L'objectif de ce projet est de prédire, via l'apprentissage automatique, le prix de vente d'une maison ou d'un appartement en fonction de sa ville, sa surface, son diagnistique de performance énergétique, etc.

Pour ce faire, on s'appuiera sur le site `https://www.immo-entre-particuliers.com/`, qui a l'avantage de ne pas posséder de protection contre les bots. Par respect pour les propriétaires du site, on veillera à limiter au maximum le nombre de requêtes. En particulier, on s'assurera d'avoir un code fonctionnel avant de scraper l'intégralité des annonces, pour éviter les répétitions.

Instructions

Ce projet est à réaliser impérativement en binôme : les projets individuels seront pénalisés.
Seules les bibliothèques spécifiquement mentionnées dans le sujet sont autorisées.
Le code pour le premier jalon devra être uploadé sur Eprel au plus tard le **14 février 2025**.

Premier jalon : récupération des données en `python`

Tout projet de machine learning s'appuie sur un ensemble massif de données. L'objectif de cette première partie est la récupération de ces données, et leur stockage dans le format CSV.
Cette partie est à coder en `python`.

Étudier la page d'une maison ou d'un appartement à vendre sur le site
`https://www.immo-entre-particuliers.com/`.
Une telle page est donnée en exemple dans les Figures 1 et 2.

Question 1 En utilisant les bibliothèques `requests` et `Beautiful soup` comme on l'a vu dans le TP2, écrivez une fonction `getsoup()` qui prend en entrée l'URL d'une annonce, et renvoie la soupe correspondant à cette page HTML.

Question 2 Dans la suite, on va éliminer un certain nombre d'annonces qui ne correspondent pas à certains critères (par exemple les parkings, fonds de commerce, ou les annonces trop peu chères, qui correspondent souvent à une proposition d'échange plutôt qu'à une vente).

Pour ce faire, les fonctions chargées d'extraire les informations importantes de la soupe lèveront une exception `NonValide`.

Créez donc une classe `NonValide`, héritant de la classe `Exception`.

Question 3 En haut de chaque page (cf. l'encadré rouge sur la Figure 1) est indiqué le prix de l'annonce.

Écrivez une fonction `prix()` qui prend en entrée la soupe d'une annonce, et renvoie son prix, sous forme de chaine de caractères et sans le symbole `"€"`. On ne considérera pas les annonces de moins de 10.000€ (qui sont probablement des propositions d'échanges de maison ou d'appartement) : dans ce cas, `prix()` devra lever une exception `NonValide`.

Question 4 Écrivez une fonction `ville()` qui attend en entrée la soupe d'une annonce et renvoie la ville dans laquelle se trouve le bien. Cette information est disponible en haut de la page, comme indiqué en bleu sur la Figure 1. Plus précisément, la ville est située après la dernière occurrence de la sous-chaîne de caractères ", ". On pourra utiliser la fonction `rfind()` de python.

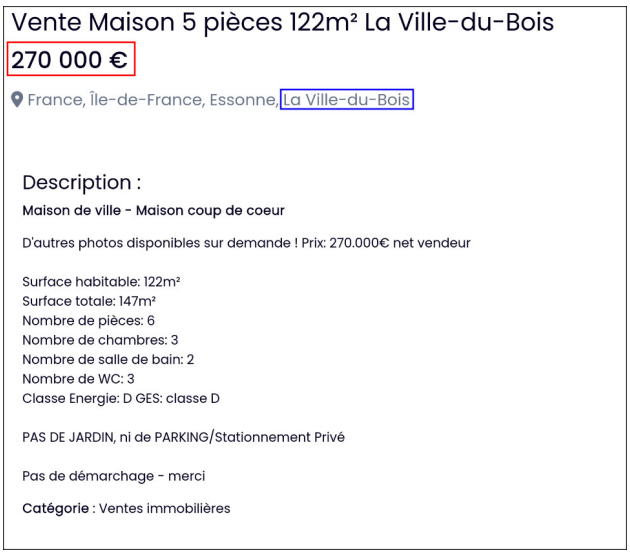

FIGURE 1 – Le prix (en rouge) et la ville (en bleu) de l'annonce.

On va maintenant se pencher sur les autres caractéristiques des annonces. Parmi toutes celles répertoriées sur le site, nous en retiendrons 6, en bleu sur la Figure 2 : le type de bien, sa surface, le nombre de pièces, de chambres, de salles de bain, et le DPE (diagnostique de performance énergétique). Ces informations sont indiquées sous le titre "Caractéristiques" (en rouge sur la Figure 2).

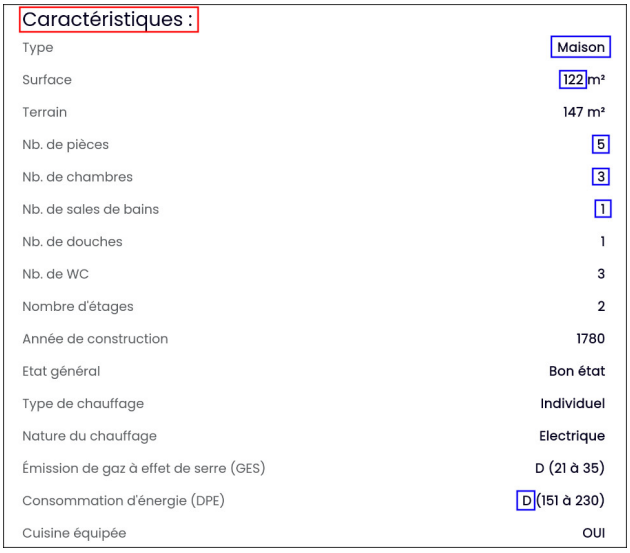

FIGURE 2 – Les caractéristiques de l'annonce. Celles qui nous intéressent sont encadrées en bleu.

Question 5 Écrivez des fonctions `type()`, `surface()`, `nbrpieces()`, `nbrchambres()`, `nbrsdb()` et `dpe()`, qui prennent une soupe en argument et renvoient la valeur de la caractéristique associée (sous forme de chaîne de caractères).

On pourra commencer par écrire une fonction renvoyant la balise (sous forme de soupe) contenant toutes les caractéristiques. Pour cela, on pourra chercher le header `"Caractéristiques :"`, comme indiqué en rouge sur la Figure 2.

Dans la mesure où l'on ne cherche que des maisons et des appartements, `type()` lèvera une exception `NonValide` si le type n'est ni `"Maison"` ni `"Appartement"`.

Pour les cinq autres fonctions, si la donnée est manquante, on renverra la chaîne `"-"`. **En particulier, les annonces aux caractéristiques partielles seront acceptées, du moment qu'il s'agit bien de maisons ou d'appartements d'une valeur d'au moins 10.000€.**

Question 6 Écrivez une fonction `informations()` qui attend la soupe d'une annonce en argument, et renvoie une chaîne de caractères contenant toutes les informations de l'annonce, séparées par des virgules, dans l'ordre :

`"Ville,Type,Surface,NbrPieces,NbrChambres,NbrSdb,DPE,Prix"`

Sur l'annonce représentée dans les Figures 1 et 2, cette fonction renverra donc :

`"La Ville-du-Bois,Maison,122,5,3,1,D,270000"`

Cette fonction `informations()` lèvera donc une exception `NonValide` en cas d'annonce non conforme.

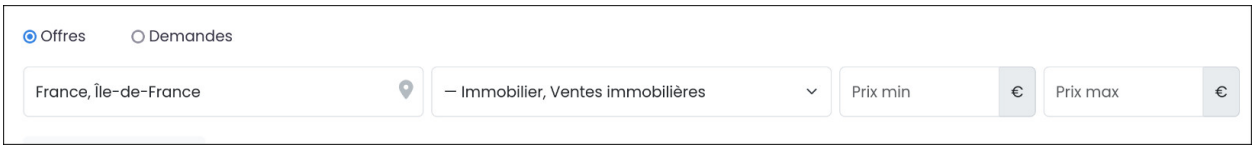

FIGURE 3 – Recherche des offres de ventes immobilières en Île-de-France.

Question 7 Nous allons limiter nos recherches aux annonces de ventes immobilières dans la région Île-de-France. On veillera à se limiter aux offres (cf. Figure 3).

En étudiant l'URL et la structure des différentes pages de résultats (il devrait y en avoir autour d'une centaine), écrivez un script qui parcourt toutes les annonces proposées par le site et appelle `informations()` sur les soupes correspondantes (en attrapant les exceptions `NonValide` soulevées).

Les résultats obtenus devront être enregistrés, à raison d'une ligne par annonce, dans un fichier CSV (comma-separated values) dont la première ligne indiquera les étiquettes (sans guillemets) des différents champs :

```
Ville,Type,Surface,NbrPieces,NbrChambres,NbrSdb,DPE,Prix
```

Pour la suite du projet, on travaillera à partir de ce fichier CSV local.

À titre indicatif, il y avait 449 annonces valides (et donc 450 lignes au total dans le fichier CSV) au jour où nous rédigeons cet énoncé. Ce nombre est évidemment susceptible d'évoluer en fonction des annonces ajoutées ou retirées du site, mais les variations sont probablement minimes. Si vous déviez beaucoup de ce nombre, c'est que votre code est sans doute incorrect.