

# VéScraper

## I. Introduction

### 1. Rappel du sujet

Pour notre projet annuel, il nous a été demandé de concevoir un Micro-Langage.

Ce Micro-Langage doit nous permettre d'extraire du contenu sur internet en rapport avec notre sujet.

### 2. Application choisie

Nous avons décidé de développer un micro-langage pour remplacer le moteur de recherche Google à la façon d'un langage SQL.

Celui-ci pourra nous aider à récupérer des modèles de vélos, des images et les sites où ils sont vendus en exploitant la recherche avancée de Google.

<https://www.astuces-aide-informatique.info/9691/commandes-recherche-avancees-google>

## II. Focus sur l'application

### 1. Menu

Lorsque nous lançons l'application (en ligne de commande), nous arrivons sur un menu nous détaillant ce qu'il est possible de faire sur l'application :

Il suffit ensuite de taper la commande spéciale, ou la requête désirée pour lancer une action.

### 2. Mode débogage

Le mode débogage permet de visualiser l'arbre de décision de notre requête grâce à la génération d'un fichier PDF.

Cet arbre (sous la forme de tuples) sera également affiché dans la console.

Pour utiliser le mode débogage, il faut obligatoirement avoir installé graphviz sur sa machine.

## 3. Exemples de requêtes

Il est possible de taper différents types de requêtes afin de trouver des liens ayant du contenu, des images, ou les 2.

On doit spécifier quel est le produit que l'on recherche et sur quel site on veut récupérer les résultats.

Il est possible aussi d'ajouter des mots-clés avec des opérateurs booléens.

Enfin, on peut spécifier la limite du nombre de résultats qui par défaut est à 10.

# III. Choix d'implémentations

## 1. Langage choisi

Pour ce projet, nous avons choisi d'utiliser le langage python, car c'est selon nous celui qui répondait le mieux à notre besoin. Il possède beaucoup de bibliothèques permettant de parser et traiter des arbres de décision, ce qui nous a permis d'aller relativement vite.

## 2. Technologies et bibliothèques principales utilisées

Afin de passer correctement nos requêtes pour le langage, il a fallu utiliser différentes bibliothèques dont 2 principales.

### Ply

Ply est un outil d'analyse écrit uniquement en python il s'agit d'une ré implémentation de Lex Yacc à l'origine en langage c.

**Lex** : Générateur d'analyseur lexical.

Prends en entrée la définition des unités lexicales

Produit un automate fini minimal permettant de reconnaître les unités lexicales

**Yacc** : Générateur d'analyseur syntaxique.

Prends en entrée la définition d'un schéma de traduction (produit par Lex)

Produit un analyseur syntaxique pour le schéma de traduction.

## Graphviz

Graphviz est un logiciel de visualisation graphique open source. La visualisation de graphes est un moyen de représenter des informations structurelles sous forme de diagrammes, de graphes abstraits et de réseaux.

Il a des applications importantes dans les réseaux, la bio-informatique, le génie logiciel, la conception de bases de données et de sites Web, l'apprentissage automatique et les interfaces visuelles pour d'autres domaines techniques.

Dans notre cas, il est utilisé dans le mode débogage pour visualiser notre arbre de décision générée par Ply.

# IV. Bilan du projet

## 1. Problèmes rencontrés

### Construction de la requête Google

N'ayant pas trouvé de documentation officielle de Google sur toutes les requêtes avancées possibles, il a fallu chercher sur des forums et essayer nous-mêmes différentes combinaisons avant de trouver celle qui fonctionnait le mieux.

Il fallait donc adapter le langage aux combinaisons possibles pour que cela fonctionne.

### Théorie d'un langage de programmation

Même si nous avons eu des cours cette année, la théorie des langages reste un domaine que nous ne maîtrisons difficilement dans le groupe et il a fallu redoubler d'efforts pour arriver à créer un nouveau langage.

## 2. Conclusion

Pour conclure, cette application a été difficile à réaliser par la complexité de ce qu'est un langage de programmation que par le fait de trouver une idée concrète qui puisse venir enrichir le projet de base. L'application finale nous a tout de même été très utile au moment du remplissage de la base de données, car elle nous a permis de trouver très rapidement des modèles de vélo électrique ainsi

que des images associées

---

Revision #1

Created 2022-09-27 18:39:36 UTC by Noé Larrieu-Lacoste

Updated 2022-09-27 19:00:24 UTC by Noé Larrieu-Lacoste