

WEB-SCRAPING AVEC PYTHON

Objectifs d'apprentissage

Scraping web

Utilisation d'API avec requests

Manipuler les fichiers HTML avec BeautifulSoup

Prérequis

HTML/CSS

Pandas

JSON

Utilisation d'api

Programmation orientée objet

Ressources

Attention : Lire les ressources dans l'ordre

[Module requests : installation - exemple - documentation](#)

[Gestion des exceptions en Python](#)

[Tutoriel détaillé de l'utilisation du module requests \[en\]](#)

[Beautiful Soup : Installation - manipulation des fichiers HTML - liste des méthodes de BeautifulSoup](#)

[Beautiful Soup : analyseurs - navigation dans le DOM](#)

[Documentation officielle BeautifulSoup](#)

Exercices

Exercice 1

Ecrire un programme python qui récupère et affiche à la console le contenu du fichier robot.txt de fr.wikipedia.org

Pistes à explorer : requests.get(url), text

Exercice 2

Ecrire un programme python qui récupère le nombre de datasets disponibles dans <http://www.data.gov/>

Pistes à explorer : requests.get(url), BeautifulSoup, html.parser, findall(balise), get_text()

Exercice 3

Ecrire un programme python qui extrait le tag h1 à partir de <https://www.linkedin.com/>

Pistes à explorer : requests.get(url), BeautifulSoup, html.parser, content.h1

Exercice 4

Ecrire un programme python qui extrait et affiche tous les tags de type headers à partir de en.wikipedia.org/wiki/Main_Page

Pistes à explorer : requests.get(url), BeautifulSoup, html.parser, find_all(list_of_tags)

Exercice 5

Ecrire un programme python qui extrait et affiche tous les liens vers les images de la page wikipedia de la reine Elisabeth II.

Pistes à explorer : requests.get(url), BeautifulSoup, html.parser, find_all(tag)

Exercice 6

Ecrire un programme python qui affiche le nombre de followers d'un compte twitter.

Pistes à explorer : requests.get(url), BeautifulSoup, lxml, find(tag,class), io

Exercice 7

Ecrire un programme python qui affiche la météo (température, vitesse du vent, description, etc.) d'une ville donnée.

Pistes à explorer : requests.get(url), json, [openweathermap API](#)

Exercice 8

Exercice proposé par Nicolas Duval

1. Rendez-vous sur mon compte Github : nclsdvl
2. Clonez le repo : “exoScrap” La méthode find() (lignes : 34, 36 et 38) de beautiful soup manque d’argument pour trouver les éléments recherchés.
3. Modifier l’adresse du repo afin de pouvoir faire un push sur votre Github

Brief projet

Analyse de l’évolution des meilleures performances en triple saut de 1891 à 2019.

Etape 1

Analyse le contenu du site <http://trackfield.brinkster.net/Main.asp?P=F>

Etape 2

Récupérer l’url des 25 meilleures performances en triple saut de l’année 2019

Etape 3

Récupérer les informations que vous jugez utiles sous forme de DataFrame

Etape 4

Ecrire un programme qui permet d’extraire les 25 meilleures performances de l’année 1891 à 2019 en triple saut homme et femme dans un fichier Excel.

Etape 5

Effectuer une étude statistique sur ces données.

