

pyRefSearchUdeS.py

python 3.10+

Recherche de références dans les bases de données Scopus et de brevets US, ou de profils Scopus, pour une liste d'auteur.e.s

- Appel du script: *python.exe pyRefSearchUdeS.py*
- Les paramètres d'exécution sont lus dans le fichier *pyRefSearchUdeS.toml*
- Les données d'auteur.e.s sont lues dans un fichier Excel d'entrée
- Les résultats de la recherche sont écrits dans un fichier Excel de sortie
- Le script doit être exécuté à partir d'un poste sur le réseau universitaire pour que l'accès à Scopus soit autorisé (une "clé" est requise pour interroger l'API Scopus, voir <https://dev.elsevier.com/index.jsp>. La première exécution du script demandera à l'utilisateur de saisir la clé).

Fichier des paramètres d'exécution *pyRefSearchUdeS.toml* (voir le fichier pour plus d'infos sur les paramètres) :

- *search_type* : Type de recherche (par publications/brevets ou profils d'auteur.e.s)
- *in_excel_file* : Fichier Excel d'entrée
- *in_excel_file_author_sheet* : Feuillet dans le fichier Excel d'entrée contenant les informations des auteur.e.s
- *first_year* : Première année de la gamme d'années (recherche par publications/brevets)
- *last_year* : Dernière année de la gamme d'années (recherche par publications/brevets)
- *publication_types* : Liste des types de publications (recherche par publications/brevets)
- *local_affiliations* : Liste des institutions considérées comme des affiliations "locales"
- *scopus_database_refresh*: Intervalle de mise à jour de la copie locale de la base de données Scopus

Fichier Excel d'entrée *in_excel_file* :

Noms et identifiants Scopus des auteur.e.s spécifiés dans les colonnes suivantes du feuillet *in_excel_file_sheet_name*:

- *Nom* : Nom de famille de l'auteur.e
- *Prénom* : Prénom de l'auteur.e
- *Scopus ID* : Identifiant Scopus de l'auteur.e pour une recherche de publications/brevets (laisser la cellule vide si aucun identifiant Scopus n'est disponible)

Fichier Excel de sortie :

- Recherche de publications/brevets:

- Nom du fichier: `<in_excel_file mantissa>_publications_<first_year>-<last_year>.xlsx`
- Feuille *Résultats* : résumé des résultats de la recherche
- Feuilles *Articles*, *Conférences*, *Lettres*, *Livres*, *Chapitres de livres*, *Rapports*, *Brevets* : résultats de la recherche par type de publication (omis si aucun résultat)
- Feuille *Profils par identifiants*, colonne *Erreurs*: erreurs dans les profils d'auteur.e.s et/ou disparités entre les informations dans le fichier Excel d'entrée et la base de données Scopus
- Recherche de profils :
 - Nom du fichier: `<in_excel_file mantissa>_profils.xlsx`
- Recherche de publications/brevets ou de profils :
 - Feuille *Homonymes* ou *Profils* selon le type de recherche, colonne *Affl/ID*: repérage des noms d'auteur.e.s correspondant à plusieurs identifiants Scopus, afin de s'assurer que les bons identifiants sont fournis dans le fichier Excel d'entrée pour la recherche de publications :
 - *Affl.* : l'affiliation est comprise dans la liste *local_affiliations*
 - *ID* : l'identifiant Scopus est identique à celui dans le fichier Excel d'entrée

Problèmes / points à considérer :

- Il y a des erreurs fréquentes dans la base de données Scopus (ex : publications attribuées à tort à des auteur.e.s ayant le même nom, affiliation incorrecte, etc.). Les informations dans le fichier Excel de sortie (feuille *Homonymes* ou *Profils* selon le type de recherche, colonne *Affl/ID*) aident à repérer ces erreurs. Le cas échéant, il faut demander à la personne concernée de faire une mise à jour de son profil Scopus. La mise à jour de la base de données peut prendre jusqu'à une semaine, il faut ensuite mettre à jour la copie locale de la base de données via le paramètre *scopus_database_refresh* dans le fichier *pyScopus.toml* avant de lancer une nouvelle recherche avec le script.
- Google Scholar semble être plus fiable, en particulier pour les conférences, mais il ne semble pas y avoir de moyen facile de faire des recherches scriptées dans cette base de données.

Installation:

- Projet *pyRefSearchUdeS* disponible sur [github](#)
- `pip install -r requirements.txt`