

Web services & API disponibles sur Stocknet

Généralités

Le format JSON est utilisé pour les échanges de données de l'API.

Le domaine Stocknet.fr dans la syntaxe de l'URL peut être remplacé par Stocknet2.fr ou Stocknet3.fr

Le premier argument est le code du dossier Stocknet à consulter.

Le deuxième argument est le code d'un Utilisateur existant dans ce dossier.

Les valeurs 'Non autorise' dans les statuts apparaissent si l'option API est fermée dans le paramétrage du dossier ou si la fiche Utilisateur n'est pas autorisée à effectuer des consultations par API.

Les résultats sont obtenus sur une seule ligne. Si vous souhaitez des retours à la ligne pour une meilleure lisibilité, il suffit d'ajouter l'argument : &br=oui

Les trois premières rubriques sont des lectures pour effectuer des consultations ou extractions de données. La quatrième rubrique permet d'effectuer des écritures ou importations de données dans Stocknet.

1) Présence d'un article et consultation de ses stocks :

Syntaxe :

<http://www.stocknet.fr/apilectures.asp?socorig=DOSSIER&usrorig=USER&codeart=CODE>

Arguments :

Le troisième argument codeart=CODE peut être remplacé par codebarre=CODEBARRE

Les arguments suivants sont facultatifs :

- pour demander le stock global -> ajouter &stock=oui
- pour demander la liste des emplacements -> ajouter &empla=DEPOT
- pour demander les emplacements avec leurs stocks respectifs -> ajouter &emplastock=DEPOT

Réponses :

Une page au format JSON est affichée, de forme :

```
{
'Dossier' : 'DOSSIER'
'Dossier_statut' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'
'Utilisateur' : 'USER'
'Utilisateur_statut' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'

'Article' : 'CODE'
'Article_barre' : 'CODEBARRE' ( si code article absent)
'Article_statut' : 'Trouve' / 'Non trouve' / 'Identifiant non communiqué'
'Article_designation' : 'DESIG' ( si l'article est trouvé )
'Article_stock' : 'NNN' ( si demandé et si l'article est trouvé )
'Emplacements' : 'DEPOT,{ 'E1','E2','E3',...}' ( si demandé et si l'article est trouvé )
'Stocks' : 'DEPOT,{ 'E1:stock1','E2 :stock2','E3 :stock3',...}' ( si demandé et si l'article est trouvé )
}
```

2) Présence d'un client :

Syntaxe :

<http://www.stocknet.fr/apilectures.asp?socorig=DOSSIER &usrorig=USER&codecli=CODE>

Arguments :

Les 2 premiers arguments permettent d'identifier le dossier et l'utilisateur

L'argument suivant « codecli » est le code client recherché.

Il est possible d'interroger sur d'autres caractéristiques du client :

- la designation -> ajouter &clidesig=oui
- le code postal de facturation -> ajouter &clipc=oui
- la ville de facturation -> ajouter &cliville=oui
- le nom -> ajouter &clinom=oui

Réponses :

Une page au format JSON est affichée, de forme :

```
{
  'Dossier' : 'DOSSIER'
  'Dossier_statut' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'
  'Utilisateur' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'

  'Client' : 'CODE'                                ( si l'utilisateur est trouvé)
  'Client_statut' : 'Trouve' / 'Non trouve'         ( si l'utilisateur est trouvé)
  'Client_designation' : '...'                      ( si le client est trouvé)
  'Client_codepostal' : '...'                      ( si demandé et si le client est trouvé)
  'Client_ville' : '...'                          ( si demandé et si le client est trouvé)
  'Client_nom' : '...'                            ( si demandé et si le client est trouvé)
}
```

3) Extraction avec requêtes SQL

Ce procédé a des possibilités illimitées en s'appuyant sur la fonction Tâches d'extraction de Stocknet qui permet de paramétrer des requêtes SQL. Plutôt que d'obtenir le résultat de l'extraction dans un fichier CSV en Ascii délimité, il se présente dans une réponse API.

Syntaxe :

<http://www.stocknet.fr/apilectures.asp?socorig=DOSSIER&usrorig=USER&extract=CODE&arg1=&arg2=>

Arguments :

Les 2 premiers arguments permettent d'identifier le dossier et l'utilisateur

L'argument suivant « extract » est le code de la tâche d'extraction qui va être exécutée

Un argument supplémentaire "&br=oui" permet d'ajouter des retours à la ligne dans la réponse qui sera ainsi plus lisible.

Les arguments arg1 et arg2 facultatifs permettent de spécifier des sélections correspondant aux contenus #1 et #2 de votre requête paramétrée dans Stocknet.

Réponses :

Une page au format JSON est affichée, de forme :

```
{
  'Dossier' : 'DOSSIER'
  'Dossier_statut' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'
  'Utilisateur' : 'Trouve' / 'Non trouve' / 'Non autorise' / 'Non alimente'
  'Extract_statut' : 'Trouve' / 'Non trouve'
  Lignes de résultat { }
}
```

4) Ecritures dans Stocknet en API

Un préambule consiste à mettre en place un certain type d'importation dans Stocknet au menu Importations. Des tests pourront être faits avec des données contenues dans un fichier de Transfert au format CSV.

Une première possibilité est d'utiliser notre module API avec une seule ligne dont le contenu sera explicitement déclaré comme argument en méthode GET dans l'adresse URL.

Exemple pour un état de stock :

```
www.stocknet2.fr/apiecritures.asp?socorig=DOSSIER&usrorig=USER&jeton=YYY&cle=TYPEIMPORT
&contenu=ARTICLE;DEPOT;EMPLAC;LOT;15.
```

Si l'argument Jeton est spécifié (ici avec la valeur YYY) alors le champ « Connexion API » dans la fiche utilisateur devra contenir cette valeur YYY qui servira de jeton de reconnaissance.

Une deuxième possibilité est d'envoyer un ensemble complet de données en méthode POST.

L'utilisateur doit être identifié dans Stocknet comme pouvant utiliser des transferts par API.

Le type d'importation est le code de la structure d'importation visible depuis le menu Importations en cliquant sur le bouton « Voir structure d'importation » et utilisé dans la première étape ci-dessus.

Les données importées ont comme séparateur de champ le caractère ; et comme séparateur de ligne le caractère |.

Exemple depuis un formulaire :

```
<form action="http://www.stocknet2.fr/apiecritures.asp" method="post">
  <input type="text" name="socorig" value="DOSSIER" placeholder="socorig"> <br/>
  <input type="text" name="usrorig" value="USER" placeholder="usrorig"><br/>
  <input type="text" name="cle" value="TYPEIMPORT" placeholder="cle"><br/>
  <textarea name="contenu">DONNEES</textarea><br/>
  <input type="submit" name="submit" value="Envoyer" />
</form>
```

Exemple en langage PHP :

```
$url = 'http://www.stocknet2.fr/apiecritures.asp';
$data = array('socorig' => 'DOSSIER', 'usrorig' => 'USER', 'cle' => 'TYPEIMPORT', 'contenu' => 'DONNEES');
$options = array(
  'http' => array(
    'header' => 'Content-type: application/x-www-form-urlencoded\r\n',
    'method' => 'POST',
    'content' => http_build_query($data)
  )
);
$context = stream_context_create($options);
$result = file_get_contents($url, false, $context);
```

Exemple en langage ASP :

```
Dim url, data, httpRequest, postResponse

url = "http://www.stocknet2.fr/apiecritures.asp"
data = "socorig=DOSSIER&usrorig=USER&cle=TYPEIMPORT&contenu=DONNEES"

Set httpRequest = Server.CreateObject("MSXML2.ServerXMLHTTP")
httpRequest.Open "POST", url, False
httpRequest.setRequestHeader "Content-Type", "application/x-www-form-urlencoded"
httpRequest.Send data

postResponse = httpRequest.ResponseText
```