

P5 / Utiliser les données publiques d'Open Food Facts

Lien du projet : <https://github.com/CoLoDot/P5>

1 - Vision d'ensemble

Le projet 5 a pour but de développer un programme en ligne de commande, utilisant le SGBDR (système de gestion de base de données relationnelle) MySQL et l'API OpenFOODfacts. Ce projet requiert une analyse et un apprentissage préalable du fonctionnement des bases de données relationnelles ainsi que des API. L'intégralité de ce projet est réalisé en python 3.4.

2 - La démarche de production

En premier lieu, j'ai étudié le fonctionnement des API et de MySQL par le biais des cours prodigués par OpenClassrooms. Par la suite, j'ai effectué des tests de récupérations de données sur le site OpenFoodFacts, crée ma première base de données en ligne de commande et découvert les concepts de clefs primaires et étrangères.

Après une phase de tests de plusieurs semaines, j'ai pu améliorer l'ensemble de mes scripts python et les scinder afin de clarifier mon code. La phase finale de la production a donc impliqué la rédaction du présent document et l'analyse de mon travail, l'optimisation des scripts, de la base de données, et la création d'une présentation pour la soutenance

Par ailleurs, le but de ce projet était également de mettre en place un système de gestion de projet Agile, en utilisant un tableau de type Trello. Cela m'a permis de planifier les tâches que j'avais à réaliser, l'ordre dans lequel les traiter et de leur affecter des deadlines.

Tableau public Trello :

<https://trello.com/b/YRj0zbli/projet-5-mysql-api>

3 - L'algorithme

L'algorithme dans son ensemble:

1. Lancement du programme en ligne de commande
2. Création de la base de données relationnelle MySQL « OPENFOODFACTS » en local ainsi que ses tables (avec clef étrangère)
3. Récupération des données avec la librairie Requests via l'API OpenFoodFacts

4. Stockage des données dans une liste Python
5. Insertion des données récupérées dans les tables de la BDD précédemment créée
6. Affichage du menu principal permettant au client de choisir entre le choix d'une catégorie de produit, l'affichage de ses produits sauvegardés ou de quitter le programme.
7. Si le client décide d'afficher une catégorie de produit, il peut choisir d'en sauvegarder un à la fois. A terme, les produits sauvegardés sont accessibles via le menu principal
8. A tout moment, le client peut quitter le programme. Si il fait ce choix, toutes les données sont effacées.

Afin de réaliser au mieux le présent projet, il a fallu importer des librairies permettant d'effectuer les requêtes venant d'OpenFoodFacts et pour insérer les données dans la base de données. Pour ce faire, j'ai utilisé la librairie Requests, permettant, par le biais du format JSON, de récupérer des données. Ainsi, j'ai pu les stocker dans une liste python. Une fois cette liste remplie, c'est par le biais de la librairie MySQL.connector que j'ai pu insérer les données dans la base de données.

Il est possible donc possible de résumer l'algorithme aux trois mots suivants : **récupération, stockage, insertion.**

4 - Les difficultés et leurs solutions

La première difficulté fut la récupération des données via la librairie Requests. De fait, même si j'avais déjà pu utiliser le format JSON auparavant, j'ai eu quelques difficultés à récupérer des données liées à certaines clefs du dictionnaire des produits OpenFoodFacts. J'ai cependant réussi à palier à ces difficultés à force de tests lors de la première phase de production.

Par la suite, des difficultés sont apparues au niveau de l'insertion des données dans la base de données. De fait, j'ai fait nombre d'erreurs dans mes scripts tests concernant les requêtes SQL (problème de syntaxe lié à mon apprentissage et la découverte d'un nouveau langage). C'est en m'entraînant que j'ai pris de meilleures habitudes et intégré la syntaxe de ce nouveau langage.

Afin de palier aux nombreuses erreurs auxquelles j'ai du faire face, j'ai fréquenté quotidiennement la documentation MySQL et mysql.connector et j'ai repris étape par étape les exemples python afin de comprendre d'où venaient les

« bugs » de mon programme. Le site StackOverFlow fut également une grande source d'information concernant MySQL.