

Exercice 1

Récupérer une clé d'authentification sur le site de api-sports [Ici](#)

Pour récupérer des données sur la saison d'une équipe de football de ligue 1 en 2022 sur le site api-sports, on utilise la requête GET `"/teams/statistics?season=2022&team=85&league=61"`.

Ici le nombre 85 désigne l'identifiant du PSG .

Le nombre 61 désigne l'identifiant de la ligue 1.

Le programme suivant se connecte au site **api-sports** en utilisant une clé d'authentification (api-key)

```
1 import requests
2 from math import inf
3 import json
4 url= 'https://v3.football.api-sports.io/teams/statistics'
5 headers = {
6 'x-rapidapi-host': "v3.football.api-sports.io",
7 'x-rapidapi-key': api_key}
8 }
9
10 params= { 'season': '2022', 'team': '81', 'league': '61' }
11 response = requests.get(url, headers=headers ,params=params)
12 j = response.json()
```

1. Quelles sont les clés du dictionnaire `resultat`?
2. Ecrire une fonction `lire_cles` pour afficher les clés d'un dictionnaire .
3. Quelle est l'équipe concernée par ce programme?
4. Que renvoie la commande `resultat['response']`?
5. Examinez la réponse retournée par l'instruction `resultat['response']['goals']`.
 - (a) Donner le nombre de buts marqués par l'équipe à domicile et à l'extérieur.
 - (b) Donner le nombre de cartons rouges reçus par l'équipe.
6. le dictionnaire `equipes` donne le nom et l'identifiant de équipes de ligue1:

```
equipe = {'Lille':79 , 'Lyon': 80 , 'Marseille' : 81, 'Montpellier':82,'PSG':85 , 'Nice':84,'Monaco':91}
```

Ecrire une fonction `buts_marques(equipes)` qui prend pour paramètre un dictionnaire du type `equipe` et qui renvoie le total des buts marqués par toutes les équipes inscrites dans le dictionnaire.

7. Ecrire une fonction `cartons_rouge(equipe,id)` qui prend pour paramètre un dictionnaire du type `equipe` et l'identifiant `id` d'une équipe et qui renvoie le total de cartons rouges pris par l'équipe d'identifiant `id` au cours de la saison 2022.

Exercice 2 (utilisation de l'api NeoWs (Near Earth Object Web Service))

Cette api permet de récupérer des données sur des astéroïdes ayant approché la planète Terre

- Récupérer une clé d'authentification sur le site de la NASA [Ici](#)
- Dans la rubrique *Browse APIS* , lire la documentation sur cette api.
- Utilisez ce que vous avez appris dans ce qui précède pour compléter le programme suivant qui permet de récupérer les données sur les astéroïdes ayant approché la planète Terre.

```
1  import requests
2  import json
3  api_key = ".." # à remplacer par la clé d'authentification
4  url = "http://api.nasa.gov/neo/rest/v1/feed?/"
5  date_début='..' # à compléter
6  date_fin='..' # à compléter
7  params= .....
8  response = requests.get(url, params=params)
9
10 data = json.loads(response.text)
11 print(data)
12
13
```

- Ecrire une fonction *plus_gros_astro(date)* qui prend pour paramètre une date donnée (au format année-mois-jour) et qui renvoie le nom et le diamètre (max) du plus gros astéroïde ayant approché la Terre à *date*.
- De même , écrire une fonction *astero_plus_proche(date)* qui renvoie le nom et la distance (min_dist) de l'astéroïde ayant approché le plus près la planète Terre à *date*