Guide d’étapes clés : Utilisez les bases de Python pour réaliser une analyse de marché

Comment utiliser ce document ?

Ce guide vous propose un découpage du projet en étapes. Vous pouvez suivre ces étapes selon vos besoins. Dans chacune, vous trouverez :

- des recommandations pour accomplir la mission ;
- les points de vigilance à garder en tête ;
- une estimation de votre avancement sur l’ensemble du projet (attention, celle-ci peut varier d’un apprenant à l’autre).

Suivre ce guide vous permettra :

- d’organiser votre temps ;
- de gagner en autonomie ;
- d’utiliser les cours et ressources de façon efficace ;
- de mobiliser une méthodologie professionnelle que vous pourrez réutiliser.

Gardez en tête que votre progression sur les étapes n’est qu’une estimation, et sera différente selon votre vitesse de progression.

Recommandations générales

- Pendant que vous travaillez, stockez votre code dans un repository GitHub et faites des commits fréquents.
- N'oubliez pas de faire un commit d’un fichier requirements.txt, mais ne stockez pas l'environnement virtuel dans le repository.
- Assurez-vous que les fichiers d’output (par exemple, les fichiers CSV ou d'image) ne sont pas stockés dans le repository.

Étape 1 : Mettre en place l'environnement de développement

10% de progression

Avant de démarrer cette étape, je dois avoir :

- Lu le projet entier, y compris le document des exigences.

Une fois cette étape terminée, je devrais avoir :

- Configuré l'IDE de mon choix.
- Installé Python.
- Créé un compte GitHub.

Recommandations :

- La majeure partie du temps que vous passerez à développer des applications sera consacrée à travailler dans un IDE (environnement de développement intégré), il est donc important d'être à l'aise avec l'IDE que vous choisissez.
    - Passez un peu de temps à explorer différents IDE. Étudiez les ressources ci-dessous sur la configuration d'un IDE et d'autres d'installer et d'essayer des IDE différents afin de vous aider à choisir celui sur lequel vous souhaitez travailler.
- Vous devrez également vous familiariser avec la ligne de commande dans un terminal pour de nombreuses tâches de développement, notamment pour exécuter un script Python et gérer un repository Git.

Points de vigilance :

- Python est un langage et un écosystème en constante évolution.

Ressources :

- Pour vous aider à configurer l'IDE, les chapitres suivants du cours Mettez en place votre environnement Python vous seront utiles :
    - Choisissez l'éditeur le plus approprié à votre cas contient des détails sur certains IDE disponibles.
    - À partir du chapitre Téléchargez, installez et exécutez PyCharm, le cours contient des détails sur PyCharm.
- Si vous souhaitez utiliser Visual Studio Code (VS Code) comme IDE, suivez le cours OpenClassrooms Mettez en place votre environnement front-end, en particulier les deux premiers chapitres.
    - Bien que ce cours soit destiné à ceux qui mettent en place un environnement de développement front-end, il est également applicable à ceux qui veulent utiliser VS Code pour Python.

    - Si vous le choisissez, il serait utile d'explorer les extensions VS Code pertinentes pour le développement sur Python.
- Pour vous aider à configurer le repository GitHub et à l'utiliser pour travailler sur le projet, suivez le cours Gérer du code avec Git et GitHub.
- Le cours Apprendre à utiliser la ligne de commande dans un terminal vous aide à maîtriser les commandes basiques qui sont nécessaires pour gérer les fichiers et les dossiers de votre ordinateur.

Étape 2 : Extraire les données d'un seul produit

25% de progression

Avant de démarrer cette étape, je dois avoir :

- Compris la structure HTML de la page Produit et identifié chacun des champs obligatoires dans la structure HTML de la page.
- Compris la structure du fichier CSV :
    - La définition des headers.
    - L'utilisation de la virgule comme délimiteur de champ et de la nouvelle ligne comme délimiteur de ligne.
    - L'échappement des délimiteurs du texte du champ, ce qui importe particulièrement car certaines descriptions peuvent contenir des virgules et des nouvelles lignes.

Une fois cette étape terminée, je devrais avoir :

- Écrit un script qui permet de :
    - Extraire les données de **la page sélectionnée**.
    - Stocker ces données dans un fichier local au format CSV.

Recommandations :

- Commencez par choisir n'importe quelle page Produit (c'est-à-dire un seul livre) sur Books to Scrape, et écrivez un script Python qui visite cette page et extrait les informations détaillées dans le document des exigences.
- Les données doivent être enregistrées dans un fichier CSV en utilisant les champs extraits comme titres de colonne.

Points de vigilance :

Assurez-vous de bien comprendre les points suivants :

- Comment analyser un fichier HTML pour identifier de manière unique comment extraire chaque champ de la page.
    - Il s'agira notamment de comprendre les balises, les IDs et les noms de classe, et de sélectionner les filles/sœurs dans la structure et le contenu HTML.

- Ce qu'est une structure de fichier CSV. Même si le résultat de cette phase n'est qu'un fichier CSV de deux lignes (une ligne de header et une ligne de données), une bonne compréhension de la structure générale des fichiers CSV constitue une bonne base pour le reste du projet.
- Nommage des fichiers : c'est une bonne occasion d'explorer les options de nommage des fichiers outputs.
    - Bien que vous puissiez choisir n'importe quel nom pour le fichier, il est important d'utiliser une extension qui identifie correctement le type de fichier (.csv). Vous devez également choisir un nom qui décrit l'output (par exemple, le nom de l'élément, le type d'output, les horodatages).

Étape 3 : Extraire toutes les données des produits d'une catégorie

50% de progression

Avant de démarrer cette étape, je dois avoir :

- Complété le script qui extrait les données d'un seul produit et les enregistre dans un fichier CSV local.

Une fois cette étape terminée, je devrais avoir :

- Écrit un script qui peut réussir à :
    - Extraire les données de **tous les livres de la catégorie sélectionnée**.
    - Stocker ces données dans un fichier local au format CSV.

Recommandations :

- Vous devez choisir n'importe quelle catégorie de livres (listée dans la colonne de gauche de la page d'accueil) sur Books to Scrape et écrire un script Python qui visite la page de cette catégorie et extrait l'URL de la page Produit pour chaque livre de la catégorie.
- Combinez ce script avec ce que vous avez effectué lors de l'étape précédente pour extraire les données produit de chaque livre de la catégorie choisie et pour inscrire les données dans un seul fichier CSV.
    - Ne réinventez pas la roue ! Un code réussi n'a pas besoin de partir de zéro.

Points de vigilance :

- Certaines pages de catégories contiennent plus de 20 livres et les livres sont répartis sur plusieurs pages.

    - Vous devrez comprendre la pagination et savoir comment la gérer en écrivant du code qui détecte la présence d'un lien "next" sur la page.
    - S'il y a un lien "next" sur la page, le script doit visiter ce lien et extraire les informations de cette page.
- Le nommage des fichiers (comme à l'étape précédente) est une occasion d'explorer les options pour nommer des fichiers outputs.

Ressources :

- Lisez les ressources suivantes pour en savoir plus de les fichiers CSV, que vous utiliserez pour stocker les données que votre code extrait :
    - How-to Geek : What Is a CSV File, and How Do I Open It? (rédigé en anglais)
    - La page Wikipedia sur les CSV

Étape 4 : Extraire toutes les produits de toutes les catégories

75% de progression

Avant de démarrer cette étape, je dois avoir :

- Complété le script qui extrait les données de tous les produits d'une seule catégorie et les enregistré dans un fichier CSV local.

Une fois cette étape terminée, je devrais avoir :

- Écrit un script qui peut réussir à :
    - Extraire les données de **tous les livres de toutes les catégories**.
    - Générer un fichier au format CSV pour **chaque catégorie**.

Recommandations :

- Écrivez un script qui visite la page d'accueil de Books to Scrape et extrait les liens vers toutes les catégories de livres disponibles.
- Combinez ce script avec ce que vous avez effectué lors de l'étape précédente. Le code doit visiter la page de chaque catégorie et extraire les données produit pour tous les livres de chaque catégorie.
- Inscrivez les données de chaque catégorie de livre dans un fichier CSV distinct.

Points de vigilance :

- Vous devrez peut-être revoir les conventions de nommage des fichiers du script. La convention de nommage des fichiers choisis précédemment fonctionne-t-elle en termes de clarté du contenu de plusieurs fichiers, ou devez-vous l'améliorer ?

Étape 5 : Extraire et enregistrer les fichiers images

90% de progression

Avant de démarrer cette étape, je dois avoir :

- Complété le script qui extrait toutes les catégories et leurs livres et enregistre les résultats dans un fichier CSV local.

Une fois cette étape terminée, je devrais avoir :

- Ajouté les fonctionnalités suivantes au script :
    - Télécharger les images associées de tous les livres de toutes les catégories.
    - Sauvegarder les fichiers images localement.

Recommandations :

- Vous devez étendre la partie du script qui extrait les détails d'un livre spécifique pour télécharger et enregistrer le fichier image du livre.
- Vous devrez choisir un module Python qui vous permet de télécharger les fichiers images.
- Portez une attention particulière aux chemins relatifs et à la façon dont vous devez les convertir en chemins absolus pour que le script puisse les télécharger.

Points de vigilance :

- N'oubliez pas que vous pouvez transformer les données extraites (dans ce cas, le nom de fichier de l'image) avant de les enregistrer.
- Nommage aux fichiers de téléchargement d'images : améliorez la manière de choisir une convention de nommage des fichiers. Il doit être facile de savoir quelle image appartient à quel livre (par exemple, seront-ils tous dans un seul dossier ou auront-ils une structure de dossiers imbriqués ? Quel(s) attribut(s) sera utilisé pour le nom du fichier ?).

Étape 6 : Terminer les livrables rédigés

100% de progression

Avant de démarrer cette étape, je dois avoir :

- Écrit tout le code.
- Terminé le fichier ZIP des données de données nettoyées et préparées (les données extraites et les images associées).

Une fois cette étape terminée, je devrais avoir :

- Rédigé le mail au responsable d'équipe, Sam, décrivant comment l'application permet d'établir une pipeline ETL.
- Rédigé un fichier README.md et l'ajouté au repository en donnant des instructions pour exécuter le code avec succès et sortir des données.

Recommandations :

- Assurez-vous que vous avez envoyé le commit d'un fichier requirements.txt au repository et que vous n'avez pas ajouté l'environnement virtuel.

Points de vigilance :

- Vous devrez savoir comment configurer un fichier .gitignore pour le repository.
- Vous pouvez envisager le formatage Markdown pour styliser le contenu du fichier README, en particulier les styles de base tels que les titres et les blocs de code.
- Le mail doit être bref et précis. N'ayez pas peur de rester simple.

Ressources :

- Ce guide de styles Markdown (rédigé en anglais) vous aidera à formater ce que vous rédigez dans le repository.
- Pour vous aider à configurer un fichier .gitignore pour votre repository, lisez la documentation officielle sur gitignore (rédigé en anglais).

Projet terminé !