

# Tutoriel Webservice REST

## Exemple d'utilisation du service web PrestaShop : REST

Ce tutorial vous montre comment utiliser le service web de PrestaShop avec la librairie PHP en créant un « CRUD ».

### Prérequis

- Une boutique PrestaShop 1.4 installée sur un serveur Apache avec l'extension mod\_rewrite.
- Un serveur XAMP sous PHP5

### Qu'est ce que CRUD et REST ?

**CRUD** est un acronyme anglais qui signifie "Create, Read, Update, Delete" pour Créer, Récupérer, Mettre à jour et Supprimer. Ce sont les 4 opérations de base qui permettent de gérer des données dans une application.

Le service web de PrestaShop utilise une architecture REST afin d'être disponible sur un maximum de plateforme, en effet les protocoles HTTP et XML sont présent sur un nombre incalculable de plateforme.

**REST** définit une architecture représentant un ensemble des bonnes méthode à pratiquer sur le web. Elle définit plusieurs règles, dont une que nous allons décrire car elle s'apparente à CRUD.

Dans le protocole HTTP nous retrouvons 4 méthodes principales qui permettent d'effectuer des traitement sur les données et qui sont définit dans l'architecture REST, nous pouvons d'ailleurs faire cette correspondance avec CRUD :

- GET-> Retrieve (Récupérer)
- POST -> Create (Créer)
- PUT-> Update (Modifier)
- DELETE -> Delete (Supprimer)

Nous allons voir ensemble comment créer une petite application permettant d'effectuer ces 4 opérations sur les clients ("customers" en anglais).

Les chapitres 1, 2 et 3 sont obligatoires.

Vous allez voir dans les chapitres suivant comment interagir avec le service web avec chacune des opérations de CRUD pour vous donner les clés pour faire un CRUD complet.

Si vous ne voulez que récupérer des données, par exemple dans l'élaboration d'une application web de notification des commandes alors vous pouvez ne vous intéresser qu'au chapitre 4.

Si vous préférez développer une application plus complète, les chapitre 4 à 7 vous intéresseront.

Chapitre 1 - Mise en place : Création des accès dans le Back Office  
Chapitre 2 - Découverte : Tester l'accès au service web avec le navigateur  
Chapitre 3 - Premiers pas : Accéder au service web et lister les clients  
3.1 - Accéder au service web grâce à la librairie  
3.2 - Gérer les erreurs  
3.3 - Lister les clients  
Chapitre 4 - Récupérer des données : Récupérer un client  
Chapitre 5 - Modification : Mettre à jour un client  
Chapitre 6 - Création : Formulaire d'ajout à distance  
Chapitre 7 - Suppression : Retirer des comptes client de la base  
Chapitre 8 - Utilisation avancée  
Chapitre 9 - Gestion des images  
Chapitre 10 - Gestion des prix  
Mémento : Notions énoncées dans ce tutorial

## Fichiers

### Bibliothèque du Web service

La dernière version du fichier **PSWebServiceLibrary.php** fichier se trouve sur notre dépôt de code : <https://github.com/PrestaShop/PrestaShop-webservice-lib/blob/master/PSWebServiceLibrary.php>

Pour télécharger ce fichier :

1. Cliquez ici pour voir le fichier brut : <https://raw.githubusercontent.com/PrestaShop/PrestaShop-webservice-lib/master/PSWebServiceLibrary.php>
2. Copiez/collez le fichier dans un fichier texte vide, en utilisant par exemple Notepad.
3. Enregistrez ce fichier sous le nom `PSWebServiceLibrary.php`

Vous pouvez également directement télécharger une archive zip de tous les fichiers de ce dépôt, y compris les fichiers d'exemple, en cliquant ici : <https://github.com/PrestaShop/PrestaShop-webservice-lib/archive/master.zip>

### Fichiers d'exemple

Tous les fichiers d'exemple peuvent être trouvés dans notre dépôt de code : <https://github.com/PrestaShop/PrestaShop-webservice-lib/tree/master/examples>

## Implémentations

- Gérer son site PrestaShop via Open ERP : [Prestashop OpenERP Connector](#) par Akretion & CampToCamp (GNU Affero GPLv3)
- Une bibliothèque Scala pour accéder au service Web de PrestaShop : [prestasac](#) par Alex Dean (GNU Affero GPLv3)
- Une bibliothèque Python pour accéder au service Web de PrestaShop : [prestapyt](#) par Guewen Baconnier (GNU Affero GPLv3)