

# Documentation

Le projet est composé de 6 fichiers python :

- **main.py** : utilisé pour appeler les autres fichiers que nous importons dans celui-ci. Les autres fichiers ont des fonctions « `exec()` » qui reprennent tout le code, afin d'appeler un programme on saisit `*nom*.exec()`
- **create\_CSS.py** : Est composé de 3 fonctions qui créer des fichiers .css :
  1. `creation_CSS_acceuil(path)` : pour stylisé les pages d'accueil des fichiers HTML des choix 1 et 2 de **main.py** .
  2. `creation_CSS_joueurs(path)` : pour stylisé les pages des joueurs des fichiers HTML des choix 1 et 2 de **main.py** .
  3. `creation_CSS_matches(path)` : pour stylisé la page HTML des matchs de la coupe du monde (choix 3 de **main.py**).

Ces fonctions seront appelées dans les autres fichiers python, elles ont pour argument « path », ce qui signifie que chacun des fichiers spécifiera un chemin pour l'enregistrement de ces fichiers .css

- **one\_squad\_data.py** : Ce fichier utilise le package Selenium avec geckodriver.exe fournit dans le projet, pour récupérer l'ensemble des données nécessaires au résultat attendu c'est-à-dire obtenir toutes les informations sur une équipe de la coupe du monde. Nous avons ici accompagné Selenium de BeautifulSoup4, le combo des deux permet plus de flexibilité avec un accès rapide sur des balises simples avec BS4 et un accès plus lent mais bien plus profond sur des balises/données plus complexes, nous avons récupéré les données sur <https://www.rugbyworldcup.com/2023/> car c'est ici que nous avons le plus de données sur les joueurs et les images étaient plus « pro ». Ensuite après que toutes les données aient été récupérées et stockées dans un dataframe pandas, on créer des pages HTML pour chaque joueur ainsi qu'une page d'accueil pour l'équipe, les fichiers CSS sont appelés depuis le fichier **create\_CSS.py**. Enfin le programme créer le dossier « Choice\_1 » s'il n'existe pas et ajoute l'ensemble des fichiers HTML et CSS dans un dossier « Web\_Files ». L'utilisateur sera directement amené sur la page web d'accueil grâce au module webbrowser et en spécifiant le chemin du fichier d'accueil.
- **create\_CSV.py** : A pour but de créer des fichiers CSV (une quarantaine), qui seront nécessaires au fichier **all\_squads\_data.py**. Le code s'inspire de la partie collecte de données de **one\_squad\_data.py** mais récupèrent les données de tous les joueurs de toutes les équipes, impliquant une exécution plus longue (environ 30 minutes). Après collecte des données celles-ci sont séparées par équipe on obtient donc 20 data frame pour les informations des joueurs de chaque équipe, et 20 autres data frame avec leurs statistiques. On extrait ensuite ces fichiers au

format CSV dans un dossier « CSV\_Files », en plus des 40 fichiers on y ajoute un autre fichier CSV avec le nom des équipes, leur identifiant unique, et les images associées.

- **all\_squads\_data.py** : Relativement similaire à **one\_squad\_data.py** mais ne collecte pas les données, ils les récupèrent depuis les fichiers CSV générés auparavant. Ensuite création d'une page d'accueil HTML où on pourra sélectionner une équipe, puis après la sélection de l'équipe on est sur la même chose que les pages de **one\_squad\_data.py**, il en va de même pour les pages des joueurs. Nous avons décidé d'opérer comme cela car la collecte de données pour chaque équipe est d'environ une minute et trente secondes, désormais on crée une fois les CSV grâce à **create\_CSV.py** et on peut générer toutes les équipes en quelques secondes.
- **Matches.py** : Ici on utilise seulement BeautifulSoup4 car on parcourt seulement 2 pages :
  - [https://fr.wikipedia.org/wiki/Coupe\\_du\\_monde\\_de\\_rugby\\_%C3%A0\\_XV\\_2023](https://fr.wikipedia.org/wiki/Coupe_du_monde_de_rugby_%C3%A0_XV_2023)  
Pour récupérer les classements des poules. Choix de ce site car récupération très simple du tableau final de chaque poule
  - <https://www.rugbyrama.fr/resultats/rugby/coupe-du-monde/calendrier>  
Pour récupérer les résultats de l'ensemble des matchs et les infos. Choix de ce site car beaucoup d'infos (date, lieu, score, ...) sur les matchs sans parcourir des dizaines d'URL

Après récupération de l'ensemble des matchs dans un data frame, on crée des data frame par équipe, par poules/phases grâce à la fonction groupby de pandas.

Enfin on crée un site qui a une barre de navigation avec les différentes poules et phases, puis on crée une section avec les résultats de chaque équipe. On a aussi créé un dictionnaire de traduction afin de tout mettre en anglais pour correspondre aux autres sites générés par les fichiers sur les joueurs et équipes.