèque	Banque Européenne...	
opartdevis	Euro	Chèque	Caisse	
ps_checkout	Euro	Chèque	Caisse	
ps_checkpayment	Euro	Chèque	Banque Rivas & Duras	
ps_wirepayment	Euro	Chèque	Banque Rivas & D...	▼

## SALES - Shippings

Cette requête retourne la liste des modes d'expédition configurés dans l'ERP.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **ShippingKey** String, le code unique du mode d'expédition dans l'ERP (exemple: 01)
- **ShippingName** String, le nom du mode d'expédition dans l'ERP (exemple: DPD Point Relais)

### Exemple de requête

```
SELECT
  ShippingKey as 'ShippingKey' /* Type: String */,
  ShippingName as 'ShippingName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration des mappages des frais de port entre la boutique et l'ERP, afin d'indiquer au logiciel Atoo-Sync quel transporteur de l'ERP renseigner sur le document de vente.

Mappage des frais de port

Transport par défaut :

Trouver par :  Son Nom  Son ID

Transporteur	Gestion Commerciale
PrestaShop 1.7.6.7	DPD Relais
My carrier	DPD Relais

## SALES - Warehouses

Cette requête retourne la liste des dépôts de l'ERP pour savoir sur quel dépôt créer la commande dans l'ERP.

## Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **WarehouseKey** **String** le code du dépôt dans l'ERP (exemple: DE01)
- **WarehouseName** **String** le nom du dépôt dans l'ERP (exemple: Principal)

## Exemple de requête

```
SELECT
  WarehouseKey as 'WarehouseKey' /* Type: String */,
  WarehouseName as 'WarehouseName' /* Type: String */
FROM TABLE
```

## Utilisation

Cette requête est utilisée dans la configuration de la création des commandes dans l'ERP.



**Commandes**  
Configurer le mode de création des commandes dans la Gestion Commerciale

Création des commandes | Articles | Règlements

Création de la commande

Date de la commande :

Précision des prix :

Souche de vente :

Dépôt à utiliser :

Référence de la commande :  Exemple : Web 187 - John DOE

Mappage des Commandes par Zone/Pays... ✓

Mappage des Taxes des Commandes... ✓

Mappage des Modes de Paiement... ✓

## TAXES - AccountingCategories

Cette requête retourne la liste des catégories comptables pour le mappage avec la boutique. Elle sert à définir dans la commande la catégorie comptable de vente pour l'application de la TVA.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants:

- **AccountingCategoryKey** `String`, le code unique de la catégorie comptable dans l'ERP (exemple VF)
- **AccountingCategoryName** `String`, le nom de la catégorie comptable (exemple Vente France)

### Exemple de requête

```
SELECT
  AccountingCategoryKey as 'AccountingCategoryKey' /* Type: String */,
  AccountingCategoryName as 'AccountingCategoryName' /* Type: String */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration du mappage de la catégorie comptable du document selon le pays ou la zone de vente.

Mappage des Commandes par Zone / Pays

Configurer sur :

Par défaut avec taxe :

Par défaut sans taxe :

Pays de la boutique	Ventes avec taxe	Ventes sans taxe
France	<input type="text" value="Ventes France"/>	<input type="text" value="Ventes France"/>

## TAXES - Taxes

Cette requête retourne une liste de taxes pour le mappage avec la boutique. Elle sert à définir dans la boutique la taxe de l'article.

### Champs obligatoires

La requête doit retourner obligatoirement les champs suivants :

- **TaxKey** `String`, le code unique de la taxe dans l'ERP (exemple C20).
- **TaxName** `String`, le nom de la taxe (exemple TVA Collectée 20%).
- **TaxRate** `Numeric`, Le taux de taxe de la taxe (exemple 20.00)

### Exemple de requête

```
SELECT
  TaxeKey as 'TaxeKey' /* Type: String */,
  TaxName as 'TaxName' /* Type: String */,
  TaxRate as 'TaxRate' /* Type: Numeric */
FROM TABLE
```

### Utilisation

Cette requête est utilisée dans la configuration des taxes des articles pour que le logiciel Atoo-Sync sache quelle taxe associer à l'article dans la boutique.

**Mappage des taxes des articles**

TVA par défaut :

Code de TVA	Taux de taxe	Site web
Crédit de TVA à reporter	100,00 %	No VAT
Taxes sur la publicité	1,00 %	No VAT
Tva collectée à 10 %	10,00 %	FR Taux réduit (10%)
Tva collectée à 20 %	20,00 %	FR Taux standard (20%)
TVA collectée à 5,5 %	5,50 %	FR Taux réduit (5.5%)
TVA collectée lié à taxe paraf...	20,00 %	No VAT
Tva déd. sur immo à 20%	20,00 %	No VAT
Tva déductible à 10 %	10,00 %	No VAT
Tva déductible à 20 %	20,00 %	No VAT
TVA déductible à 5,5 %	5,50 %	No VAT
TVA export à 0% (pour mém...	0,00 %	No VAT
TVA Import	0,00 %	No VAT
TVA sur achats intracom. à 2...	20,00 %	No VAT

Elle est utilisée également dans la configuration des taxes lors de l'import des commandes dans l'ERP, pour que le logiciel Atoo-Sync sache quelle TVA associer aux lignes du document de vente dans l'ERP.

**Mappage des taxes des commandes**

TVA par défaut :

Taux de taxe	Taxe	Gestion Commerciale
2,10 %	TVA FR 2.1%	TVA export à 0% (pour mémoire)
5,50 %	TVA FR 5.5%	TVA collectée à 5,5 %
10,00 %	TVA FR 10%	Tva collectée à 10 %
17,00 %	TVA LU 17%	TVA export à 0% (pour mémoire)
18,00 %	VAT MT 18%	TVA export à 0% (pour mémoire)
19,00 %	MwSt. DE 19%	TVA export à 0% (pour mémoire)
19,00 %	ΦΠΑ CY 19%	TVA export à 0% (pour mémoire)
20,00 %	DPH SK 20%	TVA export à 0% (pour mémoire)
20,00 %	km EE 20%	TVA export à 0% (pour mémoire)
20,00 %	TVA FR 20%	Tva collectée à 20 %
20,00 %	TVA RO 20%	TVA export à 0% (pour mémoire)
20,00 %	USt. AT 20%	TVA export à 0% (pour mémoire)
20,00 %	VAT UK 20%	TVA export à 0% (pour mémoire)
20,00 %	ДДС BG 20%	TVA export à 0% (pour mémoire)
21,00 %	BTW NL 21%	TVA export à 0% (pour mémoire)



## Configurer Atoo-Sync GesCom dans Shopify

### Créer le champ de regroupement dans Shopify

Pour pouvoir faire des regroupements de produits dans **Shopify**, il faut créer un **champ méta produit** permettant d'enregistrer la **référence de regroupement** (*référence maître dans Atoo-Sync*).

Pour créer ce champ il faut procéder de la manière suivante:

### Afficher les paramètres Shopify

Une fois connecté au back-office de votre boutique Shopify, cliquez sur le bouton **Paramètres** en bas à gauche.



La page **Informations sur la boutique** s'affiche.

**Informations sur la boutique**

**Profil** Modifier

Nom de la boutique: **Atoo**      Téléphone: **02 20 20 20 20**

Adresse personnelle de contact (entre Shopify et vous): **123 rue de la Paix**      E-mail de l'expéditeur: **contact@atoo.com**

Les e-mails que vous envoyez peuvent s'afficher différemment selon les boîtes de réception de vos clients. [Afficher les paramètres des e-mails](#)

**Informations de facturation** Modifier

Raison sociale: **Atoo SAS**      Adresse professionnelle: **123 rue de la Paix, 75001 Paris**

**Devise de la boutique**

Devise utilisée pour vendre vos produits. Pour changer votre devise de versement, aller à [paramètres de paiement](#).

euro (EUR) Modifier le formatage

Pour changer la devise de votre boutique, [contacter l'assistance Shopify](#).

**Fuseau horaire et unités de mesure**


Utilisés pour calculer les prix des produits, les poids d'expédition et les horaires des commandes.

Cliquez sur le lien **Données personnalisées** dans la barre de menu disponible sur la gauche:

- Informations sur la boutique
- Forfait
- Facturation
- Marque
- Notifications
- Données personnalisées**
- Langues
- Politiques

La page **Données personnalisées** s'affiche.

## Données personnalisées

Champs méta 

Ajoutez un élément de donnée personnalisé à une partie spécifique de votre boutique.

Nom	Définitions
 Produits	3
 Variantes	1
 Collections	0

## Créer le champ méta produit

Cliquez sur l'élément **Produits** dans la liste des **champs méta**.

La page **Définitions de champs méta produit** s'affiche

← Définitions de champs méta Produit Ajouter une définition

---

Définitions Champs méta sans définition

Nom de la définition	Type de contenu	Utilisé dans
ÉPINGLE (3 SUR 20)		
 <b>montant_ecotaxe</b> product.metafields.test.montant_ecotaxe	<b>A</b> Une seule ligne de texte	1 produit 
 <b>montant_ecotaxe</b> product.metafields.my_fields.montant_ecotaxe	<b>A</b> Une seule ligne de texte	0 produit 
 <b>product</b> product.metafields.my_fields.product	<b>A</b> Une seule ligne de texte	0 produit 

Cliquez sur le bouton **Ajouter une définition**:



La page **Ajouter une définition de champs méta produit** s'affiche:

Entrez les valeurs suivantes dans les champs disponibles:

Champ	Valeur
Nom	Référence maître
Espace de nom	custom.reference_maitre
Description	Référence Maître pour Atoo-Sync
Type de champ	Une seule ligne de texte
Options d'accès	Ne pas renseigner
Validation	Ne pas renseigner

La configuration du champ du doit ressembler à cet écran.

← Ajouter une définition de champs méta produit Définition épinglée

Nom

Espace de nom et clé

Le système utilise cela pour identifier la définition.

Description

 31/100

A Une seule ligne de texte Changer

Une valeur  Liste de valeurs

Options d'accès ⓘ

Donner un accès en lecture aux options sélectionnées.

Boutiques en ligne

Collections automatisées

Validation

Les valeurs peuvent contenir des lettres, des chiffres et des caractères spéciaux.

Règles

Limiter aux choix prédéfinis

Longueur minimum

Longueur maximum

Expression régulière

Cliquez sur le bouton **Enregistrer** pour enregistrer le nouveau **champ méta produit** dans votre boutique **Shopify**.

Le champ est ajouté dans la liste.

← Définitions de champs méta Produit
Ajouter une définition

Nom de la définition	Type de contenu	Utilisé dans
ÉPINGLÉ (4 SUR 20)		
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">⋮</div> <div> <p><b>reference maitre</b></p> <p>product.metafields.custom.reference_maitre</p> </div> </div>	<p><b>A</b> Une seule ligne de texte</p>	<p>0 produit</p> <div style="text-align: right; color: #e67e22;">📌</div>
<div style="display: flex; align-items: center;"> <div style="margin-right: 10px;">⋮</div> <div> <p><b>montant ecotaxe</b></p> </div> </div>	-	-

**Félicitations !** Vous pouvez maintenant regrouper des articles dans **Shopify**.

## Onglet Site Web : Option Shopify


### Options Shopify dans Site Web :



Dans **Configurer le profil > Site Web** :

Cliquez sur **Options Shopify...**

Options



### Options

#### Produits

- Mettre à jour le nom de l'article
- Mettre à jour la description de l'article
- Mettre à jour les prix de l'article
- Utiliser la configuration de Shopify pour l'envoi des prix HT ou TTC
- Mettre à jour le stock de l'article
- Mettre à jour le poids de l'article
- Mettre à jour le code barre de l'article
- Mettre à jour le fabricant de l'article
- Mettre a jour les caractéristiques de l'article
- Associer les images au produit parent
- profile.cms.options.readProductsSleep

#### Clients

- Remplacer les tags des clients lors de la mise à jour
- Remplacer les adresses des clients lors de la mise à jour

#### Mise à jour des commandes

- Notifier le client

#### Réinitialiser les commandes transférées

ID des commandes  [Réinitialiser les commandes transférées](#)

Entrez les ID des commandes à réinitialiser. Vous pouvez les séparer par une virgule (par exemple 124,3,48) ou entrez ALL pour réinitialiser toutes les commandes.

[Sauvegarder](#)

Le bouton **Options Shopify** est spécifique au CMS Shopify, il permet de paramétrer Atoo-Sync pour indiquer ce qui est autorisé à mettre à jour et réinitialiser le statut "transféré" d'une ou plusieurs commande(s).

## Produits

La case à cocher **Mettre à jour le nom de l'article** autorise, Atoo-Sync à mettre à jour le nom des articles.

La case à cocher **Mettre à jour la description de l'article** autorise, Atoo-Sync à mettre à jour la description des articles.

La case à cocher **Mettre à jour les prix** de l'article autorise, Atoo-Sync à mettre à jour les prix des articles.

La case à cocher **Utiliser la configuration de Shopify pour l'envoi des prix HT ou TTC** permet à Atoo-Sync d'appliquer le paramétrage défini dans Shopify concernant l'affichage des prix. Selon ce paramétrage, les prix seront envoyés soit en HT, soit en TTC.

La case à cocher **Mettre à jour le stock** de l'article autorise, Atoo-Sync à mettre à jour la quantité des articles.

La case à cocher **Mettre à jour le poids** de l'article autorise, Atoo-Sync à mettre à jour le poids des articles.

La case à cocher **Mettre à jour le code barre** de l'article autorise, Atoo-Sync à mettre à jour le code barre des articles.

La case à cocher **Mettre à jour le fabricant** de l'article autorise, Atoo-Sync à mettre à jour le fabricant des articles.

La case à cocher **Mettre à jour les caractéristiques de l'article** autorise Atoo-Sync à mettre à jour les caractéristiques des articles exportés.

La case à cocher **Associer les images au produit parent** permet à Atoo-Sync de rattacher les images aux produits parents des articles avec déclinaisons/variations.

La case à cocher **profile.cms.options.readProductsSleep** autorise Atoo-Sync de lire les produits en sommeil pour les exporter vers Shopify.

## Clients

La case à cocher **Remplacer les tags des clients lors de la mise à jour** autorise Atoo-Sync de mettre à jour la valeur du tag sur les fiches clients de Shopify.

La case à cocher **Remplacer les adresses des clients** lors de la mise à jour autorise, Atoo-Sync à mettre à jour les adresses des clients.

## Mise à jour des commandes

La case à cocher **Notifier le client** autorise Atoo-Sync d'envoyer un signal à Shopify de notifier le client lors de la mise à jour de sa commande.

## Réinitialiser les commandes transférées

Le champ **Réinitialiser les commandes transférées**, permet d'effectuer un RAZ du statut transféré de la commande, vous permettant ainsi de récupérer à nouveau une commande.

Entrez les ID des commandes à réinitialiser. Vous pouvez les séparer par une virgule (par exemple 124,3,48) ou entrez ALL pour réinitialiser toutes les commandes.

## Création d'une application privée dans Shopify

### Créer une application privée Shopify

Cette documentation explique pas à pas comment créer une **application privée** dans Shopify en utilisant l'**Espace Développeur (Shopify Dev Dashboard)**.

Image not found or type unknown



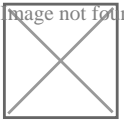
## 1. Prérequis

Avant de commencer, vous devez disposer :

- D'un **accès à l'Espace Développeur Shopify** (accès développeur du client).

**Note** : Il sera nécessaire d'avoir le compte propriétaire du client. sans ce compte, l'accès Dev Dashboard ne sera pas possible.

Image not found or type unknown



## 2. Accéder au Shopify Dev Dashboard

### depuis la liste des boutiques

1. Cliquez en haut à droite sur votre avatar.
2. Choisissez **Dev Dashboard** dans le menu.

Image not found or type unknown

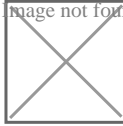


Image not found or type unknown



### 3. Créer une nouvelle application

1. Dans le Dev Dashboard, cliquez sur **Create app (Créer une application)**

Image not found or type unknown



Image not found or type unknown



2. Donnez un **nom** à l'application.  
Exemple : *Atoo-Sync GesCom*

Image not found or type unknown



3. Dans la section **URLs**, renseignez l'URL du site Atoo-Sync :

<https://www.atoo-next.net/>

Image not found or type unknown

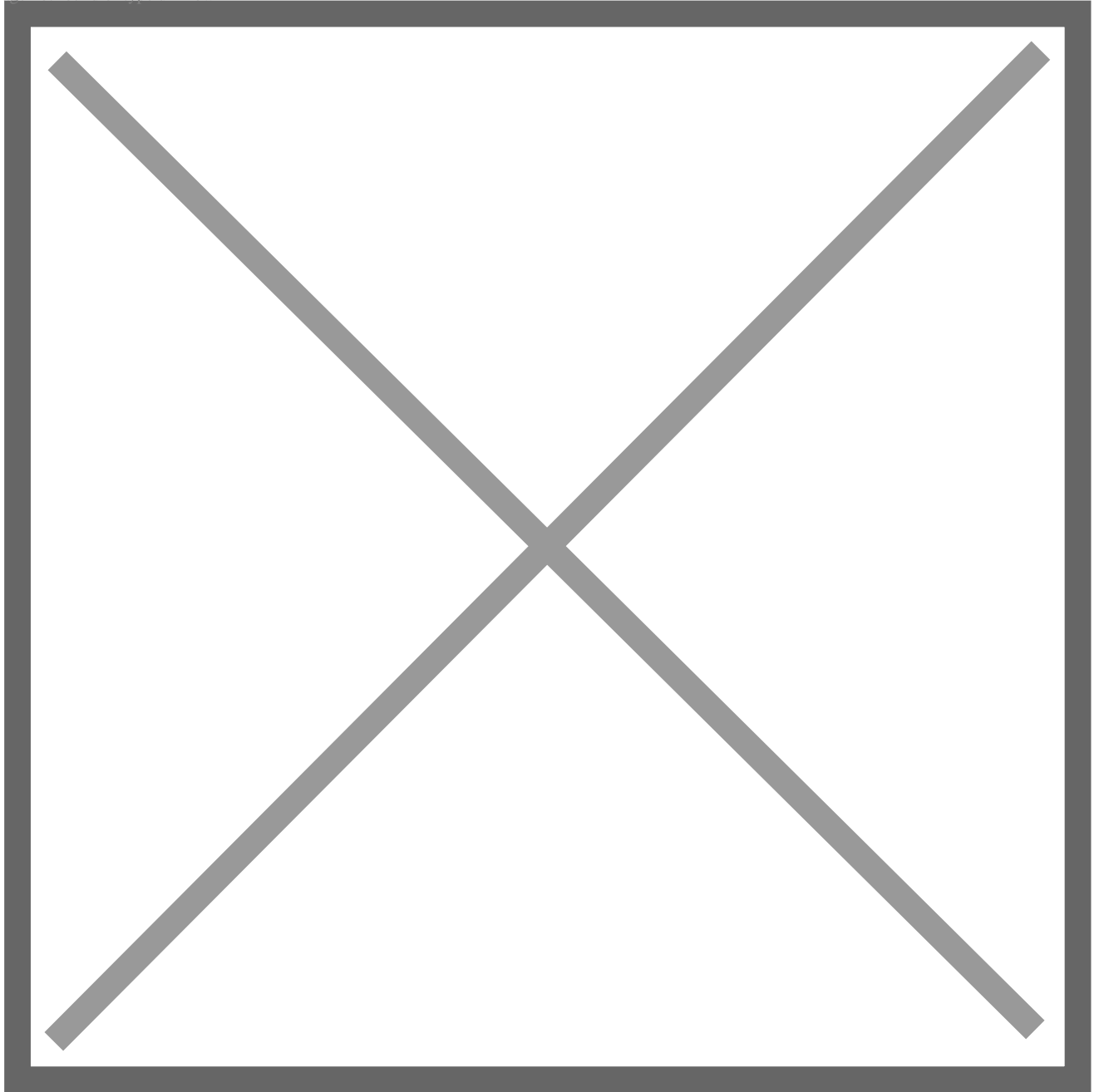


Image not found or type unknown



#### 4. Sélectionner les permissions (Scopes)

1. Descendez jusqu'à la section **Access**.
2. Cliquez sur **Select scopes**.

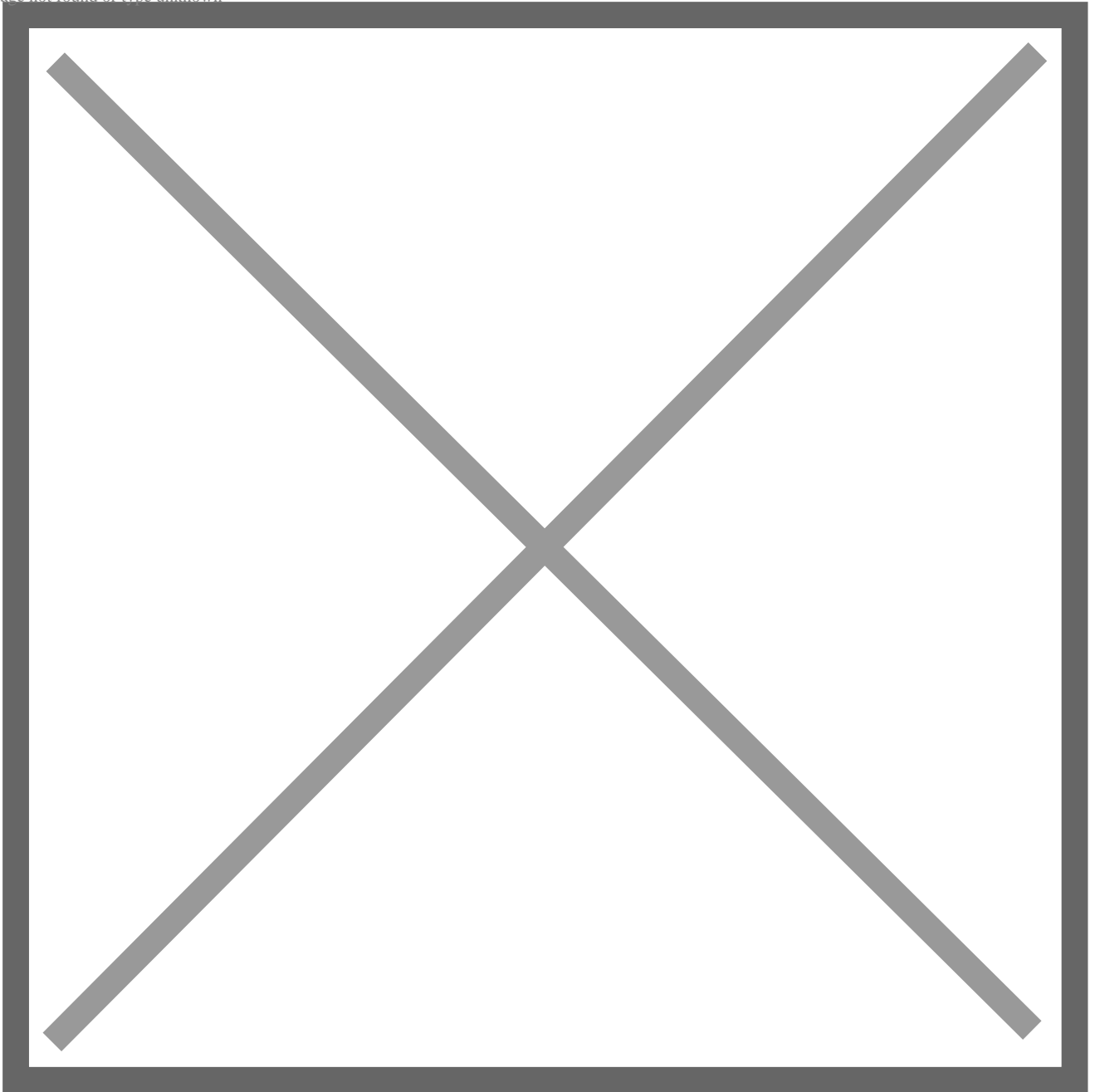
Image not found or type unknown



Décochez Use Legacy install flow (**Utiliser le flux d'installation hérité**)

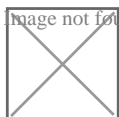
3. Une nouvelle fenêtre s'ouvre : cochez **toutes les permissions** nécessaires. (Tout autoriser)

Image not found or type unknown



4. Cliquez sur **Done**.

Image not found or type unknown



## 5. Publier la version de l'application

1. En haut à droite, cliquez sur **Release**.
2. Indiquez :
  - **Version name** : par ex. 1.0
  - **Version message** : date de création ou commentaire

Image not found or type unknown



3. Validez.

Votre application apparaît alors dans la liste des versions.

Image not found or type unknown

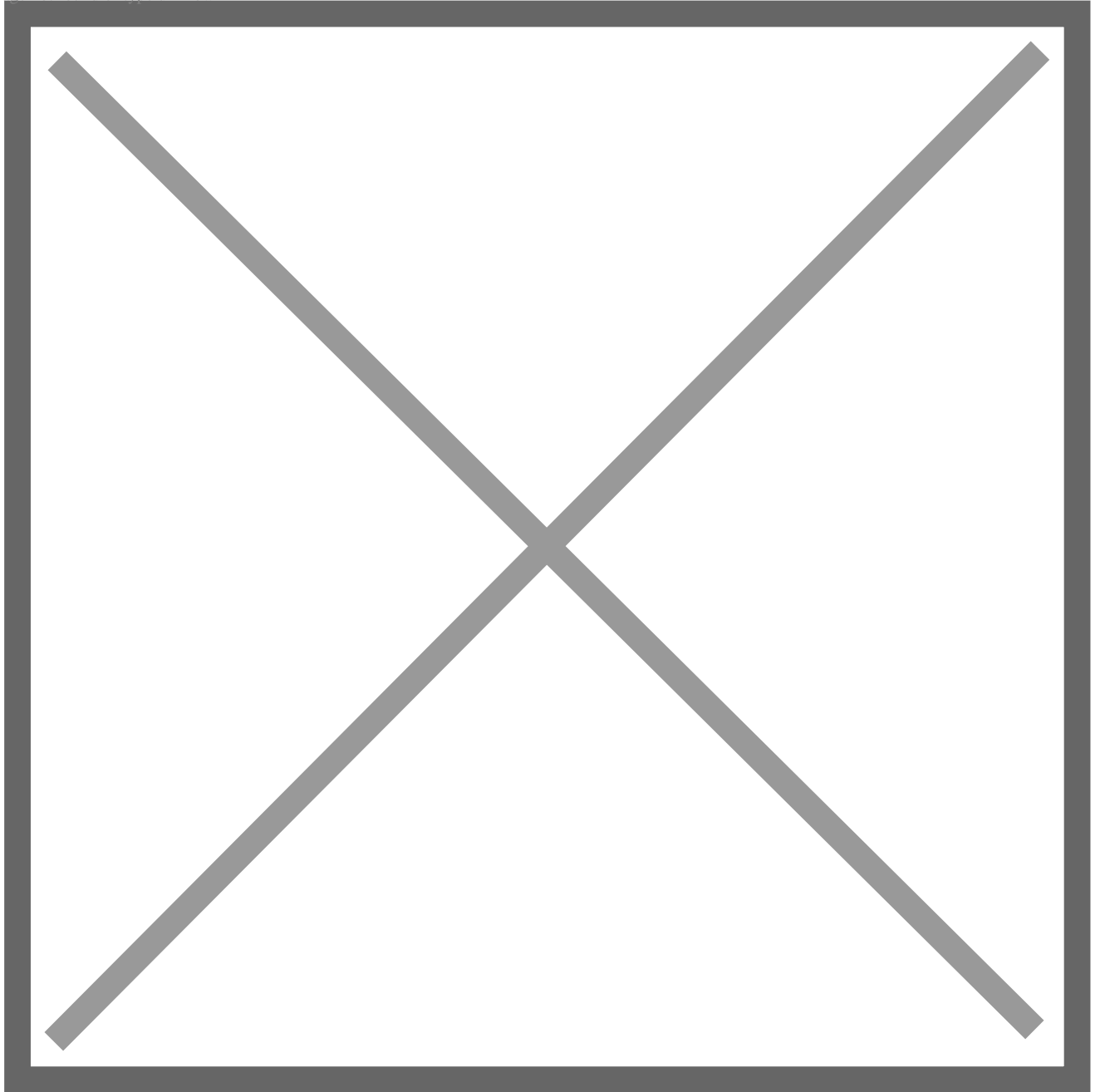


Image not found or type unknown



## 6. Installer l'application sur une boutique Shopify

1. Dans le menu de gauche, cliquez sur **Home**.
2. À droite, cliquez sur **Install app**.

Image not found or type unknown



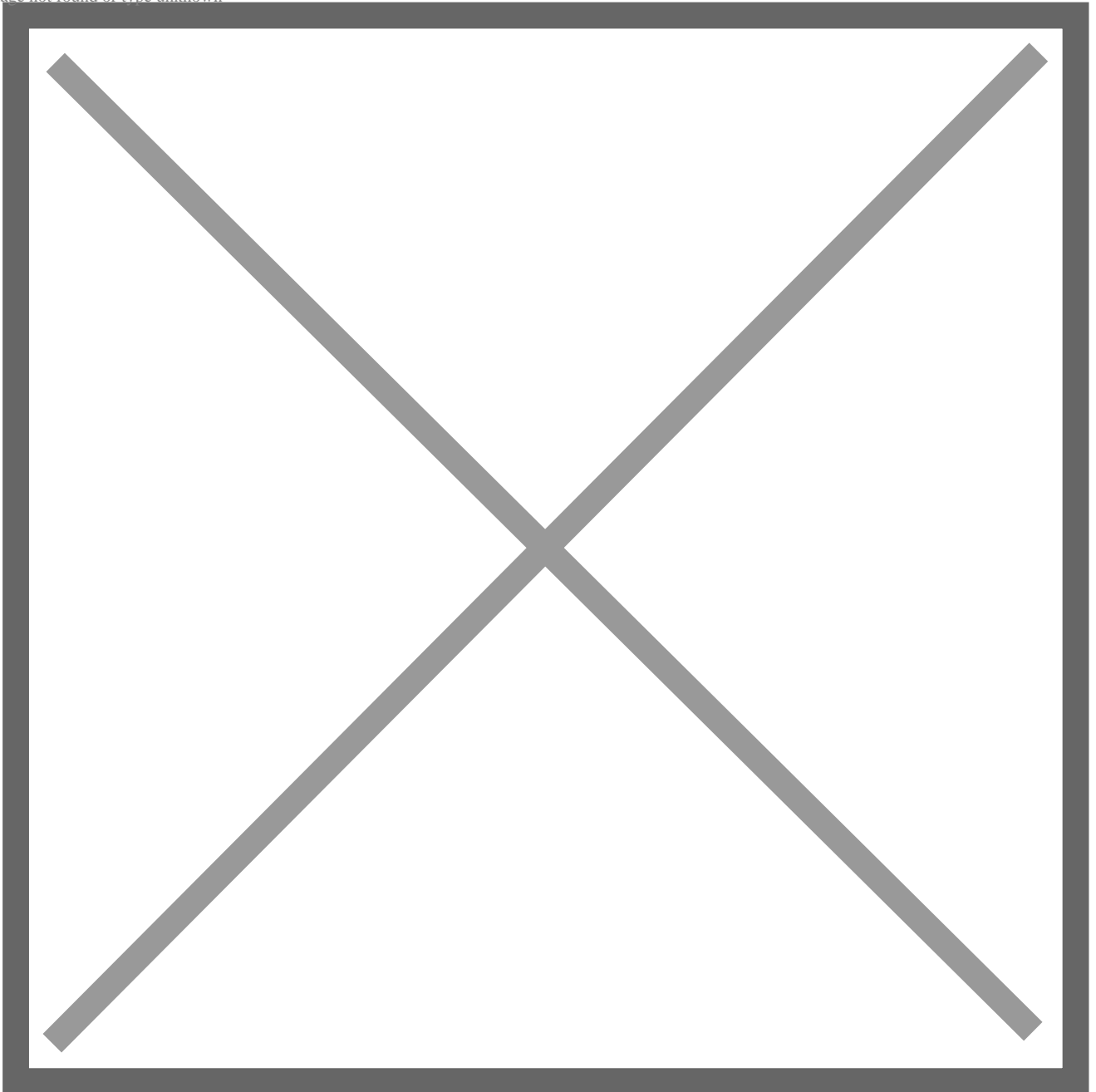
3. Choisissez la boutique cible dans la liste.

Image not found or type unknown



4. Shopify ouvre un nouvel onglet : cliquez sur **Installer**.

Image not found or type unknown



Après installation, retournez sur le Dev Dashboard :  
Le compteur **Installs** doit afficher **1**.

Image not found or type unknown

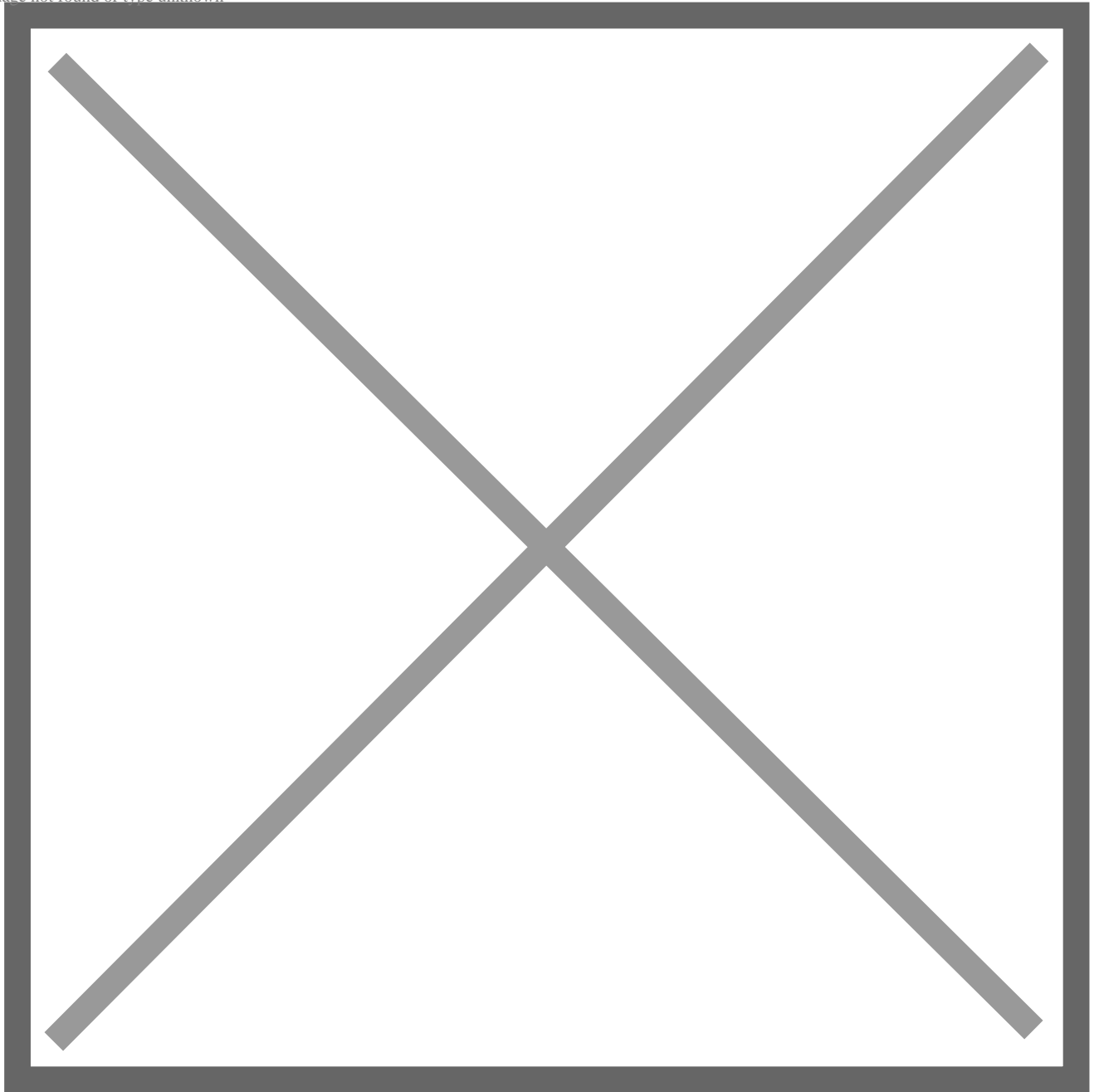


Image not found or type unknown



## 7. Récupérer les identifiants API

Dans le Dev Dashboard :

1. Ouvrez l'application créée.
2. Allez dans **Settings**.
3. Vous trouverez :
  - **Client ID**
  - **Secret**

Image not found or type unknown



Ces informations sont nécessaires pour configurer la connexion dans Atoo-Sync.

Image not found or type unknown



## 8. Configurer la connexion dans Atoo-Sync

Dans Atoo-Sync :

1. Ouvrez **Configuration du site Web**.
2. Renseignez :
  - URL de la boutique : [https://\\*\\*\\*\\*.myshopify.com/](https://****.myshopify.com/)
  - Remplacez les \*\*\*\* par le nom de votre boutique
  - **Application Client ID**
  - **Application Secret**
3. Cliquez sur **Enregistrer** puis **Tester la connexion**.

Image not found or type unknown

