GLPI - Utiliser l'API REST de GLPI avec cURL et Python

Présentation de l'activité

Vous avez **expérimenté** une première utilisation de l'**Gras**API REST de GLPI avec l'outil **Postman**. Cet outil est très pratique pour **tester** et **déterminer les bons paramètres** à définir dans vos requêtes d'API REST qui **utilisent les protocoles HTTTP ou HTTPS**.

Pour rappel il faut s'appuyer sur la **documentation** de l'API pour :

- connaître le point d'entrée (Endpoint) de l'API c'est à dire l'URL à utiliser ;
- connaître les types de requête proposés : GET, POST mais aussi DELETE, PUT et PATCH ;
- interpréter correctement les codes retour de réponse de l'API : 2xx (succès), 3xx (redirection) 4xx (erreur côté client), 5xx (erreur côté serveur) ;
- savoir utiliser les clés d'API pour s'authentifier et obtenir un jeton de session (session_token) ;
- fournir les paramètres nécessaire soit dans l'URL (Query string) ou dans l'entête (Header) ;
- gérer le format des données obtenues qui sont le plus souvent au format JSON.

Dans cette activité, vous allez **réutiliser** ce que vous avez expérimenté avec **Postman** pour exécuter des requêtes avec l'utilitaire **cURL** et le langage de programmation **Python**.

Préparer votre environnement Python

Vous allez utiliser la bibliothèque requests de Python. Si nécesaier mettez à jour vote environnement python :

- mettre à jour pip
- installer le module requests

```
C:> python -m pip install --upgrade pip
C:> pip install request
```

Obtenir un jeton de session (Session-Token)

Pour obtenir un jeton de session, il faut effectuer une requête Init session en renseignant des paramètres dans l'entête (Headers) :

- le jeton d'application (**App-Token**) qui a été créé pour vous permettre d'utiliser l'API de GLPI. Demandez à l'enseignant de vous communiquer cet **App-Token**.
- une authentification soit par un couple de login /mot de passe soit en utilisant un jeton d'utilisateur (*user_token*). C'est cette deuxième méthode que vous allez utiliser.

Exécution de la requête avec cURL

```
C:> curl -X GET -H "Authorization: user_token uuu..." -H "App-Token: aaa..."
    "http://path/to/glpi/apirest.php/initSession"
{"session_token":"sss..."}
```

Exécution de la requête en Python

script Python sessiontoken.py permettant d'obtenir un jeton de session <code python> import json import requests url = "http://path/to/glpi/apirest.php/initSession" headers = { "Authorization": "user_token uuu...", "App-Token": "aaa..." } reponse = requests.get(url, headers = headers) if reponse.statuscode == 200: print('success: ' + str(reponse.statuscode)) jsondata = reponse.json() print(jsondata) else: print('error: ' + str(reponse.status_code)) </code> ==== Exécution de la requête en Powershell ==== * script Powershell sessiontoken.ps1 permettant d'obtenir un jeton de session <code powershell> \$apptoken = "aaa..." \$usertoken = "uuu..." \$headers = @ { "Authorization" = "user_token" + \$usertoken "App-Token" = \$apptoken } \$baseURL = 'http://path/to/glpi/apirest.php' \$url = \$baseURL + "/initSession" \$req = Invoke-RestMethod -Uri \$url -Headers \$headers # conversion JSON ⇒ texte \$resultat = write-host \$req | ConvertFrom-Json write-host \$resultat </code> ==== Requête pour obtenir la liste des entités accessibles à un utilisateur ==== Pour la liste des entités vous devez exécuter la requête getMyEntities en renseignant des paramètres dans l'entête (Headers) : * le jeton de session Session-Token que vous venez d'obtenir ; * le jeton d'application App-Token. ==== Exécution de la requête avec cURL ==== <code shell> C:>curl -X GET -H "App-Token: aaa..." -H "Session-Token: sss..." "http://192.168.1.70/glpi/apirest.php/getMyEntities?is_recursive=true" {"myentities":[{"id":1,"name":"Entité racine > Bâtiment 1 > Etage 0"}, {"id":0,"name":"Entité racine"}]}

=== Exécution de la requête en Python ==== * script Python getmyentities.py permettant d'obtenir la liste des entités (de manière récursive) <code shell> import json import requests url = "http://path/to/glpi/apirest.php/getMyEntities?is_recursive=true" headers = { "App-Token": "aaa...", "Session-Token": "sss...", } reponse = requests.get(url, headers = headers) if reponse.statuscode == 200: print('success: ' + str(reponse.statuscode)) jsondata = reponse.json() print(jsondata) else: print('error: ' + str(reponse.status_code)) </code>

Travail à faire : Créer le script python **itemcomputer.py** permettant d'afficher les informations sur votre ordinateur renseignées dans GLPI.

Fonctionnement du script python :

- o vous aurez à renseigner en début de script ou en ligne de commande ;
 - I'ID de votre ordinateur ;
 - le type d'élément (itemtype) : Computer
 - le jeton d'application communiqué par l'enseignant : App-Token ;
- o il est nécessaire de faire un première requête afin d'obtenir le jeton de session (requête **initSession**);
- une deuxième requête doit afficher ensuite les informations sur votre ordinateur pour l'ID indiqué (requête getItem)

==== Retour Activité A8 ==== * A8 - La gestion des configurations avec GLPI

From

/ - Les cours du BTS SIO

Permanent link:

/doku.php/si7/configuration/glpi_gestionapirestpython?rev=1614941680

Last update: 2021/03/05 11:54

