



Web scraping Ligues

As we saw last time how to scrap a web page in google teams table for only botola pro, in this small article we going to try to scrap all ligues around the world using the web site

Foot Mercato : Info Transferts Football - Actu Foot Transfert
Suivez toute l'actu du football en direct : Info transfert, mercato et résultats. L'actualité des joueurs et des clubs de foot en France et en Europe avec Footmercato
<https://www.footmercato.net/>



FULL Analysing:

Analysing the URL:

Botola pro ⇒ www.footmercato.net/maroc/botola-pro/classement

Serie-A ==> www.footmercato.net/italie/serie-a/classement

Ligue-1==> www.footmercato.net/france/ligue-1/classement

if we notice the URL of the rank table of every ligue contain:

- the country
- the ligue

So, we can easily make the URL we manipulate dynamic, we need only to change the Country and The Ligue

Analysing the HTML:

#	Équipe	Pts	J	G	N	D	BP	BC	DIF
= 1	PSG	32	12	10	2	0	32	5	27
^ 2	Lens	27	12	8	3	1	19	8	11
▼ 3	Lorient	26	11	8	2	1	21	14	7
▼ 4	Marseille	23	12	7	2	3	20	9	11
^ 5	Rennes	21	11	6	3	2	23	11	12
▼ 6	Monaco	21	11	6	3	2	20	14	6
= 7	Lille	19	11	6	1	4	21	18	3
^ 8	Lyon	17	12	5	2	5	21	16	5
▼ 9	Clermont	17	11	5	2	4	15	16	-1

for example, this ligue-1 ranking table if we inspect the html, we will notice that:

```

▼ <tr class="ranking--qualifying">
  ► <td class="classement__acronym">...</td>
  <td class="classement__rank"> 1 </td> == $0
  ► <td class="classement__img">...</td>
  ► <td class="classement__team">...</td>
  <td class="classement__highlight">32</td>
  <td>12</td>
  <td>10</td>
  <td>2</td>
  <td>0</td>
  <td class="noSmall">32</td>
  <td class="noSmall">5</td>

```

- The td tags of ranking are classed by “classement__rank”
- The td tags of team names are classed by “classement__team”
- the td rags of team point are classed by “classement__highlight”

And that's what we all need for giving simple ranking table

Coding phase:

First of All, we import the BeautifulSoup lib:

```

from bs4 import BeautifulSoup
import requests

```

Second, we create a function that use a Soup Object and Str as inputs and return elements defined by the Str as the class :

```

def findbyID(str,soup):
    try:
        b=soup.find_all("td",class_=str)
        classment=[]
        for i in b:
            classment.append(i.getText())
        Classment=[]
        for i in classment:
            Classment.append(' '.join(i.split()))
        return Classment
    except:
        print(str,"not found")

```

explicitly:

```
b=soup.find_all("td",class_=str)
classment=[]
```

this part is to get all td tag elements that are defined by a certain class

```
for i in b:
    classment.append(i.getText())
classment=[]
```

and this part is to get the text inside the tags

```
for i in classment:
    classment.append(' '.join(i.split()))
```

to remove all spaces and new lines

third to get the data and print it to console:

```
def getclassement(Country,Ligue):
    try:
        data="https://www.footmercato.net/"+Country+"/"+Ligue+"/classement"
        page = requests.get(data)
        page
        soup = BeautifulSoup(page.content, 'html.parser')
        classment=findbyID("classement_rank",soup)
        team=findbyID("classement_team",soup)
        points=findbyID("classement_highlight",soup)
        for (a, b, c) in zip(classment, team, points):
            print(a,"-----",b,"-----",c,"\n")
    except:
        print("not found")
```

to explains:

```
data="https://www.footmercato.net/"+Country+"/"+Ligue+"/classement"
page = requests.get(data)
page
soup = BeautifulSoup(page.content, 'html.parser')
```

As we talk about in analysing the URL, we concatenate the inputs of the function (Country and Ligue) with the URL and start using it

```
classment=findbyID("classement_rank",soup)
team=findbyID("classement_team",soup)
points=findbyID("classement_highlight",soup)
```

by using the function `findbyId(class,soup_object)` we retrive the data of the ranking table and finally print it to the console using the code :

```
for (a, b, c) in zip(classment, team, points):
    print(a,"-----",b,"-----",c,"\n")
```

Test case:

Country ⇒ maroc

Ligue ==> botola-pro

```
get the country maroc
get the ligue [redacted]
```

```
get the country maroc
get the ligue [botola-pro]
```

Voilaaaaaa:

get the country maroc	
get the ligue botola-pro	
1 ----- FAR Rabat -----	12
2 ----- FUS Rabat -----	12
3 ----- Wydad AC -----	11
4 ----- Olympic Safi -----	11
5 ----- JS Soualem -----	10
6 ----- UTS -----	10
7 ----- El Jadida -----	8
8 ----- Tétouan -----	8
9 ----- HUSA -----	8
10 ----- Chabab Moham. -----	8
11 ----- Maghreb Fès -----	6
12 ----- Raja -----	6
13 ----- MC Oujda -----	5
14 ----- Berkane -----	2
15 ----- Khouribga -----	2
16 ----- IR Tanger -----	1