Youtube Scraper

But : Créer un script exécutable permettant de scraper une liste de vidéos youtube et stockant les données récupérées dans un fichier au format json

#10 des Tendances

Pierre Niney : L'interview face cachée par HugoDécrypte

HugoDécrypte
749 k abonnés

Abonné

20 k

Partager

Extrait

Enregistrer

Interview réalisée à l'occasion de la sortie du film « Mascarade » réalisé par Nicolas Bedos, le 1er novembre 2022 au cinéma. Avec Pierre Niney, Isabelle Adjani, François Cluzet, Marine Vacth.

Chaleureux remerciements au cinéma mk2 Bibliothèque pour son accueil.

–

–

Présenté par Hugo Travers

Réalisateur : Julien Potié
Journalistes : Benjamin Aleberteau, Blanche Vathonne

Chargée de production déléguée : Romane Meissonnier
Assistant de production déléguée : Clément Chaulet
Chargée de production exécutive : Marie Delvallée

Chef OPV : Lucas Stoll
OPV : Pierre Amilhat, Vanon Borget
Electricien : Alex Henry
Chef OPS : Victor Arnaud
Stagiaire image : Magali Faizeau

Maquilleuse : Kim Desnoyers
Photographe plateau : Erwann Tanguy

Monteur-étalonneur : Stan Duplan
Mixeuse : Romane Meissonnier

Cheffe de projets partenariats : Mathilde Rousseau
Assistante cheffe de projets partenariats : Manon Montoriol

Quelles informations obtenir ?

- Titre de la vidéo
- Nom du vidéaste
- Nombre de pouces bleu
- Description de la vidéo (format plain text)
- Liens exceptionnels de la description (s'il y en a, par exemple, des liens vers un timestamp vidéo ou un compte Twitter)
- id de la vidéo youtube
- Les n premiers commentaires (s'ils existent)

Paramètres

Entrée:

- Fichier JSON au format suivant (input.json) :

```
{
    “videos_id”: [
        “fmsoym8I-3o”,
        “JhWZWXvN_yo”,
        …
    ]
}
```

Paramètres

Sortie:

- Fichier JSON dans un format que vous déterminerez (output.json)

Utilisation en CLI

Le script sera lançable selon les commandes suivantes (strictement) sur une machine Linux (Pas de window, mais ça vous le savez déjà) :

- python3.8 -m venv .venv
- source .venv/bin/activate
- pip install --upgrade pip
- python scrapper.py --input input.json --output output.json

Technos

- Utilisation de Python 3.8+
- Utilisation de la lib de scraping BeautifulSoup ou bien requests si vous préférez le natif
- Utilisation des fonctions map/filter/reduce dès que possible (programmation fonctionnelle, vue au cours précédent)
- Bonne pratique de développement
  - Découpage de vos fonctions les plus grandes (visez la trentaine de ligne maximum)
  - Une fonction = une action
  - Un peu d'objet si nécessaire ?
- Tests unitaires de vos fonctions avec pytest =)
  - Ne pas hésiter à en écrire autant que vous avez de fonctions
  - NB: Plus vos fonctions sont atomiques, plus elles sont simples à tester =)

Modalités d’évaluation

- TP en solo
- Rendu en fin de session Mercredi 16 novembre

Code quality & bonnes pratiques = 6 points

Code répondant à la problématique (test avec ~1000 id youtube) = 6 points

- Tests automatisés réalisés, si exceptions lors du run -> 0 points

Tests unitaires + couverture de code +80% = 8 points (barème dégressif)

- Lancement de “pytest” à la source du repo ciblant un dossier “tests”: “python -m pytest tests”
- Lancement d’un test de coverage automatique avec “pytest-cov”

Total sur 20 points

Cette note sera un complément à la note du TP de fin de module, il comptera pour 4 points sur 20 pour les IA, 2 points sur 20 pour les ICC

Modalités d’évaluation

Le TP sera mis en **public** sur Github:

- Merci de ne **PAS** m’inviter sur votre repository mais bien de le laisser **public**

Le rendu sera fait par mail:

- Il devra mentionner l’url vers votre repo Github

Points d’attention

Le scraping est une des pratiques les plus fréquentes quand il s’agit de récupérer de la donnée. Cependant certains sites rendent le scrapping plus difficile que d’autres, c’est le cas de Youtube, vous allez devoir faire attention à ce que vous arrivez à obtenir via BeautifulSoup (ou requests), il se peut que l’information que vous cherchez ne soit pas forcément à l’endroit où vous l’attendez.

Good luck !