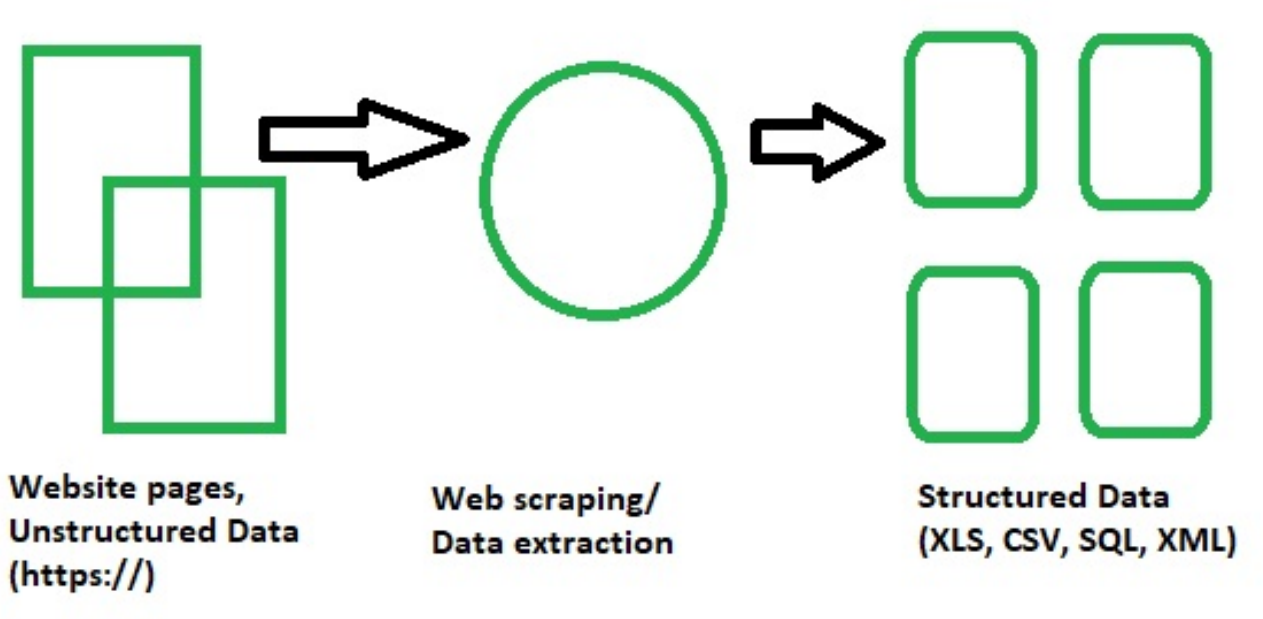
Open in Colab

Data Processing and Cleaning To Extract Information From The Scraped Data

In [1]:
```
from IPython import display
display.Image("/content/drive/MyDrive/Data_Engineering/1.png", width = 1200, height = 600)
```

Out[1]:



Scraping Data Of Football Players Using Python

In [2]:
```
!pip install lxml
!pip install html5lib
!pip install beautifulsoup4
!pip install csv
import pandas as pd
import requests
import numpy as np
from pandas import DataFrame
from pprint import pprint
from typing import List
from datetime import datetime
import csv
from google.colab import files
from google.colab import drive
!pip install openpyxl
#import pandas
#dir(pandas)
#!pip install beautifulsoup4
#dir('beautifulsoup4')
#!pip install selenium
#dir('selenium')
url = "http://sofifa.com/players?offset=0"
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: lxml in /usr/local/lib/python3.8/dist-packages (4.9.2)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: html5lib in /usr/local/lib/python3.8/dist-packages (1.0.1)
Requirement already satisfied: webencodings in /usr/local/lib/python3.8/dist-packages (from html5lib) (0.5.1)
Requirement already satisfied: six>=1.9 in /usr/local/lib/python3.8/dist-packages (from html5lib) (1.15.0)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: beautifulsoup4 in /usr/local/lib/python3.8/dist-packages (4.6.3)
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
ERROR: Could not find a version that satisfies the requirement csv (from versions: none)
ERROR: No matching distribution found for csv
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: openpyxl in /usr/local/lib/python3.8/dist-packages (3.0.10)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.8/dist-packages (from openpyxl) (1.1.0)
```

In [3]:
```
!pip install pandas-datareader
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: pandas-datareader in /usr/local/lib/python3.8/dist-packages (0.9.0)
Requirement already satisfied: lxml in /usr/local/lib/python3.8/dist-packages (from pandas-datareader) (4.9.2)
Requirement already satisfied: requests>=2.19.0 in /usr/local/lib/python3.8/dist-packages (from pandas-dataread
er) (2.25.1)
Requirement already satisfied: pandas>=0.23 in /usr/local/lib/python3.8/dist-packages (from pandas-datareader)
(1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.8/dist-packages (from pandas>=0
.23->pandas-datareader) (2.8.2)
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.8/dist-packages (from pandas>=0.23->pand
as-datareader) (1.21.6)
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.8/dist-packages (from pandas>=0.23->panda
s-datareader) (2022.7)
Requirement already satisfied: chardet<5,>=3.0.2 in /usr/local/lib/python3.8/dist-packages (from requests>=2.19
.0->pandas-datareader) (4.0.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.8/dist-packages (from requests>=2.1
9.0->pandas-datareader) (2022.12.7)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.8/dist-packages (from requests>=2.19.0->p
andas-datareader) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.8/dist-packages (from requests>=
2.19.0->pandas-datareader) (1.24.3)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.8/dist-packages (from python-dateutil>=2.7.3-
>pandas>=0.23->pandas-datareader) (1.15.0)
```

In [4]:
```
header = {
  "User-Agent": "Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/50.0.2661.75 Saf
  "X-Requested-With": "XMLHttpRequest"
}

r = requests.get(url, headers=header)

dfs = pd.read_html(r.text)
```

In [5]:
```
print(len(dfs))
```

```
1
```

In [6]:
```
# Get first table
df = dfs[0]
```

In [7]:
```
# Get first table
df = dfs[0]

# Extract columns
df2 = df.iloc[:, 1:9]
df2.dtypes
```

Out[7]:
```
Name               object
Age                 int64
OVA                 int64
POT                 int64
Team & Contract    object
Value              object
Wage               object
Total               int64
dtype: object
```

In [8]:
```
DF = pd.DataFrame(df2)
DF # [['Age','OVA','POT']]
```

Out[8]:

	Name	Age	OVA	POT	Team & Contract	Value	Wage	Total
0	F. Balogun ST	20	71	79	Reims Jun 30, 2023 On Loan	€4.1M	€35K	1657
1	E. Fernández CM CDM	21	80	87	Benfica 2022 ~ 2027	€42.5M	€13K	2169
2	Balde LB LM RB	18	76	87	FC Barcelona 2021 ~ 2024	€15M	€27K	1819
3	A. Ounahi CM	22	73	83	Angers SCO 2021 ~ 2026	€7M	€17K	1876
4	K. Mitoma LM LW LWB	25	75	78	Brighton & Hove Albion 2021 ~ 2025	€7.5M	€51K	1943
5	M. Mudryk LM CAM	21	75	85	Chelsea 2023 ~ 2031	€12.5M	€57K	1829
6	D. Núñez ST LW	23	82	89	Liverpool 2022 ~ 2028	€61.5M	€120K	2075
7	B. White RB CB	24	80	84	Arsenal 2021 ~ 2026	€28M	€69K	1791
8	João Cancelo LB RB	28	87	87	Manchester City 2019 ~ 2027	€72M	€210K	2252
9	22 K. Miura ST	54	56	56	Yokohama 2005 ~ 2022	€0	€700	1267
10	N. Madueke RW	20	77	86	Chelsea 2023 ~ 2030	€23M	€72K	1896
11	M. Ødegaard CAM CM	23	85	89	Arsenal 2021 ~ 2026	€77.5M	€115K	2125
12	H. Souttar CB	23	73	77	Stoke City 2016 ~ 2025	€3.9M	€18K	1508
13	W. Weghorst ST	29	79	79	Manchester United Jun 30, 2023 On Loan	€17.5M	€59K	1900
14	K. Kvaratskhelia LW CAM	21	79	88	Napoli 2022 ~ 2027	€39.5M	€46K	1949
15	G. Rutter ST	20	75	85	Leeds United 2023 ~ 2028	€12.5M	€30K	1752
16	Jorginho CM CDM	30	84	84	Chelsea 2018 ~ 2023	€40.5M	€150K	2120

17	Pedro Porro RWB RM	22	82	88	Sporting CP 2022 ~ 2025	€46M	€16K	2157
18	W. Saliba CB	21	82	89	Arsenal 2019 ~ 2024	€57.5M	€67K	1868
19	B. Saka RM RW	20	84	90	Arsenal 2018 ~ 2024	€65.5M	€90K	2143
20	Gavi CM LW RW	17	80	89	FC Barcelona 2020 ~ 2026	€43.5M	€18K	2035
21	J. Kiwior CB CDM	22	72	81	Arsenal 2023 ~ 2028	€4.9M	€37K	1696
22	M. Caicedo CDM CM	20	75	83	Brighton & Hove Albion 2021 ~ 2025	€11.5M	€35K	1955
23	M. Sabitzer CM CDM CAM	28	81	81	FC Bayern München 2021 ~ 2025	€25.5M	€72K	2179
24	L. Trossard LM ST	27	80	80	Arsenal 2023 ~ 2027	€22.5M	€88K	1975
25	F. Miretti CM CAM	18	74	88	Juventus 2021 ~ 2026	€10M	€24K	2011
26	M. Rashford LW LM ST	24	83	86	Manchester United 2014 ~ 2024	€49.5M	€140K	2101
27	J. Araujo RB RWB RM	20	70	80	LA Galaxy 2019 ~ 2025	€3.3M	€3K	1727
28	O. Zinchenko LB	25	80	82	Arsenal 2022 ~ 2027	€24.5M	€77K	2154
29	K. Mbappé ST LW	23	91	95	Paris Saint Germain 2018 ~ 2024	€190.5M	€230K	2181
30	J. Sancho LW RW LM	22	83	87	Manchester United 2021 ~ 2026	€53M	€140K	1898
31	Gabriel Martinelli LM LW	21	80	88	Arsenal 2019 ~ 2024	€47.5M	€65K	1982
32	N. Zaniolo CF RW RM	22	80	85	Roma 2018 ~ 2024	€33M	€66K	2089
33	Pablo Torre CAM CM LM	19	69	86	FC Barcelona 2022 ~ 2026	€3.6M	€27K	1779
34	T. Tomiyasu RB LB	23	79	84	Arsenal 2021 ~ 2026	€24.5M	€65K	1837
35	Ansu Fati LW ST	19	79	89	FC Barcelona 2018 ~ 2027	€38M	€84K	1913
36	R. Araujo CB RB	23	83	89	FC Barcelona 2018 ~ 2026	€52M	€135K	1910
37	C. Chukwuemeka CM CAM CF	18	64	84	Chelsea 2022 ~ 2028	€1.6M	€7K	1687
38	Alejandro Garnacho LW LM	18	66	85	Manchester United 2021 ~ 2023	€2.2M	€11K	1656
39	D. Doué CAM	17	70	86	Rennes 2022 ~ 2025	€4M	€3K	1699
40	Adama Traoré RW LW ST	26	77	77	Wolverhampton Wanderers 2018 ~ 2023	€11.5M	€77K	1920
41	João Félix CF ST	22	83	89	Chelsea Jun 30, 2023 On Loan	€59M	€71K	2102
42	Fresneda RB	17	67	81	Real Valladolid 2021 ~ 2025	€2.2M	€500	1677
43	A. Mac Allister CM CAM CDM	23	78	83	Brighton & Hove Albion 2019 ~ 2025	€21M	€56K	2129
44	L. Messi RW	35	91	91	Paris Saint Germain 2021 ~ 2023	€54M	€195K	2190
45	V. Osimhen ST	23	83	89	Napoli 2020 ~ 2025	€58M	€72K	1999
46	J. Koundé CB RB	23	84	89	FC Barcelona 2022 ~ 2027	€57.5M	€150K	1926
47	E. Camavinga CM CDM	19	79	89	Real Madrid 2021 ~ 2027	€38M	€78K	2125
48	R. Cherki CAM LW RW	18	73	88	Olympique Lyonnais 2019 ~ 2023	€7M	€15K	1719
49	Vitinha ST	22	76	84	Sporting Braga 2020 ~ 2027	€17M	€14K	1872
50	Vinícius Jr. LW	21	86	92	Real Madrid 2018 ~ 2025	€109M	€200K	1985
51	E. Smith Rowe LM CAM	21	80	87	Arsenal 2017 ~ 2026	€43M	€65K	1830
52	Rafael Leão LW LM	23	84	90	Milan 2019 ~ 2024	€66.5M	€90K	1971
53	Pedri CM	19	85	92	FC Barcelona 2020 ~ 2026	€104.5M	€115K	2112
54	W. Gnonto LM RM ST	18	72	86	Leeds United 2022 ~ 2027	€5.5M	€9K	1778
55	R. Lewis RB RWB	17	65	85	Manchester City 2022 ~ 2024	€1.9M	€1K	1622
56	Danilo CDM CM	21	76	84	Nottingham Forest 2023 ~ 2029	€16M	€36K	1907
57	P. Foden LW CF CAM	22	85	92	Manchester City 2016 ~ 2027	€109.5M	€180K	2087
58	Y. Moukoko ST	17	75	89	Borussia Dortmund 2020 ~ 2026	€13M	€8K	1828
59	Antony RW RM	22	82	88	Manchester United 2022 ~ 2027	€49M	€130K	2046

In [9]:
```
DF.dtypes
```

Out[9]:
```
Name              object
Age                int64
OVA                int64
POT                int64
Team & Contract    object
Value              object
Wage               object
Total              int64
dtype: object
```

In [9]:
```
```

In [10]:
```
import re
DF['Name'] = DF['Name'].str.replace('\d+', '')
```

```
DF
```

```
<ipython-input-10-f97a7e879b69>:2: FutureWarning: The default value of regex will change from True to False in a future version.
  DF['Name'] = DF['Name'].str.replace('\d+', '')
```

Out[10]:

	Name	Age	OVA	POT	Team & Contract	Value	Wage	Total
0	F. Balogun ST	20	71	79	Reims Jun 30, 2023 On Loan	€4.1M	€35K	1657
1	E. Fernández CM CDM	21	80	87	Benfica 2022 ~ 2027	€42.5M	€13K	2169
2	Balde LB LM RB	18	76	87	FC Barcelona 2021 ~ 2024	€15M	€27K	1819
3	A. Ounahi CM	22	73	83	Angers SCO 2021 ~ 2026	€7M	€17K	1876
4	K. Mitoma LM LW LWB	25	75	78	Brighton & Hove Albion 2021 ~ 2025	€7.5M	€51K	1943
5	M. Mudryk LM CAM	21	75	85	Chelsea 2023 ~ 2031	€12.5M	€57K	1829
6	D. Núñez ST LW	23	82	89	Liverpool 2022 ~ 2028	€61.5M	€120K	2075
7	B. White RB CB	24	80	84	Arsenal 2021 ~ 2026	€28M	€69K	1791
8	João Cancelo LB RB	28	87	87	Manchester City 2019 ~ 2027	€72M	€210K	2252
9	K. Miura ST	54	56	56	Yokohama 2005 ~ 2022	€0	€700	1267
10	N. Madueke RW	20	77	86	Chelsea 2023 ~ 2030	€23M	€72K	1896
11	M. Ødegaard CAM CM	23	85	89	Arsenal 2021 ~ 2026	€77.5M	€115K	2125
12	H. Souttar CB	23	73	77	Stoke City 2016 ~ 2025	€3.9M	€18K	1508
13	W. Weghorst ST	29	79	79	Manchester United Jun 30, 2023 On Loan	€17.5M	€59K	1900
14	K. Kvaratskhelia LW CAM	21	79	88	Napoli 2022 ~ 2027	€39.5M	€46K	1949
15	G. Rutter ST	20	75	85	Leeds United 2023 ~ 2028	€12.5M	€30K	1752
16	Jorginho CM CDM	30	84	84	Chelsea 2018 ~ 2023	€40.5M	€150K	2120
17	Pedro Porro RWB RM	22	82	88	Sporting CP 2022 ~ 2025	€46M	€16K	2157
18	W. Saliba CB	21	82	89	Arsenal 2019 ~ 2024	€57.5M	€67K	1868
19	B. Saka RM RW	20	84	90	Arsenal 2018 ~ 2024	€65.5M	€90K	2143
20	Gavi CM LW RW	17	80	89	FC Barcelona 2020 ~ 2026	€43.5M	€18K	2035
21	J. Kiwior CB CDM	22	72	81	Arsenal 2023 ~ 2028	€4.9M	€37K	1696
22	M. Caicedo CDM CM	20	75	83	Brighton & Hove Albion 2021 ~ 2025	€11.5M	€35K	1955
23	M. Sabitzer CM CDM CAM	28	81	81	FC Bayern München 2021 ~ 2025	€25.5M	€72K	2179
24	L. Trossard LM ST	27	80	80	Arsenal 2023 ~ 2027	€22.5M	€88K	1975
25	F. Miretti CM CAM	18	74	88	Juventus 2021 ~ 2026	€10M	€24K	2011
26	M. Rashford LW LM ST	24	83	86	Manchester United 2014 ~ 2024	€49.5M	€140K	2101
27	J. Araujo RB RWB RM	20	70	80	LA Galaxy 2019 ~ 2025	€3.3M	€3K	1727
28	O. Zinchenko LB	25	80	82	Arsenal 2022 ~ 2027	€24.5M	€77K	2154
29	K. Mbappé ST LW	23	91	95	Paris Saint Germain 2018 ~ 2024	€190.5M	€230K	2181
30	J. Sancho LW RW LM	22	83	87	Manchester United 2021 ~ 2026	€53M	€140K	1898
31	Gabriel Martinelli LM LW	21	80	88	Arsenal 2019 ~ 2024	€47.5M	€65K	1982
32	N. Zaniolo CF RW RM	22	80	85	Roma 2018 ~ 2024	€33M	€66K	2089
33	Pablo Torre CAM CM LM	19	69	86	FC Barcelona 2022 ~ 2026	€3.6M	€27K	1779
34	T. Tomiyasu RB LB	23	79	84	Arsenal 2021 ~ 2026	€24.5M	€65K	1837
35	Ansu Fati LW ST	19	79	89	FC Barcelona 2018 ~ 2027	€38M	€84K	1913
36	R. Araujo CB RB	23	83	89	FC Barcelona 2018 ~ 2026	€52M	€135K	1910
37	C. Chukwuemeka CM CAM CF	18	64	84	Chelsea 2022 ~ 2028	€1.6M	€7K	1687
38	Alejandro Garnacho LW LM	18	66	85	Manchester United 2021 ~ 2023	€2.2M	€11K	1656
39	D. Doué CAM	17	70	86	Rennes 2022 ~ 2025	€4M	€3K	1699
40	Adama Traoré RW LW ST	26	77	77	Wolverhampton Wanderers 2018 ~ 2023	€11.5M	€77K	1920
41	João Félix CF ST	22	83	89	Chelsea Jun 30, 2023 On Loan	€59M	€71K	2102
42	Fresneda RB	17	67	81	Real Valladolid 2021 ~ 2025	€2.2M	€500	1677
43	A. Mac Allister CM CAM CDM	23	78	83	Brighton & Hove Albion 2019 ~ 2025	€21M	€56K	2129
44	L. Messi RW	35	91	91	Paris Saint Germain 2021 ~ 2023	€54M	€195K	2190
45	V. Osimhen ST	23	83	89	Napoli 2020 ~ 2025	€58M	€72K	1999
46	J. Koundé CB RB	23	84	89	FC Barcelona 2022 ~ 2027	€57.5M	€150K	1926
47	E. Camavinga CM CDM	19	79	89	Real Madrid 2021 ~ 2027	€38M	€78K	2125
48	R. Cherki CAM LW RW	18	73	88	Olympique Lyonnais 2019 ~ 2023	€7M	€15K	1719
49	Vitinha ST	22	76	84	Sporting Braga 2020 ~ 2027	€17M	€14K	1872
50	Vinícius Jr. LW	21	86	92	Real Madrid 2018 ~ 2025	€109M	€200K	1985

51	E. Smith Rowe LM CAM	21	80	87	Arsenal 2017 ~ 2026	€43M	€65K	1830
52	Rafael Leão LW LM	23	84	90	Milan 2019 ~ 2024	€66.5M	€90K	1971
53	Pedri CM	19	85	92	FC Barcelona 2020 ~ 2026	€104.5M	€115K	2112
54	W. Gnonto LM RM ST	18	72	86	Leeds United 2022 ~ 2027	€5.5M	€9K	1778
55	R. Lewis RB RWB	17	65	85	Manchester City 2022 ~ 2024	€1.9M	€1K	1622
56	Danilo CDM CM	21	76	84	Nottingham Forest 2023 ~ 2029	€16M	€36K	1907
57	P. Foden LW CF CAM	22	85	92	Manchester City 2016 ~ 2027	€109.5M	€180K	2087
58	Y. Moukoko ST	17	75	89	Borussia Dortmund 2020 ~ 2026	€13M	€8K	1828
59	Antony RW RM	22	82	88	Manchester United 2022 ~ 2027	€49M	€130K	2046

In [11]:
```
DF1=DF['Name'].str.split(" ", expand=True)
DF1
```

Out[11]:

	0	1	2	3	4	5
0	F.	Balogun	ST	None	None	None
1	E.	Fernández	CM	CDM	None	None
2	Balde	LB	LM	RB	None	None
3	A.	Ounahi	CM	None	None	None
4	K.	Mitoma	LM	LW	LWB	None
5	M.	Mudryk	LM	CAM	None	None
6	D.	Núñez	ST	LW	None	None
7	B.	White	RB	CB	None	None
8	João	Cancelo	LB	RB	None	None
9	K.	Miura	ST	None	None	None
10	N.	Madueke	RW	None	None	None
11	M.	Ødegaard	CAM	CM	None	None
12	H.	Souttar	CB	None	None	None
13	W.	Weghorst	ST	None	None	None
14	K.	Kvaratskhelia	LW	CAM	None	None
15	G.	Rutter	ST	None	None	None
16	Jorginho	CM	CDM	None	None	None
17	Pedro	Porro	RWB	RM	None	None
18	W.	Saliba	CB	None	None	None
19	B.	Saka	RM	RW	None	None
20	Gavi	CM	LW	RW	None	None
21	J.	Kiwior	CB	CDM	None	None
22	M.	Caicedo	CDM	CM	None	None
23	M.	Sabitzer	CM	CDM	CAM	None
24	L.	Trossard	LM	ST	None	None
25	F.	Miretti	CM	CAM	None	None
26	M.	Rashford	LW	LM	ST	None
27	J.	Araujo	RB	RWB	RM	None
28	O.	Zinchenko	LB	None	None	None
29	K.	Mbappé	ST	LW	None	None
30	J.	Sancho	LW	RW	LM	None
31	Gabriel	Martinelli	LM	LW	None	None
32	N.	Zaniolo	CF	RW	RM	None
33	Pablo	Torre	CAM	CM	LM	None
34	T.	Tomiyasu	RB	LB	None	None
35	Ansu	Fati	LW	ST	None	None
36	R.	Araujo	CB	RB	None	None
37	C.	Chukwuemeka	CM	CAM	CF	None
38	Alejandro	Garnacho	LW	LM	None	None
39	D.	Doué	CAM	None	None	None
40	Adama	Traoré	RW	LW	ST	None

41	João	Félix	CF	ST	None	None
42	Fresneda	RB	None	None	None	None
43	A.	Mac	Allister	CM	CAM	CDM
44	L.	Messi	RW	None	None	None
45	V.	Osimhen	ST	None	None	None
46	J.	Koundé	CB	RB	None	None
47	E.	Camavinga	CM	CDM	None	None
48	R.	Cherki	CAM	LW	RW	None
49	Vitinha	ST	None	None	None	None
50	Vinícius	Jr.	LW	None	None	None
51	E.	Smith	Rowe	LM	CAM	None
52	Rafael	Leão	LW	LM	None	None
53	Pedri	CM	None	None	None	None
54	W.	Gnonto	LM	RM	ST	None
55	R.	Lewis	RB	RWB	None	None
56	Danilo	CDM	CM	None	None	None
57	P.	Foden	LW	CF	CAM	None
58	Y.	Moukoko	ST	None	None	None
59	Antony	RW	RM	None	None	None

In [12]:
```
DF1[1].unique() #"GK" , "RM" , "ST"
```

Out[12]:
```
array(['Balogun', 'Fernández', 'LB', 'Ounahi', 'Mitoma', 'Mudryk',
       'Núñez', 'White', 'Cancelo', 'Miura', 'Madueke', 'Ødegaard',
       'Souttar', 'Weghorst', 'Kvaratskhelia', 'Rutter', 'CM', 'Porro',
       'Saliba', 'Saka', 'Kiwior', 'Caicedo', 'Sabitzer', 'Trossard',
       'Miretti', 'Rashford', 'Araujo', 'Zinchenko', 'Mbappé', 'Sancho',
       'Martinelli', 'Zaniolo', 'Torre', 'Tomiyasu', 'Fati',
       'Chukwuemeka', 'Garnacho', 'Doué', 'Traoré', 'Félix', 'RB', 'Mac',
       'Messi', 'Osimhen', 'Koundé', 'Camavinga', 'Cherki', 'ST', 'Jr.',
       'Smith', 'Leão', 'Gnonto', 'Lewis', 'CDM', 'Foden', 'Moukoko',
       'RW'], dtype=object)
```

In [13]:
```
DF1[2].unique() # 'CAM', 'GK', 'ST', 'RB', 'RM', 'LM', 'CB', 'CM', 'CDM',None, 'LB', 'CF', 'RW','LW'
```

Out[13]:
```
array(['ST', 'CM', 'LM', 'RB', 'LB', 'RW', 'CAM', 'CB', 'LW', 'CDM',
       'RWB', 'RM', 'CF', None, 'Allister', 'Rowe'], dtype=object)
```

In [14]:
```
DF1[3].unique() # 'CM', None, 'RM', 'CDM', 'CB','LM', 'LB', 'RW', 'CAM','LW', 'RWB', 'ST', 'LWB'
```

Out[14]:
```
array([None, 'CDM', 'RB', 'LW', 'CAM', 'CB', 'CM', 'RM', 'RW', 'ST', 'LM',
       'RWB', 'LB', 'CF'], dtype=object)
```

In [15]:
```
DF1[4].unique()
```

Out[15]:
```
array([None, 'LWB', 'CAM', 'ST', 'RM', 'LM', 'CF', 'RW'], dtype=object)
```

In [16]:
```
from numpy.lib.arraysetops import unique
A_Unique = ["GK" , "RM" , "ST",'CAM', 'GK', 'ST', 'RB', 'RM', 'LM', 'CB', 'CM', 'CDM','None', 'LB', 'CF', 'RW',
pd.Series(A_Unique, name='A_Unique').unique()
```

Out[16]:
```
array(['GK', 'RM', 'ST', 'CAM', 'RB', 'LM', 'CB', 'CM', 'CDM', 'None',
       'LB', 'CF', 'RW', 'LW', 'RWB', 'LWB', None], dtype=object)
```

In [17]:
```
sample_str = DF1[1]
# A list containing multiple characters, that needs to be deleted from the string.
list_of_chars = ['RM', 'GK', 'LB', 'LM', 'CB', 'LW', 'RWB', 'LWB', 'ST', 'RB', 'CM', 'None', 'CDM', "None", 'RW
# Remove multiple characters from the string
for character in list_of_chars:
    sample_str = sample_str.replace(character, '')
pd.DataFrame(sample_str)
```

Out[17]:

	1
0	Balogun
1	Fernández
2	
3	Ounahi
4	Mitoma
5	Mudryk
6	Núñez
7	White

8	Cancelo
9	Miura
10	Madueke
11	Ødegaard
12	Souttar
13	Weghorst
14	Kvaratskhelia
15	Rutter
16	
17	Porro
18	Saliba
19	Saka
20	
21	Kiwior
22	Caicedo
23	Sabitzer
24	Trossard
25	Miretti
26	Rashford
27	Araujo
28	Zinchenko
29	Mbappé
30	Sancho
31	Martinelli
32	Zaniolo
33	Torre
34	Tomiyasu
35	Fati
36	Araujo
37	Chukwuemeka
38	Garnacho
39	Doué
40	Traoré
41	Félix
42	
43	Mac
44	Messi
45	Osimhen
46	Koundé
47	Camavinga
48	Cherki
49	
50	Jr.
51	Smith
52	Leão
53	
54	Gnonto
55	Lewis
56	
57	Foden
58	Moukoko
59	

In [17]:

In [18]:
```python
sample_str2 = DF1[2]
# A list containing multiple characters, that needs to be deleted from the string.
list_of_chars2 = ['RM', 'GK', 'LB', 'LM', 'CB', 'LW', 'RWB', 'LWB', 'ST', 'RB', 'CM', 'None', 'CDM', "None", 'R
# Remove multiple characters from the string
for character in list_of_chars2:
    sample_str2 = sample_str2.replace(character, '')
print(sample_str2.to_string())
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42        None
43    Allister
44
45
46
47
48
49        None
50
51        Rowe
52
53        None
54
55
56
57
58
59
```

In [19]:
```python
### Python  Convert None to empty string
# Converting None to empty string
res = [str(i or '') for i in sample_str2]
print(res)
```

```
['', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', ''
, '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', 'Allister', '', '', '', '', '', '', '', 'Rowe', '
', '', '', '', '', '', '', '']
```

In [20]:
```python
pd_col2 = pd.DataFrame(res,columns=['Col2'])
pd_col2
```

Out[20]:

	Col2
0	
1	
2	
3	
4	

5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	
43	Allister
44	
45	
46	
47	
48	
49	
50	
51	Rowe
52	
53	
54	
55	
56	
57	
58	

59	

In [21]:
```
pd.DataFrame(sample_str2)
```

Out[21]:

	2
0	
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	
40	
41	
42	None
43	Allister
44	
45	
46	
47	
48	
49	None
50	

```
50
51    Rowe
52
53    None
54
55
56
57
58
59
```

In [22]:
```
sample_str3 = DF1[3]
# A list containing multiple characters, that needs to be deleted from the string.
list_of_chars3 = ['RM', 'GK', 'LB', 'LM', 'CB', 'LW', 'RWB', 'LWB', 'ST', 'RB', 'CM', 'None', 'CDM', "None", 'R
# Remove multiple characters from the string
for character in list_of_chars3:
  sample_str3 = sample_str3.replace(character, '')
print(sample_str3.to_string())
```

```
0     None
1
2
3     None
4
5
6
7
8
9     None
10    None
11
12    None
13    None
14
15    None
16    None
17
18    None
19
20
21
22
23
24
25
26
27
28    None
29
30
31
32
33
34
35
36
37
38
39    None
40
41
42    None
43
44    None
45    None
46
47
48
49    None
50    None
51
52
53    None
54
55
56    None
57
58    None
59    None
```

In [23]:
```
### Python  Convert None to empty string
# https://www.geeksforgeeks.org/python-convert-none-to-empty-string/
```

```
# using str()
# Converting None to empty string
resto = [str(i or '') for i in sample_str3]
print(resto)
```

```
['', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', ''
, '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '
', '', '', '', '']
```

In [24]:
```
pd_col3 = pd.DataFrame(resto,columns=['Col3'])
#pd_col3
```

In [25]:
```
for y in zip(*resto):
    print(y)
```

In [26]:
```
pd.DataFrame(sample_str3)
```

Out[26]:

	3
0	None
1	
2	
3	None
4	
5	
6	
7	
8	
9	None
10	None
11	
12	None
13	None
14	
15	None
16	None
17	
18	None
19	
20	
21	
22	
23	
24	
25	
26	
27	
28	None
29	
30	
31	
32	
33	
34	
35	
36	
37	
38	
39	None
40	
41	
42	None

43	
44	None
45	None
46	
47	
48	
49	None
50	None
51	
52	
53	None
54	
55	
56	None
57	
58	None
59	None

In [27]:

```
sample_str4 = DF1[4]
# A list containing multiple characters, that needs to be deleted from the string.
list_of_chars4 = ['RM', 'GK', 'LB', 'LM', 'CB', 'LW', 'RWB', 'LWB', 'ST', 'RB', 'CM', 'None', 'CDM', 'None', 'R
# Remove multiple characters from the string
for character in list_of_chars4:
  sample_str4 = sample_str4.replace(character, '')
print(sample_str4.to_string())
```

```
0     None
1     None
2     None
3     None
4         
5     None
6     None
7     None
8     None
9     None
10    None
11    None
12    None
13    None
14    None
15    None
16    None
17    None
18    None
19    None
20    None
21    None
22    None
23        
24    None
25    None
26        
27        
28    None
29    None
30        
31    None
32        
33        
34    None
35    None
36    None
37        
38    None
39    None
40        
41    None
42    None
43        
44    None
45    None
46    None
47    None
48        
49    None
50    None
51        
52    None
53    None
54        
55    None
56    None
57        
58    None
59    None
```

In [28]:

```python
# Converting None to empty string
rest = [str(i or '') for i in sample_str4]
print(rest)
```

```
['', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', ''
, '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '', '
', '', '', '', '']
```

In [29]:

```python
# using zip()
# to print list vertically
for x in zip(*rest):
    print(x)
```

In [30]:

```python
# Using + operator to combine two columns
##DF1["Names"] = DF1[['sample_str','sample_str2','sample_str3','sample_str4']].agg('-'.join, axis =1)

print('Concatenated String using join() and whitespaces =', " ".join(['sample_str','sample_str2','sample_str3',
```

```
Concatenated String using join() and whitespaces = sample_str sample_str2 sample_str3 sample_str4
```

In [31]:

```python
pd.DataFrame(sample_str4)
```

Out[31]:

	4
0	None
1	None
2	None

3	None
4	
5	None
6	None
7	None
8	None
9	None
10	None
11	None
12	None
13	None
14	None
15	None
16	None
17	None
18	None
19	None
20	None
21	None
22	None
23	
24	None
25	None
26	
27	
28	None
29	None
30	
31	None
32	
33	
34	None
35	None
36	None
37	
38	None
39	None
40	
41	None
42	None
43	
44	None
45	None
46	None
47	None
48	
49	None
50	None
51	
52	None
53	None
54	
55	None
56	None
57	

58	None
59	None

In [32]:
```
##How to Merge Two Pandas DataFrames on Inde##
#https://www.statology.org/pandas-merge-on-index/
pd.concat([DF1[0],pd.DataFrame(sample_str) ,pd.DataFrame(sample_str2) ,pd.DataFrame(sample_str3) ,pd.DataFrame(
```

Out[32]:

	0	1	2	3	4
0	F.	Balogun		None	None
1	E.	Fernández			None
2	Balde				None
3	A.	Ounahi		None	None
4	K.	Mitoma			
5	M.	Mudryk			None
6	D.	Núñez			None
7	B.	White			None
8	João	Cancelo			None
9	K.	Miura		None	None
10	N.	Madueke		None	None
11	M.	Ødegaard			None
12	H.	Souttar		None	None
13	W.	Weghorst		None	None
14	K.	Kvaratskhelia			None
15	G.	Rutter		None	None
16	Jorginho			None	None
17	Pedro	Porro			None
18	W.	Saliba		None	None
19	B.	Saka			None
20	Gavi				None
21	J.	Kiwior			None
22	M.	Caicedo			None
23	M.	Sabitzer			
24	L.	Trossard			None
25	F.	Miretti			None
26	M.	Rashford			
27	J.	Araujo			
28	O.	Zinchenko		None	None
29	K.	Mbappé			None
30	J.	Sancho			
31	Gabriel	Martinelli			None
32	N.	Zaniolo			
33	Pablo	Torre			
34	T.	Tomiyasu			None
35	Ansu	Fati			None
36	R.	Araujo			None
37	C.	Chukwuemeka			
38	Alejandro	Garnacho			None
39	D.	Doué		None	None
40	Adama	Traoré			
41	João	Félix			None
42	Fresneda		None	None	None
43	A.	Mac	Allister		
44	L.	Messi		None	None
45	V.	Osimhen		None	None
46	J.	Koundé			None
47	E.	Camavinga			None

48	R.	Cherki			
49	Vitinha		None	None	None
50	Vinícius	Jr.		None	None
51	E.	Smith	Rowe		
52	Rafael	Leão			None
53	Pedri		None	None	None
54	W.	Gnonto			
55	R.	Lewis			None
56	Danilo			None	None
57	P.	Foden			
58	Y.	Moukoko		None	None
59	Antony			None	None

In [33]:
```
pd_col4 = pd.DataFrame(rest,columns=['Col4'])
#pd_col4
```

In [34]:
```
##How to Merge Two Pandas DataFrames on Inde##
#https://www.statology.org/pandas-merge-on-index/
A = pd.concat([DF1[0],pd.DataFrame(sample_str) ,pd_col2,pd_col3 ,pd_col4], axis=1)
#A
```

In [35]:
```
A['Full Name'] = A[0].map(str) + ' ' + A[1].map(str) + ' ' + A['Col2'].map(str)+ ' ' + A['Col3'].map(str)+ ' ' 
A
```

Out[35]:

	0	1	Col2	Col3	Col4	Full Name
0	F.	Balogun				F. Balogun
1	E.	Fernández				E. Fernández
2	Balde					Balde
3	A.	Ounahi				A. Ounahi
4	K.	Mitoma				K. Mitoma
5	M.	Mudryk				M. Mudryk
6	D.	Núñez				D. Núñez
7	B.	White				B. White
8	João	Cancelo				João Cancelo
9	K.	Miura				K. Miura
10	N.	Madueke				N. Madueke
11	M.	Ødegaard				M. Ødegaard
12	H.	Souttar				H. Souttar
13	W.	Weghorst				W. Weghorst
14	K.	Kvaratskhelia				K. Kvaratskhelia
15	G.	Rutter				G. Rutter
16	Jorginho					Jorginho
17	Pedro	Porro				Pedro Porro
18	W.	Saliba				W. Saliba
19	B.	Saka				B. Saka
20	Gavi					Gavi
21	J.	Kiwior				J. Kiwior
22	M.	Caicedo				M. Caicedo
23	M.	Sabitzer				M. Sabitzer
24	L.	Trossard				L. Trossard
25	F.	Miretti				F. Miretti
26	M.	Rashford				M. Rashford
27	J.	Araujo				J. Araujo
28	O.	Zinchenko				O. Zinchenko
29	K.	Mbappé				K. Mbappé
30	J.	Sancho				J. Sancho
31	Gabriel	Martinelli				Gabriel Martinelli
32	N.	Zaniolo				N. Zaniolo

33	Pablo	Torre		Pablo Torre
34	T.	Tomiyasu		T. Tomiyasu
35	Ansu	Fati		Ansu Fati
36	R.	Araujo		R. Araujo
37	C.	Chukwuemeka		C. Chukwuemeka
38	Alejandro	Garnacho		Alejandro Garnacho
39	D.	Doué		D. Doué
40	Adama	Traoré		Adama Traoré
41	João	Félix		João Félix
42	Fresneda			Fresneda
43	A.	Mac	Allister	A. Mac Allister
44	L.	Messi		L. Messi
45	V.	Osimhen		V. Osimhen
46	J.	Koundé		J. Koundé
47	E.	Camavinga		E. Camavinga
48	R.	Cherki		R. Cherki
49	Vitinha			Vitinha
50	Vinícius	Jr.		Vinícius Jr.
51	E.	Smith	Rowe	E. Smith Rowe
52	Rafael	Leão		Rafael Leão
53	Pedri			Pedri
54	W.	Gnonto		W. Gnonto
55	R.	Lewis		R. Lewis
56	Danilo			Danilo
57	P.	Foden		P. Foden
58	Y.	Moukoko		Y. Moukoko
59	Antony			Antony

In [35]:

In [36]:
```
#https://www.statology.org/pandas-remove-special-characters/
DFF = DF['Team & Contract'].str.replace('\d+', '')
# df['my_column'] = df['my_column'].str.replace('\W', '', regex=True)
DFF.str.replace('\W', '', regex=True)
DFF = pd.DataFrame(DFF.str.replace('\W', ' ', regex=True))
#DFF
```

```
<ipython-input-36-3d41ba0d537d>:2: FutureWarning: The default value of regex will change from True to False in
a future version.
  DFF = DF['Team & Contract'].str.replace('\d+', '')
```

In [37]:
```
### Removing substrings from strings in a Series in Pandas###
# https://www.skytowner.com/explore/removing_substrings_from_strings_in_a_series_in_pandas
S= pd.Series(DFF['Team & Contract'])
S.str.replace("On Loan", "")
```

Out[37]:
```
0                  Reims Jun
1                    Benfica
2                FC Barcelona
3                  Angers SCO
4      Brighton   Hove Albion
5                    Chelsea
6                  Liverpool
7                    Arsenal
8              Manchester City
9                   Yokohama
10                   Chelsea
11                   Arsenal
12                 Stoke City
13      Manchester United Jun
14                     Napoli
15               Leeds United
16                   Chelsea
17                Sporting CP
18                   Arsenal
19                   Arsenal
20               FC Barcelona
21                   Arsenal
22     Brighton   Hove Albion
23          FC Bayern München
24                   Arsenal
25                  Juventus
26          Manchester United
27                  LA Galaxy
28                   Arsenal
29        Paris Saint Germain
30          Manchester United
31                   Arsenal
32                      Roma
33               FC Barcelona
34                   Arsenal
35               FC Barcelona
36               FC Barcelona
37                   Chelsea
38          Manchester United
39                    Rennes
40    Wolverhampton Wanderers
41                Chelsea Jun
42           Real Valladolid
43     Brighton   Hove Albion
44        Paris Saint Germain
45                     Napoli
46               FC Barcelona
47                Real Madrid
48         Olympique Lyonnais
49             Sporting Braga
50                Real Madrid
51                   Arsenal
52                     Milan
53               FC Barcelona
54               Leeds United
55             Manchester City
56          Nottingham Forest
57             Manchester City
58          Borussia Dortmund
59          Manchester United
Name: Team & Contract, dtype: object
```

In [38]:
```
A['Team'] = pd.DataFrame(S.str.replace("On Loan|Free", "") )
A
```

```
<ipython-input-38-52bc8f5a0458>:1: FutureWarning: The default value of regex will change from True to False in
a future version.
  A['Team'] = pd.DataFrame(S.str.replace("On Loan|Free", "") )
```

Out[38]:

	0	1	Col2	Col3	Col4	Full Name	Team
0	F.	Balogun				F. Balogun	Reims Jun
1	E.	Fernández				E. Fernández	Benfica
2	Balde					Balde	FC Barcelona
3	A.	Ounahi				A. Ounahi	Angers SCO
4	K.	Mitoma				K. Mitoma	Brighton Hove Albion
5	M.	Mudryk				M. Mudryk	Chelsea
6	D.	Núñez				D. Núñez	Liverpool
7	B.	White				B. White	Arsenal
8	João	Cancelo				João Cancelo	Manchester City
9	K.	Miura				K. Miura	Yokohama
10	N.	Madueke				N. Madueke	Chelsea
11	M.	Ødegaard				M. Ødegaard	Arsenal

12	H.	Souttar		H. Souttar	Stoke City
13	W.	Weghorst		W. Weghorst	Manchester United Jun
14	K.	Kvaratskhelia		K. Kvaratskhelia	Napoli
15	G.	Rutter		G. Rutter	Leeds United
16	Jorginho			Jorginho	Chelsea
17	Pedro	Porro		Pedro Porro	Sporting CP
18	W.	Saliba		W. Saliba	Arsenal
19	B.	Saka		B. Saka	Arsenal
20	Gavi			Gavi	FC Barcelona
21	J.	Kiwior		J. Kiwior	Arsenal
22	M.	Caicedo		M. Caicedo	Brighton Hove Albion
23	M.	Sabitzer		M. Sabitzer	FC Bayern München
24	L.	Trossard		L. Trossard	Arsenal
25	F.	Miretti		F. Miretti	Juventus
26	M.	Rashford		M. Rashford	Manchester United
27	J.	Araujo		J. Araujo	LA Galaxy
28	O.	Zinchenko		O. Zinchenko	Arsenal
29	K.	Mbappé		K. Mbappé	Paris Saint Germain
30	J.	Sancho		J. Sancho	Manchester United
31	Gabriel	Martinelli		Gabriel Martinelli	Arsenal
32	N.	Zaniolo		N. Zaniolo	Roma
33	Pablo	Torre		Pablo Torre	FC Barcelona
34	T.	Tomiyasu		T. Tomiyasu	Arsenal
35	Ansu	Fati		Ansu Fati	FC Barcelona
36	R.	Araujo		R. Araujo	FC Barcelona
37	C.	Chukwuemeka		C. Chukwuemeka	Chelsea
38	Alejandro	Garnacho		Alejandro Garnacho	Manchester United
39	D.	Doué		D. Doué	Rennes
40	Adama	Traoré		Adama Traoré	Wolverhampton Wanderers
41	João	Félix		João Félix	Chelsea Jun
42	Fresneda			Fresneda	Real Valladolid
43	A.	Mac	Allister	A. Mac Allister	Brighton Hove Albion
44	L.	Messi		L. Messi	Paris Saint Germain
45	V.	Osimhen		V. Osimhen	Napoli
46	J.	Koundé		J. Koundé	FC Barcelona
47	E.	Camavinga		E. Camavinga	Real Madrid
48	R.	Cherki		R. Cherki	Olympique Lyonnais
49	Vitinha			Vitinha	Sporting Braga
50	Vinícius	Jr.		Vinícius Jr.	Real Madrid
51	E.	Smith	Rowe	E. Smith Rowe	Arsenal
52	Rafael	Leão		Rafael Leão	Milan
53	Pedri			Pedri	FC Barcelona
54	W.	Gnonto		W. Gnonto	Leeds United
55	R.	Lewis		R. Lewis	Manchester City
56	Danilo			Danilo	Nottingham Forest
57	P.	Foden		P. Foden	Manchester City
58	Y.	Moukoko		Y. Moukoko	Borussia Dortmund
59	Antony			Antony	Manchester United

In [39]:
```
#DF["Team & Contract"]
```

In [40]:
```
#DFF["Team & Contract"]
```

In [41]:
```
#list_DFF_Team_Contract = DFF["Team & Contract"].tolist()
#list_DFF_Team_Contract
```

In [41]:

In [41]:

In [42]:
```
#https://stackoverflow.com/questions/30953299/pandas-if-row-in-column-a-contains-x-write-y-to-row-in-column-b
aa= DF["Team & Contract"].str.split("~ ", expand=True)
##aa
```

In [43]:
```
#https://www.datasciencemadesimple.com/extract-last-n-characters-from-right-of-the-column-in-pandas-python/
aa['Start_Contract']=aa[0].str[-5:]
#aa
```

In [44]:
```
aa['End_Contract']=aa[1]
#https://sparkbyexamples.com/pandas/pandas-replace-nan-with-blank-empty-string/
aa['End_Contract']=aa['End_Contract'].fillna(' ')
#aa
```

In [45]:
```
# https://pbpython.com/currency-cleanup.html
#https://www.orsurety.com/blog/is-it-m-for-thousand-and-mm-for-million-or-k-for-thousand-and-m-for-million-im-a
```

In [46]:
```
# https://www.scaler.com/topics/remove-special-characters-from-string-python/
# String with special characters
special_string =pd.Series(DF['Value'])
```

In [46]:

In [46]:

In [47]:
```
# https://www.scaler.com/topics/remove-special-characters-from-string-python/
# String with special characters
correct_string =pd.Series(DF['Wage'])
```

In [47]:

In [47]:

In [48]:
```
listZZ = []
s = ['K','M']
ZZ = special_string
for s in ZZ:
   if 'M' in s:
     listZZ.append(1)
   elif  'K' in s:
     listZZ.append(0.001)
   else:
    listZZ.append(0.000001)
  # https://www.easytweaks.com/pandas-multiply-two-multiple-columns/
```

In [49]:
```
#listZZ
```

In [50]:
```
df_Value = DataFrame (listZZ,columns=['Value_Million'])
#df_Value
```

In [51]:
```
listYY = []
s = ['K','M']
YY = correct_string
for s in YY:
   if 'M' in s:
     listYY.append(1)
   elif  'K' in s:
     listYY.append(0.001)
   else:
    listYY.append(0.000001)
```

In [52]:
```
#listYY
```

In [53]:
```
df_Wage = DataFrame (listYY,columns=['Wage_Million'])
#df_Wage
```

In [54]:
```
ZO= pd.Series(DF['Value'])
special_characters=['€','M','K']
for i in special_characters:
  ZO=ZO.str.replace(i," ")
```

In [55]:
```
#ZO.tolist()
```

In [55]:

In [56]:
```
df_Value_amount = DataFrame (ZO.tolist(),columns=['Value_In-Float'])
#df_Value_amount
```

In [56]:

In [57]:
```
ZA=pd.Series(DF['Wage'])
special_characters=['€','M','K']
for i in special_characters:
  ZA=ZA.str.replace(i," ")
  ZA.tolist()
#ZA
```

In [58]:
```
df_Wage_amount = DataFrame ( ZA.tolist(),columns=['Wage_In_Float'])
#df_Wage_amount
```

In [59]:
```
# https://media.licdn.com/dms/document/C4D1FAQEJE8Zxs2dgrg/feedshare-document-pdf-analyzed/0/1671561850581?e=16
```

In [60]:
```
#frames = [df1, df2, df3]
#result = pd.concat(frames)
DF11 = pd.DataFrame(A['Full Name'])
DF12 = DF[['Age','OVA','POT']]
DF13 = A['Team']
DF14 = aa[['Start_Contract','End_Contract']]
DF15 =  DF['Total']
########################################
#https://www.geeksforgeeks.org/how-to-combine-two-dataframe-in-python-pandas/
frames = [DF11,DF12,DF13,DF14]
result = pd.concat(frames, axis=1, join='inner')
result
```

Out[60]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract
0	F. Balogun	20	71	79	Reims Jun	Loan	
1	E. Fernández	21	80	87	Benfica	2022	2027
2	Balde	18	76	87	FC Barcelona	2021	2024
3	A. Ounahi	22	73	83	Angers SCO	2021	2026
4	K. Mitoma	25	75	78	Brighton Hove Albion	2021	2025
5	M. Mudryk	21	75	85	Chelsea	2023	2031
6	D. Núñez	23	82	89	Liverpool	2022	2028
7	B. White	24	80	84	Arsenal	2021	2026
8	João Cancelo	28	87	87	Manchester City	2019	2027
9	K. Miura	54	56	56	Yokohama	2005	2022
10	N. Madueke	20	77	86	Chelsea	2023	2030
11	M. Ødegaard	23	85	89	Arsenal	2021	2026
12	H. Souttar	23	73	77	Stoke City	2016	2025
13	W. Weghorst	29	79	79	Manchester United Jun	Loan	
14	K. Kvaratskhelia	21	79	88	Napoli	2022	2027
15	G. Rutter	20	75	85	Leeds United	2023	2028
16	Jorginho	30	84	84	Chelsea	2018	2023
17	Pedro Porro	22	82	88	Sporting CP	2022	2025
18	W. Saliba	21	82	89	Arsenal	2019	2024
19	B. Saka	20	84	90	Arsenal	2018	2024
20	Gavi	17	80	89	FC Barcelona	2020	2026
21	J. Kiwior	22	72	81	Arsenal	2023	2028
22	M. Caicedo	20	75	83	Brighton Hove Albion	2021	2025
23	M. Sabitzer	28	81	81	FC Bayern München	2021	2025
24	L. Trossard	27	80	80	Arsenal	2023	2027
25	F. Miretti	18	74	88	Juventus	2021	2026
26	M. Rashford	24	83	86	Manchester United	2014	2024
27	J. Araujo	20	70	80	LA Galaxy	2019	2025
28	O. Zinchenko	25	80	82	Arsenal	2022	2027
29	K. Mbappé	23	91	95	Paris Saint Germain	2018	2024
30	J. Sancho	22	83	87	Manchester United	2021	2026
31	Gabriel Martinelli	21	80	88	Arsenal	2019	2024
32	N. Zaniolo	22	80	85	Roma	2018	2024
33	Pablo Torre	19	69	86	FC Barcelona	2022	2026
34	T. Tomiyasu	23	79	84	Arsenal	2021	2026
35	Ansu Fati	19	79	89	FC Barcelona	2018	2027
36	R. Araujo	23	83	89	FC Barcelona	2018	2026
37	C. Chukwuemeka	18	64	84	Chelsea	2022	2028

38	Alejandro Garnacho	18	66	85	Manchester United	2021	2023
39	D. Doué	17	70	86	Rennes	2022	2025
40	Adama Traoré	26	77	77	Wolverhampton Wanderers	2018	2023
41	João Félix	22	83	89	Chelsea Jun	Loan	
42	Fresneda	17	67	81	Real Valladolid	2021	2025
43	A. Mac Allister	23	78	83	Brighton Hove Albion	2019	2025
44	L. Messi	35	91	91	Paris Saint Germain	2021	2023
45	V. Osimhen	23	83	89	Napoli	2020	2025
46	J. Koundé	23	84	89	FC Barcelona	2022	2027
47	E. Camavinga	19	79	89	Real Madrid	2021	2027
48	R. Cherki	18	73	88	Olympique Lyonnais	2019	2023
49	Vitinha	22	76	84	Sporting Braga	2020	2027
50	Vinícius Jr.	21	86	92	Real Madrid	2018	2025
51	E. Smith Rowe	21	80	87	Arsenal	2017	2026
52	Rafael Leão	23	84	90	Milan	2019	2024
53	Pedri	19	85	92	FC Barcelona	2020	2026
54	W. Gnonto	18	72	86	Leeds United	2022	2027
55	R. Lewis	17	65	85	Manchester City	2022	2024
56	Danilo	21	76	84	Nottingham Forest	2023	2029
57	P. Foden	22	85	92	Manchester City	2016	2027
58	Y. Moukoko	17	75	89	Borussia Dortmund	2020	2026
59	Antony	22	82	88	Manchester United	2022	2027

In [61]:
```
result.dtypes
```

Out[61]:
```
Full Name          object
Age                 int64
OVA                 int64
POT                 int64
Team               object
Start_Contract     object
End_Contract       object
dtype: object
```

In [62]:
```
#https://www.statology.org/pandas-multiply-two-columns/
vl = df_Value_amount['Value_In-Float'].astype(float)*df_Value['Value_Million'].astype(float)
result['Value_In_€Million']=pd.DataFrame(vl)
wg = df_Wage_amount['Wage_In_Float'].astype(float)*df_Wage['Wage_Million'].astype(float)
result['Wage_In_€Million']=pd.DataFrame(wg)
result['Total']= DF['Total']
# 13,14,15,16,17,18
```

In [63]:
```
result
```

Out[63]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	F. Balogun	20	71	79	Reims Jun	Loan		4.1	0.0350	1657
1	E. Fernández	21	80	87	Benfica	2022	2027	42.5	0.0130	2169
2	Balde	18	76	87	FC Barcelona	2021	2024	15.0	0.0270	1819
3	A. Ounahi	22	73	83	Angers SCO	2021	2026	7.0	0.0170	1876
4	K. Mitoma	25	75	78	Brighton Hove Albion	2021	2025	7.5	0.0510	1943
5	M. Mudryk	21	75	85	Chelsea	2023	2031	12.5	0.0570	1829
6	D. Núñez	23	82	89	Liverpool	2022	2028	61.5	0.1200	2075
7	B. White	24	80	84	Arsenal	2021	2026	28.0	0.0690	1791
8	João Cancelo	28	87	87	Manchester City	2019	2027	72.0	0.2100	2252
9	K. Miura	54	56	56	Yokohama	2005	2022	0.0	0.0007	1267
10	N. Madueke	20	77	86	Chelsea	2023	2030	23.0	0.0720	1896
11	M. Ødegaard	23	85	89	Arsenal	2021	2026	77.5	0.1150	2125
12	H. Souttar	23	73	77	Stoke City	2016	2025	3.9	0.0180	1508
13	W. Weghorst	29	79	79	Manchester United Jun	Loan		17.5	0.0590	1900
14	K. Kvaratskhelia	21	79	88	Napoli	2022	2027	39.5	0.0460	1949
15	G. Rutter	20	75	85	Leeds United	2023	2028	12.5	0.0300	1752
16	Jorginho	30	84	84	Chelsea	2018	2023	40.5	0.1500	2120

17	Pedro Porro	22	82	88	Sporting CP	2022	2025	46.0	0.0160	2157
18	W. Saliba	21	82	89	Arsenal	2019	2024	57.5	0.0670	1868
19	B. Saka	20	84	90	Arsenal	2018	2024	65.5	0.0900	2143
20	Gavi	17	80	89	FC Barcelona	2020	2026	43.5	0.0180	2035
21	J. Kiwior	22	72	81	Arsenal	2023	2028	4.9	0.0370	1696
22	M. Caicedo	20	75	83	Brighton Hove Albion	2021	2025	11.5	0.0350	1955
23	M. Sabitzer	28	81	81	FC Bayern München	2021	2025	25.5	0.0720	2179
24	L. Trossard	27	80	80	Arsenal	2023	2027	22.5	0.0880	1975
25	F. Miretti	18	74	88	Juventus	2021	2026	10.0	0.0240	2011
26	M. Rashford	24	83	86	Manchester United	2014	2024	49.5	0.1400	2101
27	J. Araujo	20	70	80	LA Galaxy	2019	2025	3.3	0.0030	1727
28	O. Zinchenko	25	80	82	Arsenal	2022	2027	24.5	0.0770	2154
29	K. Mbappé	23	91	95	Paris Saint Germain	2018	2024	190.5	0.2300	2181
30	J. Sancho	22	83	87	Manchester United	2021	2026	53.0	0.1400	1898
31	Gabriel Martinelli	21	80	88	Arsenal	2019	2024	47.5	0.0650	1982
32	N. Zaniolo	22	80	85	Roma	2018	2024	33.0	0.0660	2089
33	Pablo Torre	19	69	86	FC Barcelona	2022	2026	3.6	0.0270	1779
34	T. Tomiyasu	23	79	84	Arsenal	2021	2026	24.5	0.0650	1837
35	Ansu Fati	19	79	89	FC Barcelona	2018	2027	38.0	0.0840	1913
36	R. Araujo	23	83	89	FC Barcelona	2018	2026	52.0	0.1350	1910
37	C. Chukwuemeka	18	64	84	Chelsea	2022	2028	1.6	0.0070	1687
38	Alejandro Garnacho	18	66	85	Manchester United	2021	2023	2.2	0.0110	1656
39	D. Doué	17	70	86	Rennes	2022	2025	4.0	0.0030	1699
40	Adama Traoré	26	77	77	Wolverhampton Wanderers	2018	2023	11.5	0.0770	1920
41	João Félix	22	83	89	Chelsea Jun	Loan		59.0	0.0710	2102
42	Fresneda	17	67	81	Real Valladolid	2021	2025	2.2	0.0005	1677
43	A. Mac Allister	23	78	83	Brighton Hove Albion	2019	2025	21.0	0.0560	2129
44	L. Messi	35	91	91	Paris Saint Germain	2021	2023	54.0	0.1950	2190
45	V. Osimhen	23	83	89	Napoli	2020	2025	58.0	0.0720	1999
46	J. Koundé	23	84	89	FC Barcelona	2022	2027	57.5	0.1500	1926
47	E. Camavinga	19	79	89	Real Madrid	2021	2027	38.0	0.0780	2125
48	R. Cherki	18	73	88	Olympique Lyonnais	2019	2023	7.0	0.0150	1719
49	Vitinha	22	76	84	Sporting Braga	2020	2027	17.0	0.0140	1872
50	Vinícius Jr.	21	86	92	Real Madrid	2018	2025	109.0	0.2000	1985
51	E. Smith Rowe	21	80	87	Arsenal	2017	2026	43.0	0.0650	1830
52	Rafael Leão	23	84	90	Milan	2019	2024	66.5	0.0900	1971
53	Pedri	19	85	92	FC Barcelona	2020	2026	104.5	0.1150	2112
54	W. Gnonto	18	72	86	Leeds United	2022	2027	5.5	0.0090	1778
55	R. Lewis	17	65	85	Manchester City	2022	2024	1.9	0.0010	1622
56	Danilo	21	76	84	Nottingham Forest	2023	2029	16.0	0.0360	1907
57	P. Foden	22	85	92	Manchester City	2016	2027	109.5	0.1800	2087
58	Y. Moukoko	17	75	89	Borussia Dortmund	2020	2026	13.0	0.0080	1828
59	Antony	22	82	88	Manchester United	2022	2027	49.0	0.1300	2046

In [64]:
```
#result =  result.set_index('Full Name', append=True)
```

In [65]:
```
#result
```

In [66]:
```
result.dtypes
```

Out[66]:
```
Full Name             object
Age                    int64
OVA                    int64
POT                    int64
Team                  object
Start_Contract        object
End_Contract          object
Value_In_€Million    float64
Wage_In_€Million     float64
Total                  int64
dtype: object
```

In [67]:
```
# https://www.statology.org/pandas-select-columns-containing-string/ # df.filter(regex='string1|string2|string3
```

In [68]:
```
# filtering the rows contains specific word.
dfLoan = DF[DF['Team & Contract'].str.contains('Loan')]
dfLoan
```

Out[68]:

	Name	Age	OVA	POT	Team & Contract	Value	Wage	Total
0	F. Balogun ST	20	71	79	Reims Jun 30, 2023 On Loan	€4.1M	€35K	1657
13	W. Weghorst ST	29	79	79	Manchester United Jun 30, 2023 On Loan	€17.5M	€59K	1900
41	João Félix CF ST	22	83	89	Chelsea Jun 30, 2023 On Loan	€59M	€71K	2102

In [69]:
```
stringL = "Beşiktaş May 31, 2023 On Loan"
print(len(stringL))
print(len('May 31, 2023 On Loan'))
```

```
29
20
```

In [70]:
```
dfLoan['Loan_End']=dfLoan['Team & Contract'].str[-20:]
dfLoan['Loan_End'] = dfLoan['Loan_End'].str[:-8]
dfLoan
#dfLoan['Loan_End']
```

```
<ipython-input-70-e41db614afa8>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret
urning-a-view-versus-a-copy
  dfLoan['Loan_End']=dfLoan['Team & Contract'].str[-20:]
<ipython-input-70-e41db614afa8>:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#ret
urning-a-view-versus-a-copy
  dfLoan['Loan_End'] = dfLoan['Loan_End'].str[:-8]
```

Out[70]:

	Name	Age	OVA	POT	Team & Contract	Value	Wage	Total	Loan_End
0	F. Balogun ST	20	71	79	Reims Jun 30, 2023 On Loan	€4.1M	€35K	1657	Jun 30, 2023
13	W. Weghorst ST	29	79	79	Manchester United Jun 30, 2023 On Loan	€17.5M	€59K	1900	Jun 30, 2023
41	João Félix CF ST	22	83	89	Chelsea Jun 30, 2023 On Loan	€59M	€71K	2102	Jun 30, 2023

In [71]:
```
#dfLoan['LoanEnd']=dfLoan['Loan_End'].str[-8:]
#dfLoan
```

In [72]:
```
dfresultLoan = result[result['Start_Contract'].str.contains('Loan')]
dfresultLoan
```

Out[72]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	F. Balogun	20	71	79	Reims Jun	Loan		4.1	0.035	1657
13	W. Weghorst	29	79	79	Manchester United Jun	Loan		17.5	0.059	1900
41	João Félix	22	83	89	Chelsea Jun	Loan		59.0	0.071	2102

In [73]:
```
LL   = dfresultLoan['Team'].str[:-7]
Teams = pd.DataFrame(LL)
Teams
```

Out[73]:

	Team
0	Reims
13	Manchester United
41	Chelsea

In [74]:
```
dfresultLoan['Team']=Teams
```

In [74]:
```
dfresultLoan['Team']=Teams
dfresultLoan
```

```
<ipython-input-74-40598db3f499>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  dfresultLoan['Team']=Teams
```

Out[74]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	F. Balogun	20	71	79	Reims	Loan		4.1	0.035	1657
13	W. Weghorst	29	79	79	Manchester United	Loan		17.5	0.059	1900
41	João Félix	22	83	89	Chelsea	Loan		59.0	0.071	2102

In [75]:
```
#DFf11 = dfresultLoan['Full Name','Age','OVA','POT','Team']
DFf12 = dfLoan['Loan_End']
#DFf13 =  dfresultLoan['Total']
########################################
#https://www.geeksforgeeks.org/how-to-combine-two-dataframe-in-python-pandas/
frameS = [dfresultLoan,DFf12]
On_loan = pd.concat(frameS, axis=1, join='inner')
On_loan
```

Out[75]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total	Loan_End
0	F. Balogun	20	71	79	Reims	Loan		4.1	0.035	1657	Jun 30, 2023
13	W. Weghorst	29	79	79	Manchester United	Loan		17.5	0.059	1900	Jun 30, 2023
41	João Félix	22	83	89	Chelsea	Loan		59.0	0.071	2102	Jun 30, 2023

In [76]:
```
On_loan = On_loan[['Full Name','Age','OVA','POT','Team','Loan_End','Value_In_€Million', 'Wage_In_€Million',
#On_loan = On_loan[['Age','OVA','POT','Team','Loan_End','Value_In_€Million',    'Wage_In_€Million',     'Total'
```

In [77]:
```
#On_loan
On_loan = On_loan.style.set_properties(**{'text-align': 'center'})
#On_loan = On_loan.set_properties(**{'text-align': 'left'})
#display(On_loan)
#On_loan
```

In [78]:
```
framesresultLoan = [dfresultLoan,A]
resultLoan = pd.concat(framesresultLoan, axis=1, join='inner')
resultLoan
```

Out[78]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total	0	1	Col2	C
0	F. Balogun	20	71	79	Reims	Loan		4.1	0.035	1657	F.	Balogun		
13	W. Weghorst	29	79	79	Manchester United	Loan		17.5	0.059	1900	W.	Weghorst		
41	João Félix	22	83	89	Chelsea	Loan		59.0	0.071	2102	João	Félix		

In [79]:
```
dfFree = DF[DF['Team & Contract'].str.contains('Free')]
dfFree
```

Out[79]:

	Name	Age	OVA	POT	Team & Contract	Value	Wage	Total

In [80]:
```
dfresultFree = result[result['Start_Contract'].str.contains('Free')]
dfresultFree
```

Out[80]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total

In [81]:
```
Are_free = dfresultFree[['Full Name','Age','OVA','POT','Team','Total']]
Are_free
Are_free.rename(columns={'Team': 'From_country'},
          inplace=True, errors='raise')
Are_free = Are_free.style.set_properties(**{'text-align': 'center'})
Are_free
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:5039: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  return super().rename(
```

Out[81]:

Full Name	Age	OVA	POT	From_country	Total

In [82]:
```
#Are_free
drive.mount('/content/drive', force_remount=True)
path = '/content/drive/MyDrive/Are_free.xlsx'
Are_free.to_excel(path, index = None)
```

```
Mounted at /content/drive
```

In [83]:
```
#Are_free.rename(columns={'Team': 'From_country'},
          #inplace=True, errors='raise')
```

In [84]:
```
On_contract = result[~result['Start_Contract'].str.contains('Loan|Free')]
On_contract
On_contract = On_contract.style.set_properties(**{'text-align': 'center'})
On_contract
```

Out[84]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
1	E. Fernández	21	80	87	Benfica	2022	2027	42.500000	0.013000	2169
2	Balde	18	76	87	FC Barcelona	2021	2024	15.000000	0.027000	1819
3	A. Ounahi	22	73	83	Angers SCO	2021	2026	7.000000	0.017000	1876
4	K. Mitoma	25	75	78	Brighton Hove Albion	2021	2025	7.500000	0.051000	1943
5	M. Mudryk	21	75	85	Chelsea	2023	2031	12.500000	0.057000	1829
6	D. Núñez	23	82	89	Liverpool	2022	2028	61.500000	0.120000	2075
7	B. White	24	80	84	Arsenal	2021	2026	28.000000	0.069000	1791
8	João Cancelo	28	87	87	Manchester City	2019	2027	72.000000	0.210000	2252
9	K. Miura	54	56	56	Yokohama	2005	2022	0.000000	0.000700	1267
10	N. Madueke	20	77	86	Chelsea	2023	2030	23.000000	0.072000	1896
11	M. Ødegaard	23	85	89	Arsenal	2021	2026	77.500000	0.115000	2125
12	H. Souttar	23	73	77	Stoke City	2016	2025	3.900000	0.018000	1508
14	K. Kvaratskhelia	21	79	88	Napoli	2022	2027	39.500000	0.046000	1949
15	G. Rutter	20	75	85	Leeds United	2023	2028	12.500000	0.030000	1752
16	Jorginho	30	84	84	Chelsea	2018	2023	40.500000	0.150000	2120
17	Pedro Porro	22	82	88	Sporting CP	2022	2025	46.000000	0.016000	2157
18	W. Saliba	21	82	89	Arsenal	2019	2024	57.500000	0.067000	1868
19	B. Saka	20	84	90	Arsenal	2018	2024	65.500000	0.090000	2143
20	Gavi	17	80	89	FC Barcelona	2020	2026	43.500000	0.018000	2035
21	J. Kiwior	22	72	81	Arsenal	2023	2028	4.900000	0.037000	1696
22	M. Caicedo	20	75	83	Brighton Hove Albion	2021	2025	11.500000	0.035000	1955
23	M. Sabitzer	28	81	81	FC Bayern München	2021	2025	25.500000	0.072000	2179
24	L. Trossard	27	80	80	Arsenal	2023	2027	22.500000	0.088000	1975
25	F. Miretti	18	74	88	Juventus	2021	2026	10.000000	0.024000	2011
26	M. Rashford	24	83	86	Manchester United	2014	2024	49.500000	0.140000	2101
27	J. Araujo	20	70	80	LA Galaxy	2019	2025	3.300000	0.003000	1727
28	O. Zinchenko	25	80	82	Arsenal	2022	2027	24.500000	0.077000	2154
29	K. Mbappé	23	91	95	Paris Saint Germain	2018	2024	190.500000	0.230000	2181
30	J. Sancho	22	83	87	Manchester United	2021	2026	53.000000	0.140000	1898
31	Gabriel Martinelli	21	80	88	Arsenal	2019	2024	47.500000	0.065000	1982
32	N. Zaniolo	22	80	85	Roma	2018	2024	33.000000	0.066000	2089
33	Pablo Torre	19	69	86	FC Barcelona	2022	2026	3.600000	0.027000	1779
34	T. Tomiyasu	23	79	84	Arsenal	2021	2026	24.500000	0.065000	1837
35	Ansu Fati	19	79	89	FC Barcelona	2018	2027	38.000000	0.084000	1913
36	R. Araujo	23	83	89	FC Barcelona	2018	2026	52.000000	0.135000	1910
37	C. Chukwuemeka	18	64	84	Chelsea	2022	2028	1.600000	0.007000	1687
38	Alejandro Garnacho	18	66	85	Manchester United	2021	2023	2.200000	0.011000	1656
39	D. Doué	17	70	86	Rennes	2022	2025	4.000000	0.003000	1699

Wolverhampton

40	Adama Traoré	26	77	77	Wolverhampton Wanderers	2018	2023	11.500000	0.077000	1920
42	Fresneda	17	67	81	Real Valladolid	2021	2025	2.200000	0.000500	1677
43	A. Mac Allister	23	78	83	Brighton Hove Albion	2019	2025	21.000000	0.056000	2129
44	L. Messi	35	91	91	Paris Saint Germain	2021	2023	54.000000	0.195000	2190
45	V. Osimhen	23	83	89	Napoli	2020	2025	58.000000	0.072000	1999
46	J. Koundé	23	84	89	FC Barcelona	2022	2027	57.500000	0.150000	1926
47	E. Camavinga	19	79	89	Real Madrid	2021	2027	38.000000	0.078000	2125
48	R. Cherki	18	73	88	Olympique Lyonnais	2019	2023	7.000000	0.015000	1719
49	Vitinha	22	76	84	Sporting Braga	2020	2027	17.000000	0.014000	1872
50	Vinícius Jr.	21	86	92	Real Madrid	2018	2025	109.000000	0.200000	1985
51	E. Smith Rowe	21	80	87	Arsenal	2017	2026	43.000000	0.065000	1830
52	Rafael Leão	23	84	90	Milan	2019	2024	66.500000	0.090000	1971
53	Pedri	19	85	92	FC Barcelona	2020	2026	104.500000	0.115000	2112
54	W. Gnonto	18	72	86	Leeds United	2022	2027	5.500000	0.009000	1778
55	R. Lewis	17	65	85	Manchester City	2022	2024	1.900000	0.001000	1622
56	Danilo	21	76	84	Nottingham Forest	2023	2029	16.000000	0.036000	1907
57	P. Foden	22	85	92	Manchester City	2016	2027	109.500000	0.180000	2087
58	Y. Moukoko	17	75	89	Borussia Dortmund	2020	2026	13.000000	0.008000	1828
59	Antony	22	82	88	Manchester United	2022	2027	49.000000	0.130000	2046

In [84]:

In [85]:
```
# Google Colab: Ways To Save Pandas Dataframe Data
#from google.colab import files and drive
from google.colab import files
from google.colab import drive
!pip install openpyxl
drive.mount('/content/drive', force_remount=True)
path = '/content/drive/MyDrive/On_contract.xlsx'
On_contract.to_excel(path, index = None)
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: openpyxl in /usr/local/lib/python3.8/dist-packages (3.0.10)
Requirement already satisfied: et-xmlfile in /usr/local/lib/python3.8/dist-packages (from openpyxl) (1.1.0)
Mounted at /content/drive
```

In [85]:

In [86]:
```
df_contract_xlsl= pd.read_excel('/content/drive/MyDrive/On_contract.xlsx')
df_contract_xlsl
```

Out[86]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	E. Fernández	21	80	87	Benfica	2022	2027	42.5	0.0130	2169
1	Balde	18	76	87	FC Barcelona	2021	2024	15.0	0.0270	1819
2	A. Ounahi	22	73	83	Angers SCO	2021	2026	7.0	0.0170	1876
3	K. Mitoma	25	75	78	Brighton Hove Albion	2021	2025	7.5	0.0510	1943
4	M. Mudryk	21	75	85	Chelsea	2023	2031	12.5	0.0570	1829
5	D. Núñez	23	82	89	Liverpool	2022	2028	61.5	0.1200	2075
6	B. White	24	80	84	Arsenal	2021	2026	28.0	0.0690	1791
7	João Cancelo	28	87	87	Manchester City	2019	2027	72.0	0.2100	2252
8	K. Miura	54	56	56	Yokohama	2005	2022	0.0	0.0007	1267
9	N. Madueke	20	77	86	Chelsea	2023	2030	23.0	0.0720	1896
10	M. Ødegaard	23	85	89	Arsenal	2021	2026	77.5	0.1150	2125
11	H. Souttar	23	73	77	Stoke City	2016	2025	3.9	0.0180	1508
12	K. Kvaratskhelia	21	79	88	Napoli	2022	2027	39.5	0.0460	1949
13	G. Rutter	20	75	85	Leeds United	2023	2028	12.5	0.0300	1752
14	Jorginho	30	84	84	Chelsea	2018	2023	40.5	0.1500	2120
15	Pedro Porro	22	82	88	Sporting CP	2022	2025	46.0	0.0160	2157
16	W. Saliba	21	82	89	Arsenal	2019	2024	57.5	0.0670	1868
17	B. Saka	20	84	90	Arsenal	2018	2024	65.5	0.0900	2143
18	Gavi	17	80	89	FC Barcelona	2020	2026	43.5	0.0180	2035
19	J. Kiwior	22	72	81	Arsenal	2023	2028	4.9	0.0370	1696
20	M. Caicedo	20	75	83	Brighton Hove Albion	2021	2025	11.5	0.0350	1955

21	M. Sabitzer	28	81	81	FC Bayern München	2021	2025	25.5	0.0720	2179
22	L. Trossard	27	80	80	Arsenal	2023	2027	22.5	0.0880	1975
23	F. Miretti	18	74	88	Juventus	2021	2026	10.0	0.0240	2011
24	M. Rashford	24	83	86	Manchester United	2014	2024	49.5	0.1400	2101
25	J. Araujo	20	70	80	LA Galaxy	2019	2025	3.3	0.0030	1727
26	O. Zinchenko	25	80	82	Arsenal	2022	2027	24.5	0.0770	2154
27	K. Mbappé	23	91	95	Paris Saint Germain	2018	2024	190.5	0.2300	2181
28	J. Sancho	22	83	87	Manchester United	2021	2026	53.0	0.1400	1898
29	Gabriel Martinelli	21	80	88	Arsenal	2019	2024	47.5	0.0650	1982
30	N. Zaniolo	22	80	85	Roma	2018	2024	33.0	0.0660	2089
31	Pablo Torre	19	69	86	FC Barcelona	2022	2026	3.6	0.0270	1779
32	T. Tomiyasu	23	79	84	Arsenal	2021	2026	24.5	0.0650	1837
33	Ansu Fati	19	79	89	FC Barcelona	2018	2027	38.0	0.0840	1913
34	R. Araujo	23	83	89	FC Barcelona	2018	2026	52.0	0.1350	1910
35	C. Chukwuemeka	18	64	84	Chelsea	2022	2028	1.6	0.0070	1687
36	Alejandro Garnacho	18	66	85	Manchester United	2021	2023	2.2	0.0110	1656
37	D. Doué	17	70	86	Rennes	2022	2025	4.0	0.0030	1699
38	Adama Traoré	26	77	77	Wolverhampton Wanderers	2018	2023	11.5	0.0770	1920
39	Fresneda	17	67	81	Real Valladolid	2021	2025	2.2	0.0005	1677
40	A. Mac Allister	23	78	83	Brighton Hove Albion	2019	2025	21.0	0.0560	2129
41	L. Messi	35	91	91	Paris Saint Germain	2021	2023	54.0	0.1950	2190
42	V. Osimhen	23	83	89	Napoli	2020	2025	58.0	0.0720	1999
43	J. Koundé	23	84	89	FC Barcelona	2022	2027	57.5	0.1500	1926
44	E. Camavinga	19	79	89	Real Madrid	2021	2027	38.0	0.0780	2125
45	R. Cherki	18	73	88	Olympique Lyonnais	2019	2023	7.0	0.0150	1719
46	Vitinha	22	76	84	Sporting Braga	2020	2027	17.0	0.0140	1872
47	Vinícius Jr.	21	86	92	Real Madrid	2018	2025	109.0	0.2000	1985
48	E. Smith Rowe	21	80	87	Arsenal	2017	2026	43.0	0.0650	1830
49	Rafael Leão	23	84	90	Milan	2019	2024	66.5	0.0900	1971
50	Pedri	19	85	92	FC Barcelona	2020	2026	104.5	0.1150	2112
51	W. Gnonto	18	72	86	Leeds United	2022	2027	5.5	0.0090	1778
52	R. Lewis	17	65	85	Manchester City	2022	2024	1.9	0.0010	1622
53	Danilo	21	76	84	Nottingham Forest	2023	2029	16.0	0.0360	1907
54	P. Foden	22	85	92	Manchester City	2016	2027	109.5	0.1800	2087
55	Y. Moukoko	17	75	89	Borussia Dortmund	2020	2026	13.0	0.0080	1828
56	Antony	22	82	88	Manchester United	2022	2027	49.0	0.1300	2046

In [87]:
```
drive.mount('/content/drive', force_remount=True)
path = '/content/drive/MyDrive/On_contract.html'
On_contract.to_html(path)
```

```
Mounted at /content/drive
```

In [87]:

In [88]:
```
#Connecting using SQLAlchemy
#Use the copied path in below code to connect.
```

In [89]:
```
#https://stephenallwright.com/pandas-to_sql/
import sqlalchemy
from sqlalchemy import create_engine
from sqlalchemy.engine import cursor
import sqlite3 as sql
#my_conn =create_engine("sqlite:////content/drive/MyDrive/Data_Engineering/example.db")
#my_conn=create_engine("sqlite:////content/drive/MyDrive/example.db")
my_conn=create_engine("sqlite:////content/drive/MyDrive/example.db")
#connection_obj = sql.connect("/content/drive/MyDrive/example.db")
```

In [90]:
```
# cursor object
#cursor_obj = connection_obj.cursor()
```

In [91]:
```
#my_conn
#cursor_obj = my_conn.cursor()
```

In [92]:
```
#pd.DataFrame(Are_free)
```

In [93]:
```
#Database connection is ready, we will check by using this code to list the tables available in our sample data
```

In [94]:
```
r_set=('''select name from sqlite_master
       where type = 'table' ''',my_conn)
for row in r_set:
   print(row)
```

```
select name from sqlite_master
       where type = 'table'
Engine(sqlite:////content/drive/MyDrive/example.db)
```

In [95]:
```
#pd.read_sql('SELECT * FROM Player_Under_Contract',my_conn)
```

In [96]:
```
import pandas_datareader.data as web
#df = web.DataReader("On_contract",data_source = 'url')
```

In [97]:
```
# Write data into the table in sqllite database
df_contract_xlsl.to_sql('df_contract_xlsl', my_conn,if_exists="replace")
```

In [97]:
```
```

In [98]:
```
#Each column in an SQLite 3 database is assigned one of the following type affinities:
#-TEXT
#-NUMERIC
#-INTEGER
#-REAL
#-BLOB
```

In [99]:
```
# "Full Name",  "Age",  "OVA",  "POT",  "Team", "Start_Contract",       "End_Contract", "Value_In_€Million",
```

In [100…
```
#conn = sql("sqlite:////content/drive/MyDrive/Data_Engineering/example.db")
```

In [101…
```
table1 = """CREATE TABLE IF NOT EXISTS Player_with_Contract (
       Name TEXT NOT NULL PRIMARY KEY,
  Age INTEGER NOT NULL,
       OVA INTEGER,
       POT INTEGER,
  Team TEXT,
  Start_Contract DATE  NOT NULl,
  End_Contract   DATE  NOT NULL,
  Value_In_€Million REAL,
  Wage_In_€Million REAL,
  Total INTEGER
);"""
#cursor_obj.execute.read_sql('SELECT * FROM Player_with_Contract',my_conn)
#cursor_obj.execute(table1)
my_conn.execute(table1)
#pd.read_sql(table1,my_conn)
```

Out[101]:
```
<sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f0fd01ac040>
```

In [102…
```
table4 = """CREATE TABLE  IF NOT EXISTS Player_HAVE_Contract (
       Name TEXT NOT NULL ,
  Age INTEGER NOT NULL,
       OVA INTEGER NOT NULL,
       POT INTEGER NOT NULL ,
  Team TEXT NOT NULL,
  Start_Contract INTEGER  NOT NULL,
  End_Contract   INTEGER  NOT NULL,
  Value_In_€Million REAL NOT NULL,
  Wage_In_€Million REAL NOT NULL,
  Total INTEGER NOT NULL
);"""
my_conn.execute(table4)
```

Out[102]:
```
<sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f0fd0146610>
```

In [102…
```
```

In [103…
```
print(my_conn.execute('SELECT * FROM Player_with_Contract'))
pd.read_sql('SELECT * FROM Player_with_Contract',my_conn)
```

```
<sqlalchemy.engine.cursor.LegacyCursorResult object at 0x7f0fd0146c70>
```

Out[103]:

Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total

In [104…
```
pd.read_sql('SELECT * FROM Player_UNDER_Contract',my_conn)
```

Out[104]:

	Full Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	M. Rashford	24	83	86	Manchester United	2014	2024	49.500	0.14000	2097
1	S. Botman	22	81	87	Newcastle United	2022	2027	37.500	0.07900	1749
2	Y. Moukoko	17	75	89	Borussia Dortmund	2020	2023	13.000	0.00800	1828
3	Alejandro Garnacho	18	66	86	Manchester United	2021	2023	2.400	0.01100	1656
4	Andrey Santos	18	70	84	Chelsea	2023	2028	3.700	0.01600	1828
5	J. Bellingham	19	85	91	Borussia Dortmund	2020	2025	81.500	0.04400	2188
6	D. Núñez	23	82	89	Liverpool	2022	2028	61.500	0.12000	2075
7	A. Mac Allister	23	78	83	Brighton Hove Albion	2019	2025	21.000	0.05600	2122
8	A. Schjelderup	18	71	86	Nordsjælland	2020	2024	4.400	0.00400	1753
9	R. Kolo Muani	23	81	86	Eintracht Frankfurt	2022	2027	39.500	0.03700	1897
10	R. Lewis	17	65	86	Manchester City	2022	2024	2.200	0.00100	1622
11	D. Doué	17	69	86	Rennes	2022	2025	3.500	0.00200	1689
12	D. Kamada	25	83	84	Eintracht Frankfurt	2017	2023	43.500	0.04300	2022
13	M. Mudryk	21	75	85	Shakhtar Donetsk	2018	2026	12.500	0.00075	1823
14	P. Wanner	16	62	85	FC Bayern München	2022	2027	1.200	0.00100	1584
15	J. Frimpong	21	81	87	Bayer Leverkusen	2021	2025	38.500	0.04100	2054
16	Cristiano Ronaldo	37	88	88	Al Nassr	2023	2025	31.000	0.08600	2114
17	A. Griezmann	31	84	84	Atlético Madrid	2022	2026	35.500	0.08800	2262
18	D. Fofana	19	68	83	Chelsea	2023	2024	2.900	0.02100	1691
19	J. Musiala	19	83	92	FC Bayern München	2020	2026	78.500	0.04600	2027
20	Diogo Dalot	23	80	86	Manchester United	2018	2024	31.000	0.09000	2106
21	R. Wilson	16	59	85	Aston Villa	2022	2026	0.900	0.00200	1406
22	A. Ounahi	22	73	83	Angers SCO	2021	2026	7.000	0.01700	1869
23	Bruno Guimarães	24	83	87	Newcastle United	2022	2026	52.000	0.10500	2165
24	T. Livramento	19	75	85	Southampton	2021	2026	11.500	0.02200	1862
25	C. Gallagher	22	78	85	Chelsea	2017	2025	28.500	0.08100	2047
26	J. Gvardiol	20	82	89	RB Leipzig	2021	2027	56.000	0.04400	2044
27	B. Doak	16	61	84	Liverpool	2022	2027	1.000	0.00300	1451
28	K. Mitoma	25	75	78	Brighton Hove Albion	2021	2025	7.500	0.05100	1943
29	L. Shaw	26	81	82	Manchester United	2014	2024	27.500	0.11500	2129
30	M. Acuña	30	85	85	Sevilla	2020	2024	46.500	0.04600	2304
31	T. Tomiyasu	23	79	84	Arsenal	2021	2026	24.500	0.06500	1837
32	N. Molina	24	79	84	Atlético Madrid	2022	2027	26.500	0.04800	2055
33	Fábio Vieira	22	77	87	Arsenal	2022	2027	23.500	0.06200	1852
34	L. Hall	17	61	81	Chelsea	2021	2025	0.875	0.00200	1698
35	M. van de Ven	21	75	85	VfL Wolfsburg	2021	2025	12.000	0.02500	1774
36	M. Simakan	22	79	87	RB Leipzig	2021	2027	37.000	0.04300	1842
37	E. Ferguson	17	63	82	Brighton Hove Albion	2021	2026	1.200	0.00200	1520
38	E. Palacios	23	79	84	Bayer Leverkusen	2020	2025	26.500	0.04800	2089
39	E. Smith Rowe	21	80	87	Arsenal	2017	2026	43.000	0.06500	1830
40	Willian	33	76	76	Fulham	2022	2023	4.700	0.05100	1937
41	H. Son	29	88	88	Tottenham Hotspur	2015	2025	89.500	0.23000	2134
42	A. Dreyer	24	74	80	Midtjylland	2022	2026	6.500	0.02300	1893
43	M. Oršić	29	78	78	Southampton	2023	2027	14.000	0.06300	2111
44	B. Johnson	21	75	85	Nottingham Forest	2019	2026	12.500	0.04300	1830
45	W. Saliba	21	81	89	Arsenal	2019	2024	54.000	0.06200	1855
46	B. White	24	80	86	Arsenal	2021	2026	30.000	0.06900	1769
47	C. Gakpo	23	83	88	Liverpool	2023	2028	55.000	0.13000	2058
48	B. Meijer	19	71	85	Club Brugge	2022	2026	4.300	0.00800	1844

49	Marquinhos	19	73	86	Arsenal	2022	2026	7.000	0.03000	1705
50	I. Fernández	32	80	80	River Plate	2023	2025	16.500	0.02300	2164
51	Joelinton	25	77	80	Newcastle United	2019	2024	14.000	0.07200	2059
52	D. Upamecano	23	83	87	FC Bayern München	2021	2026	47.000	0.06400	1864
53	L. Sobiech	30	70	70	Darmstadt	2021	2022	1.300	0.00800	1518
54	Casemiro	30	89	89	Manchester United	2022	2026	86.000	0.24000	2239

In [105…
```
# Delete the first table 'Player with contract' and keep the second 'Player_under_contract'

('DROP table IF EXISTS Player_with_Contract',my_conn)
```

Out[105]:
```
('DROP table IF EXISTS Player_with_Contract',
 Engine(sqlite:////content/drive/MyDrive/example.db))
```

In [105…

In [106…
```
#pd.read_sql('SELECT * FROM Player_HAVE_Contract',my_conn)
```

In [107…
```
Insert_Into_Table = ('''INSERT INTO Player_HAVE_Contract
(Name,
  Age,
        OVA,
        POT ,
  Team,
  Start_Contract ,
  End_Contract,
  Value_In_€Million ,
  Wage_In_€Million ,
  Total)
VALUES
('Jean-Paul Abalo',26 , 10,5,'Amien',2023,2026,0.25000,0.26000,4564 ),
('Komlan Amewou',32 , 44,8,'Can',2021,2025,0.35000,0.46000,3214),
('Bachirou Salou',35 , 55,4,'Strasbourg',2022,2024,0.25000,0.16000,2239);''')

my_conn.execute(Insert_Into_Table)
```

Out[107]:
```
<sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f0fd0151d90>
```

In [108…
```
Select_Table = pd.read_sql('SELECT * FROM Player_HAVE_Contract',my_conn)
Select_Table
```

Out[108]:

	Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total
0	Jean-Paul Abalo	26	10	5	Amien	2023	2026	0.25	0.26	4564
1	Komlan Amewou	32	44	8	Can	2021	2025	0.35	0.46	3214
2	Bachirou Salou	35	55	4	Strasbourg	2022	2024	0.25	0.16	2239

In [109…
```
DELETE_By_Condition = my_conn.execute('''DELETE FROM Player_HAVE_Contract WHERE Name = 'Jean-Paul Abalo';''')
DELETE_By_Condition
```

Out[109]:
```
<sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f0fd0151340>
```

In [110…
```
#Check_After_DELETE
#pd.read_sql('SELECT * FROM Player_HAVE_Contract',my_conn)
#Output:
#0 Komlan Amewou        32      44      8       Can     2021    2025    0.35    0.46    3214
#1      Bachirou Salou  35      55      4       Strasbourg      2022    2024    0.25    0.16    2239
```

In [111…
```
#TRUNCATE TABLE is similar to DELETE, but this operation is a DDL (Data Definition Language) command.
#It also deletes records from a table without removing table
#structure, but it doesn't use the WHERE clause. Here's the syntax. Example:TRUNCATE TABLE table_name;
my_conn.execute('''DELETE FROM Player_HAVE_Contract;''') # In sqlite TRUNCATE TABLE table_name = DELETE FROM ta
```

Out[111]:
```
<sqlalchemy.engine.cursor.LegacyCursorResult at 0x7f0fd015fb80>
```

In [111…

In [112…
```
Check_After_TRUNCATE= pd.read_sql('SELECT * FROM Player_HAVE_Contract',my_conn)
Check_After_TRUNCATE
```

Out[112]:

Name	Age	OVA	POT	Team	Start_Contract	End_Contract	Value_In_€Million	Wage_In_€Million	Total

In [113…
```
Drop_unuse_Table = my_conn.execute('''DROP TABLE Player_with_Contract;''')
```

In [114…
```
#Check_Drop_table_Action = pd.read_sql('SELECT * FROM Player_with_Contract',my_conn)
#Check_Drop_table_Action
```

```
# Output is OperationalError: no such table: Player_with_Contract = Table does not exist
```

In [114…

In [115…
```
#pd.read_sql('SELECT * FROM Player_HAVE_Contract',my_conn)
```

In [116…
```
#df_contract_xlsl.to_sql('Player_Under_Contract',my_conn,if_exists='replace',index=False)
```

In [117…
```
#pd.read_sql('SELECT * FROM Player_Under_Contract',my_conn)
```

In [118…
```
#r_set=my_conn.execute('''select name from sqlite_master
                #where type = 'table' ''')
#for row in r_set:
  # print(row)
```

In [119…
```
r_set=my_conn.execute('''select name from sqlite_master
 where type = 'table' ''')
for row in r_set:
    print(row)
```

```
('Player_Under_Contract',)
('Player_HAVE_Contract',)
('Player_HAVE_Contrac',)
('Saint_Louis_Pop_by_Year',)
('Saint_Louis_Education_Attained',)
('Saint_Pop_By_Race',)
('Saint_Louis_Race_Education_Attained',)
('Pop_projected_based_on_recent',)
('new_df_Saint_Louis_Race_Education_Attained',)
('df_contract_xlsl',)
```

In [120…
```
# https://colab.research.google.com/github/AmirPupko/pandas-to-sql/blob/main/pandas_to_sql_colab_example.ipynb#
#https://www.youtube.com/watch?v=DzisTRNu1dA
#https://medium.com/@sarka.pribylova/sql-operations-in-python-a01065424911
#https://www.youtube.com/watch?v=MkGQmZoMuRM&t=922s
#https://learnsql.com/blog/difference-between-truncate-delete-and-drop-table-in-sql/
```

In [121…
```
Simple_Join_Or_Inner = ('''SELECT *
FROM table1
INNER JOIN table2
ON table1.column = table2.column;''')

#my_conn.execute(Simple_Join_Or_Inner)
```