

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG  
KHOA CÔNG NGHỆ THÔNG TIN**



# **BÁO CÁO BÀI TẬP LỚN**

## **Lập trình với Python**

**Giảng viên**  
**Nhóm**  
**Tên sinh viên**  
**Mã sinh viên**

**: Kim Ngọc Bách**  
**: 09**  
**: Nguyễn Năng Vương**  
**: B22DCCN923**

Bài 1:

## Cấu trúc chương trình:

1. Đọc HTML của trang web bằng requests và dùng BeautifulSoup để đọc nội dung html:

```
6 # URL của trang web
7 url = 'https://fbref.com/en/comps/9/2023-2024/stats/2023-2024-Premier-League-Stats'
8
9 # Gửi yêu cầu và lấy nội dung trang web
10 r = requests.get(url)
11 soup = bs(r.content, 'html.parser')
```

2. Dữ liệu cầu thủ cần lấy có nhiều nhóm chỉ số khác nhau nên cần lấy ra thẻ chứa các link dẫn tới bảng chứa các nhóm chỉ số cần tìm:

```
13 # Tìm tất cả các mục <li> trong <ul> sau <p class="listhead">
14 data = []
15 for item in soup.find('p', class_='listhead').find_next('ul').find_all('li'):
16     title = item.text.strip() # Lấy tên mục
17     link = 'https://fbref.com' + item.find('a')['href'] # Lấy đường dẫn đầy đủ
18     data.append({'title': title, 'link': link}) # Lưu vào danh sách dưới dạng từ điển
```

- Kết quả chương trình:

```
Standard Stats: https://fbref.com/en/comps/9/2023-2024/stats/2023-2024-Premier-League-Stats
Goalkeeping: https://fbref.com/en/comps/9/2023-2024/keepers/2023-2024-Premier-League-Stats
Advanced Goalkeeping: https://fbref.com/en/comps/9/2023-2024/keepersadv/2023-2024-Premier-League-Stats
Shooting: https://fbref.com/en/comps/9/2023-2024/shooting/2023-2024-Premier-League-Stats
Passing: https://fbref.com/en/comps/9/2023-2024/passing/2023-2024-Premier-League-Stats
Pass Types: https://fbref.com/en/comps/9/2023-2024/passing_types/2023-2024-Premier-League-Stats
Goal and Shot Creation: https://fbref.com/en/comps/9/2023-2024/gca/2023-2024-Premier-League-Stats
Defensive Actions: https://fbref.com/en/comps/9/2023-2024/defense/2023-2024-Premier-League-Stats
Possession: https://fbref.com/en/comps/9/2023-2024/possession/2023-2024-Premier-League-Stats
Playing Time: https://fbref.com/en/comps/9/2023-2024/playingtime/2023-2024-Premier-League-Stats
Miscellaneous Stats: https://fbref.com/en/comps/9/2023-2024/misc/2023-2024-Premier-League-Stats
```

3. Vì bảng chứa dữ liệu các cầu thủ bị ẩn trong Comment của HTML nên cần lấy ra Comment trong các link:

```
21 def get_url(url):
22     # Gửi yêu cầu đến URL và tạo BeautifulSoup từ nội dung HTML
23     r = requests.get(url)
24     soup = bs(r.content, 'html.parser')
25     time.sleep(0.5)
26     print('Loading...')
27     # Tìm tất cả các comment trong nội dung HTML
28     comments = soup.find_all(string=lambda text: isinstance(text, Comment))
29     return comments
```

- Vì việc đọc dữ liệu từ trang web quá nhanh có thể làm cho máy chủ chận nên có thể thêm thời gian chờ.

4. Để đọc được các chỉ số trong mỗi link cần 1 hàm để phân tích html:

```
30  ⊞ def get_goalkeeper_stats(comments): ...
105 ⊞ def get_shooting_stats(comments): ...
187 ⊞ def get_standard_stats(comments): ...
283 ⊞ def get_passing_stats(comments): ...
379 ⊞ def get_passing_types_stats(comments): ...
548 ⊞ def get_gca_stats(comments): ...
636 ⊞ def get_defense_stats(comments): ...
723 ⊞ def get_possession_stats(comments): ...
821 ⊞ def get_playing_time_stats(comments): ...
909 ⊞ def get_misc_stats(comments): ...
```

- Vì mỗi nhóm chỉ số được lưu trong các bảng khác nhau nên cần viết các hàm riêng cho từng link, mỗi hàm trả về 1 dataframe chứa các nhóm chỉ số.

5. Hợp nhất các dataframe trong các link tìm được và sắp xếp:

```
992 dataframes = [
993     get_standard_stats(get_url(data[0]['link'])),
994     get_goalkeeper_stats(get_url(data[1]['link'])),
995     get_shooting_stats(get_url(data[3]['link'])),
996     get_passing_stats(get_url(data[4]['link'])),
997     get_passing_types_stats(get_url(data[5]['link'])),
998     get_gca_stats(get_url(data[6]['link'])),
999     get_defense_stats(get_url(data[7]['link'])),
1000     get_possession_stats(get_url(data[8]['link'])),
1001     get_playing_time_stats(get_url(data[9]['link'])),
1002     get_misc_stats(get_url(data[10]['link']))
1003 ]
1004
1005 # Khởi tạo DataFrame kết quả với DataFrame đầu tiên
1006 df = dataframes[0]
1007
1008 # Gộp tất cả các DataFrame vào df_combined
1009 for df_clone in dataframes[1:]:
1010     df = pd.merge(df, df_clone, on=['Player', 'Nation', 'Team', 'Position', 'Age'], how='outer')
1011 df['First Name'] = df['Player'].apply(lambda x: x.split()[0]) # Lấy tên đầu tiên
1012
1013 df = df.sort_values(by=['First Name', 'Age'], ascending=[True, False])
1014 df = df.drop(columns=['First Name'])
```

6. Lưu vào file result.csv:

```
1015 # Kiểm tra kết quả
1016 df.to_csv('/Users/nangvuong/Documents/CODE PTIT/Python/result.csv', sep=';', index=False)
```

- Kết quả khi mở file result.csv bằng excel:

Home

Insert

Draw

Page Layout

Formulas

Data

Review

View

Calibri (Body)

12

Bài 2:

1. Tìm top 3 cầu thủ có điểm cao nhất và thấp nhất ở mỗi chỉ số.
  - a. Lấy dữ liệu các cầu thủ từ file result.csv và đổi các dữ liệu có dạng NaN thành 0.
  - b. Dùng phương thức nlargest, smallest có sẵn trong thư viện pandas để tìm ra top 3 cao nhất và thấp nhất ở mỗi chỉ số.
  - c. In ra màn hình.

Kết quả:

Top 3 cầu thủ có Non-Penalty Goals cao nhất:

|     | Player         | Non-Penalty Goals |
|-----|----------------|-------------------|
| 221 | Erling Haaland | 27.0              |
| 148 | Cole Palmer    | 22.0              |
| 24  | Alexander Isak | 21.0              |

Top 3 cầu thủ có Non-Penalty Goals thấp nhất:

|   | Player            | Non-Penalty Goals |
|---|-------------------|-------------------|
| 0 | Aaron Cresswell   | 0.0               |
| 1 | Aaron Ramsdale    | 0.0               |
| 2 | Aaron Wan-Bissaka | 0.0               |

2. Tìm trung vị, trung bình, độ lệch chuẩn của mỗi chỉ số cho toàn giải và cho mỗi đội:
  - a. Lấy dữ liệu các cầu thủ từ file result.csv và đổi các dữ liệu có dạng NaN thành 0.
  - b. Dùng phương thức median, mean, std trong thư viện pandas để tìm trung vị, trung bình, độ lệch chuẩn cho từng chỉ số và dùng hàm round để làm tròn đến 2 chữ số sau dấu phẩy.
  - c. Lưu kết quả vào file results2.csv
    - Kết quả file results2.csv khi mở bằng excel:

| Team           | Median of A | Median of N | Median of P | Median of A | Median of P | Median of G | Median of A | Median of G | Median of G | Median of x | Median of x | Median of n | Median of M | Median of St | Median of G | Median of V |
|----------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|--------------|-------------|-------------|
| All            | 24.0        | 0.0         | 0.0         | 0.0         | 7.0         | 14.0        | 8.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.04        | 0.03        | 0.09         | 0.04        | 0.09        |
| Arsenal        | 23.5        | 0.0         | 0.0         | 0.5         | 10.0        | 21.5        | 9.0         | 0.0         | 0.02        | 0.1         | 0.0         | 0.04        | 0.04        | 0.15         | 0.04        | 0.15        |
| Aston Villa    | 24.0        | 0.0         | 0.0         | 0.0         | 3.0         | 5.0         | 1.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.04        | 0.09        | 0.04         | 0.09        | 0.0         |
| Bournemouth    | 24.0        | 0.0         | 0.0         | 0.0         | 4.0         | 20.0        | 4.0         | 0.0         | 0.0         | 0.05        | 0.0         | 0.05        | 0.02        | 0.06         | 0.08        | 0.02        |
| Brentford      | 24.0        | 0.0         | 0.0         | 0.0         | 10.0        | 21.0        | 13.0        | 0.0         | 0.0         | 0.07        | 0.0         | 0.07        | 0.04        | 0.07         | 0.14        | 0.0         |
| Brighton       | 21.0        | 0.0         | 0.0         | 0.0         | 13.0        | 21.0        | 13.0        | 0.0         | 0.0         | 0.0         | 0.0         | 0.05        | 0.04        | 0.13         | 0.05        | 0.13        |
| Burnley        | 24.0        | 0.0         | 0.0         | 0.0         | 9.0         | 19.0        | 17.0        | 0.0         | 0.0         | 0.08        | 0.0         | 0.08        | 0.05        | 0.05         | 0.12        | 0.05        |
| Chelsea        | 20.0        | 0.0         | 0.0         | 0.0         | 0.5         | 2.0         | 2.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.02         | 0.0         | 0.02        |
| Crystal Palace | 23.0        | 0.0         | 0.0         | 0.0         | 7.0         | 10.0        | 5.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.02        | 0.01        | 0.04         | 0.02        | 0.04        |
| Everton        | 24.5        | 0.0         | 0.0         | 0.0         | 6.0         | 14.0        | 18.0        | 0.0         | 0.0         | 0.02        | 0.0         | 0.02        | 0.06        | 0.06         | 0.14        | 0.06        |
| Fulham         | 26.0        | 0.5         | 0.5         | 0.0         | 9.0         | 21.5        | 6.0         | 0.02        | 0.0         | 0.09        | 0.02        | 0.09        | 0.05        | 0.02         | 0.12        | 0.05        |
| Liverpool      | 22.0        | 0.0         | 0.0         | 0.0         | 9.0         | 30.0        | 2.5         | 0.0         | 0.0         | 0.0         | 0.0         | 0.04        | 0.01        | 0.07         | 0.04        | 0.07        |
| Luton Town     | 25.0        | 0.0         | 0.0         | 0.0         | 6.5         | 18.0        | 6.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.02        | 0.02        | 0.06         | 0.02        | 0.06        |
| Manchester     | 25.0        | 0.5         | 0.5         | 0.0         | 22.5        | 33.5        | 30.0        | 0.03        | 0.0         | 0.11        | 0.03        | 0.11        | 0.08        | 0.06         | 0.16        | 0.08        |
| Manchester     | 24.0        | 0.0         | 0.0         | 0.0         | 6.0         | 10.0        | 2.0         | 0.0         | 0.0         | 0.06        | 0.0         | 0.06        | 0.01        | 0.04         | 0.08        | 0.01        |
| Newcastle U    | 25.0        | 1.0         | 1.0         | 0.0         | 11.0        | 29.0        | 21.0        | 0.07        | 0.0         | 0.13        | 0.07        | 0.13        | 0.06        | 0.04         | 0.11        | 0.06        |
| Nott'ham Fo    | 25.0        | 0.0         | 0.0         | 0.0         | 4.0         | 12.0        | 6.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.04        | 0.04        | 0.08         | 0.04        | 0.08        |
| Sheffield Utd  | 22.5        | 0.0         | 0.0         | 0.0         | 3.0         | 7.5         | 6.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.02        | 0.02        | 0.06         | 0.02        | 0.06        |
| Tottenham      | 24.0        | 0.0         | 0.0         | 0.0         | 14.0        | 26.5        | 15.0        | 0.0         | 0.0         | 0.04        | 0.0         | 0.04        | 0.09        | 0.02         | 0.14        | 0.09        |
| West Ham       | 26.0        | 0.0         | 0.0         | 0.0         | 4.0         | 20.5        | 4.5         | 0.0         | 0.0         | 0.02        | 0.0         | 0.02        | 0.04        | 0.01         | 0.08        | 0.04        |
| Wolves         | 22.0        | 0.0         | 0.0         | 0.0         | 1.0         | 7.0         | 7.0         | 0.0         | 0.0         | 0.0         | 0.0         | 0.03        | 0.03        | 0.07         | 0.03        | 0.07        |

3. Vẽ histogram phân bố của mỗi chỉ số của các cầu thủ trong toàn giải và mỗi đội:
  - a. Đọc dữ liệu cầu thủ từ file result.csv và đổi các dữ liệu NaN thành 0.
  - b. Vì chỉ số của các cầu thủ rất nhiều nên chỉ lấy những giá trị đại diện cho những nhóm chỉ số.

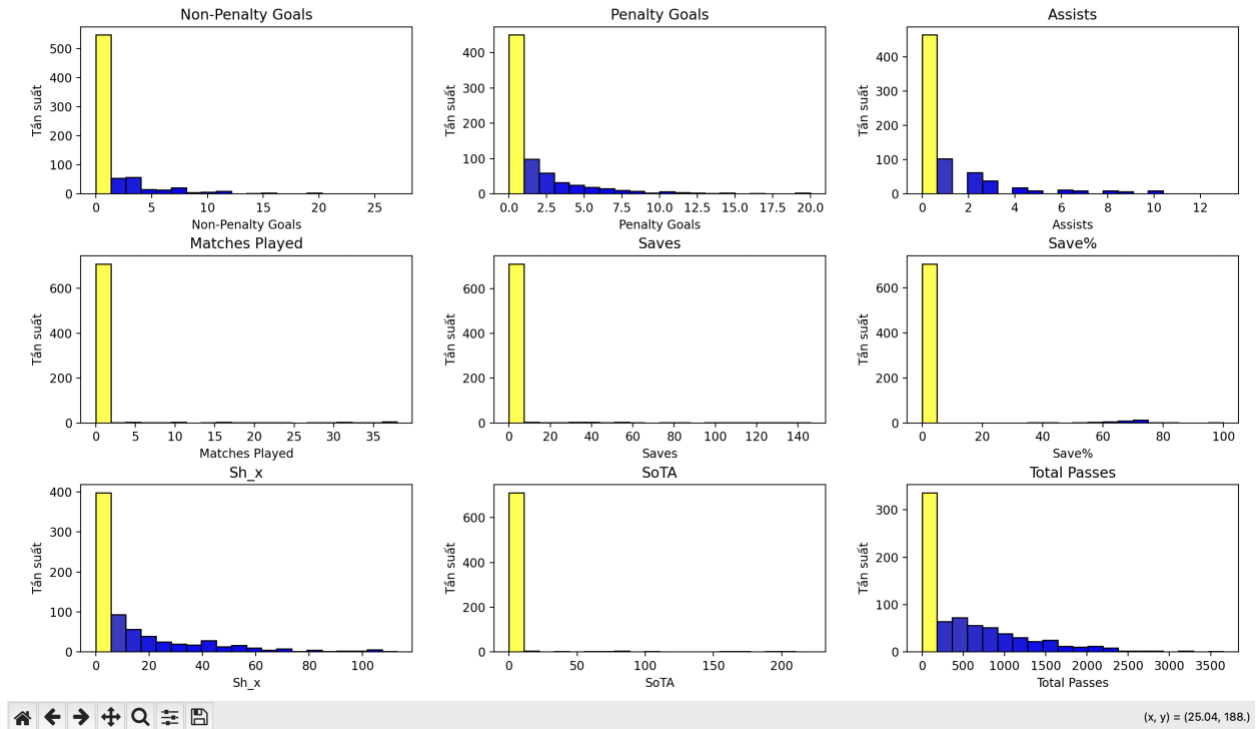
```

12 columns_to_plot = [
13     'Non-Penalty Goals', 'Penalty Goals', 'Assists', 'Matches Played',
14     'Saves', 'Save%', 'Sh_x', 'SoTA',
15     'Total Passes', 'Cmp%', 'xG', 'xAG',
16     'Tkl', 'Int', 'Minutes', 'Starts_x',
17     'Yellow Cards', 'Red Cards'
18 ]

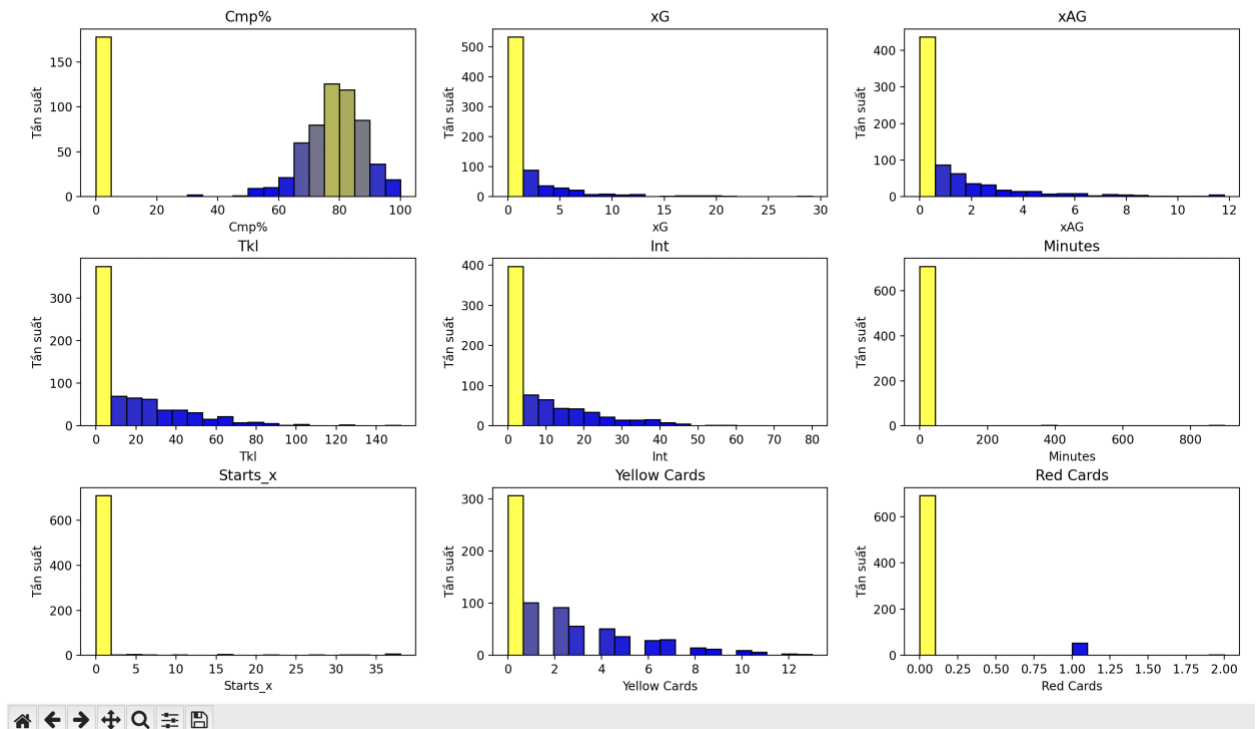
```

- c. Dùng thư viện matplotlib để vẽ histograms.
- d. Kết quả:

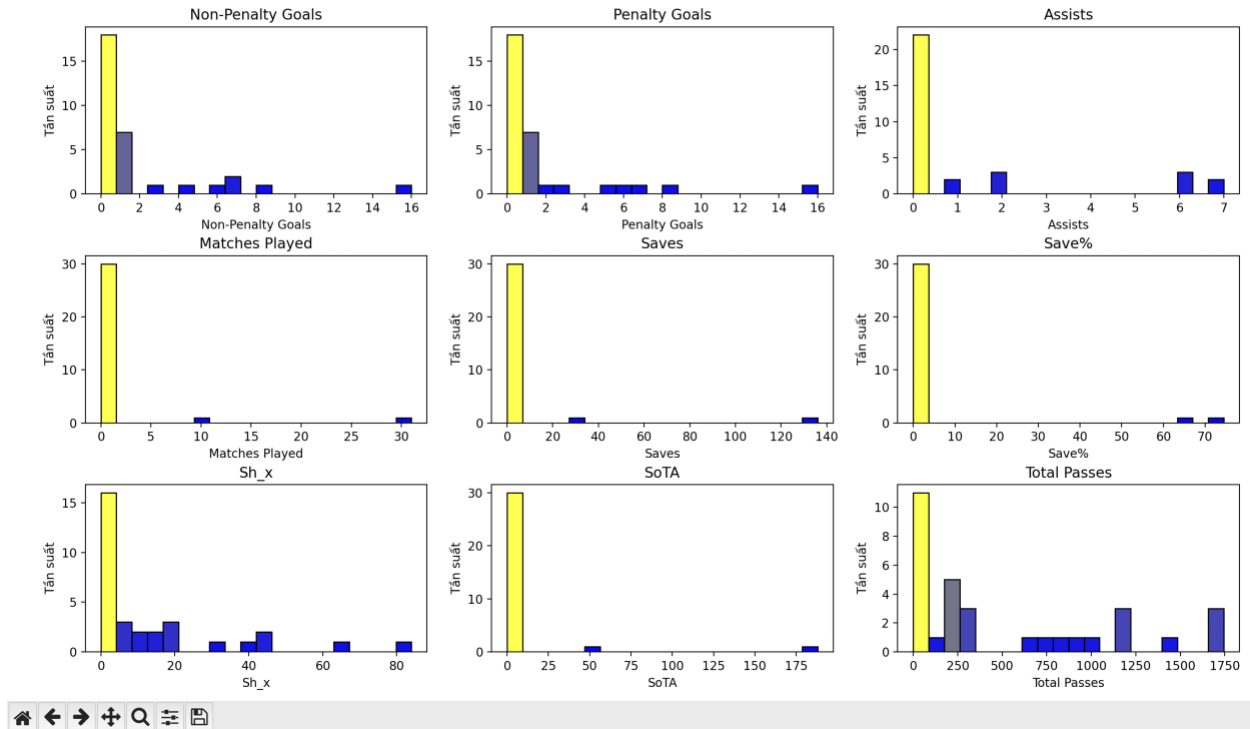
Biểu đồ phân bố cho toàn giải



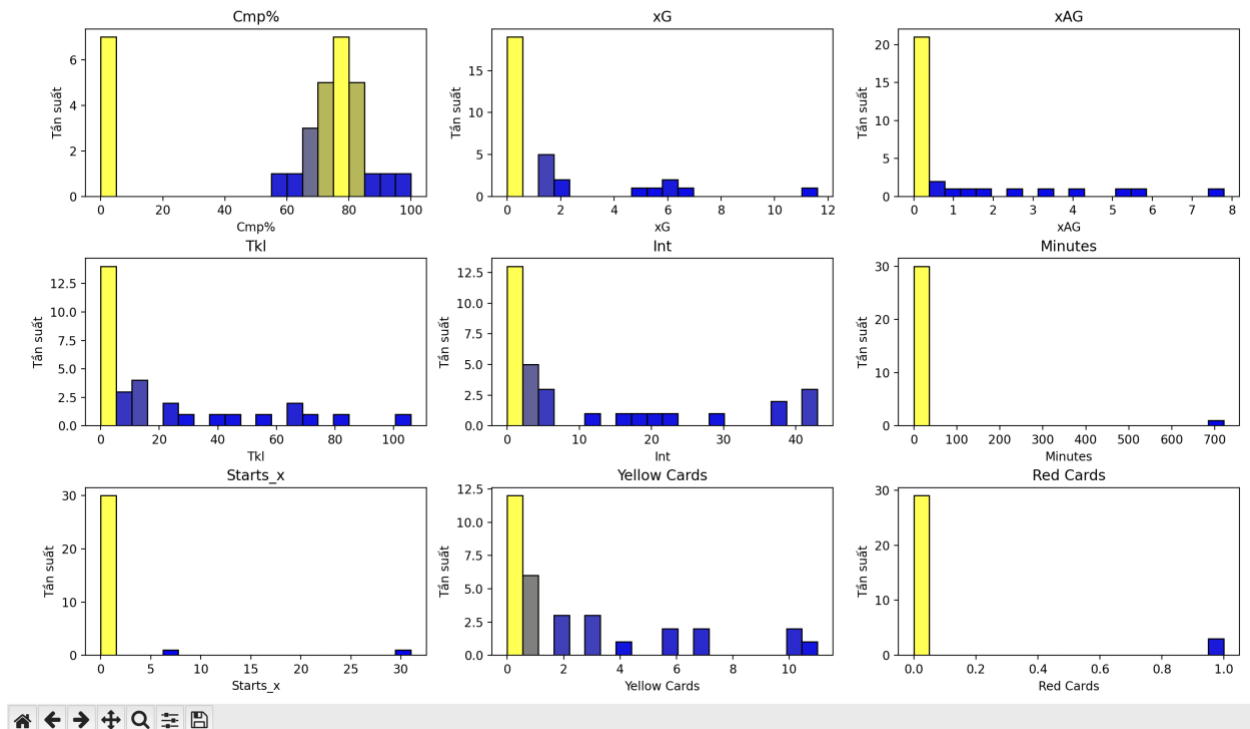
Biểu đồ phân bố cho toàn giải



Biểu đồ phân bố cho câu lạc bộ West Ham



Biểu đồ phân bố cho câu lạc bộ West Ham



4. Tìm đội bóng có chỉ số điểm số cao nhất ở mỗi chỉ số.

a. Lấy dữ liệu được lưu trong file results2.csv.

b. Sử dụng phương thức max() để tìm đội có trung bình chỉ số cao nhất.



c. In kết quả ra màn hình.

```
Yellow Cards: Wolves
Red Cards: Burnley
Yellow-Red Cards: Burnley
Fls: Everton
Fld: Tottenham
Off_y: Everton
Crs_y: Everton
Won: Everton
Lost_y: Burnley
Won%: Nott'ham Forest
Recov: Everton
```

```
Đội bóng có phong độ tốt nhất giải Ngoại Hạng Anh mùa 2023-2024:
Manchester City với 80 chỉ số cao nhất.
```

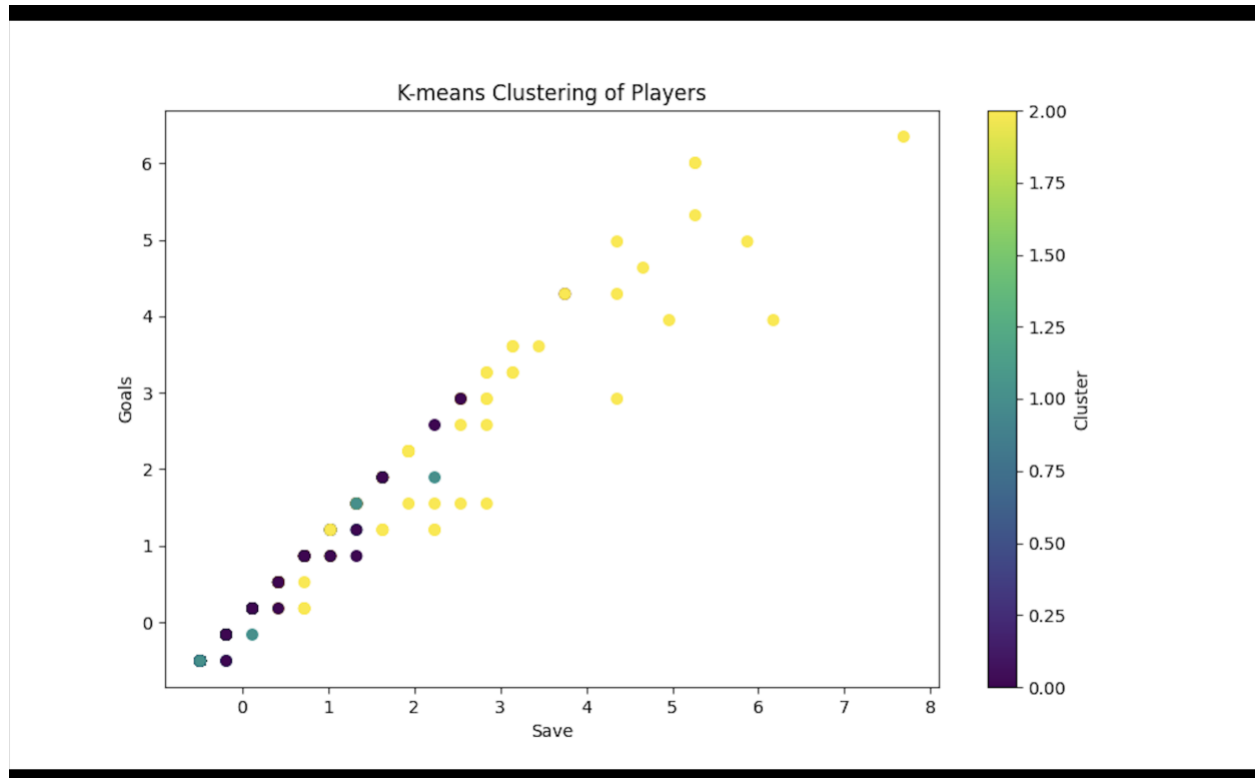
⇒ Vậy đội có phong độ tốt nhất giải là Manchester City.

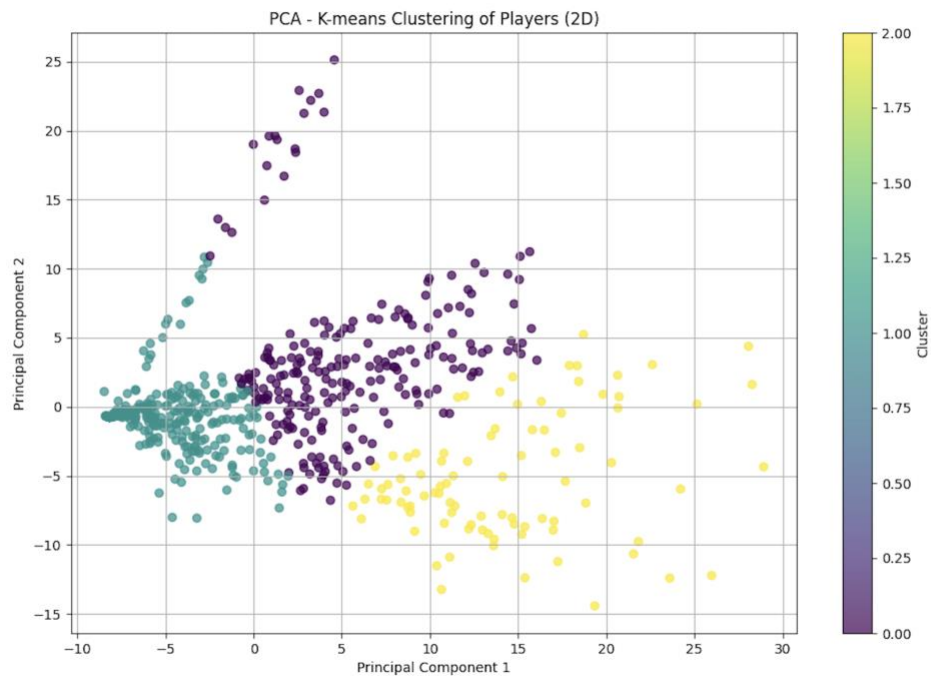
Bài 3:

1. *Sử dụng thuật toán K-means để phân loại các cầu thủ thành các nhóm có chỉ số giống nhau.*
  - a. Lấy dữ liệu cầu thủ từ file result.csv, đổi các dữ liệu NaN thành 0
  - b. Sử dụng thư viện có sẵn Sklearn để phân loại các cầu thủ bằng thuật toán Kmeans, sử dụng thư viện matplotlib để vẽ hình dữ liệu trên mặt 2D.
  - c. Chọn ra 2 chỉ số của cầu thủ để phân loại so sánh.

Kết quả với 2 chỉ số là Goals, Save:





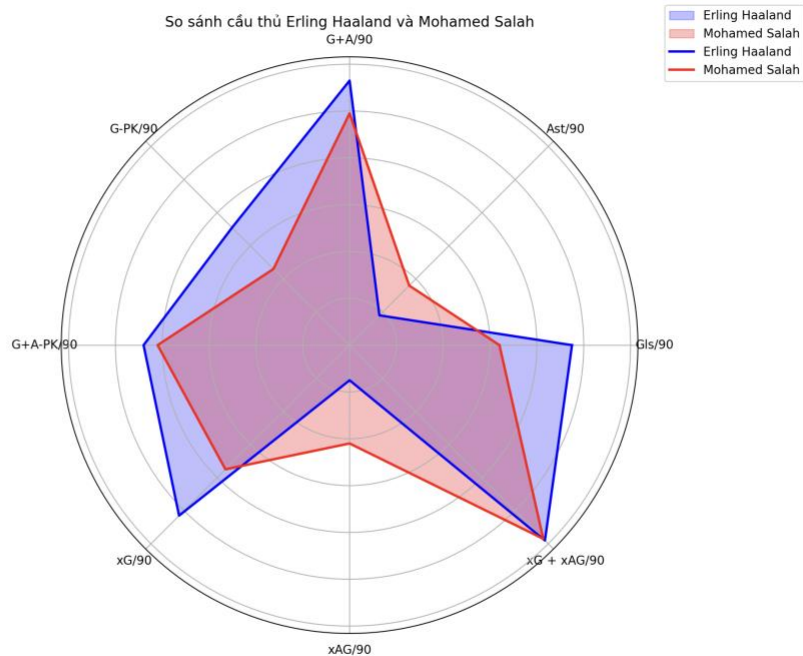


#### 4. Viết chương trình python vẽ biểu đồ rada (radar chart) so sánh cầu thủ

- Lấy dữ liệu cầu thủ từ file result.csv để so sánh, đổi các dữ liệu NaN thành 0.
- Dùng thư viện matplotlib để vẽ radar chart.
- Viết hàm chính với thư viện argparse để nhận input và xử lý dữ liệu.

Kết quả khi so sánh cầu thủ Erling Haaland và Mohamed Salah với các chỉ số Gls/90, Ast/90, G+A/90, G-PK/90, G+A-PK/90, xG/90, xAG/90, xG + xAG/90 bằng câu lệnh:

```
python3 "/Users/nangvuong/Documents/CODE PTIT/Python/BÀI3(rada chart).py" --p1 "Erling Haaland" --p2
"Mohamed Salah" --Attribute "Gls/90;Ast/90;G+A/90;G-PK/90;G+A-PK/90;xG/90;xAG/90;xG + xAG/90"
```



$\theta=0.8720\pi$  (157.0°),  $r=1.157$

Bài 4: **Lấy dữ liệu chuyển nhượng các cầu thủ từ trang <https://www.footballtransfers.com>.**

1. Vì trang <https://www.footballtransfers.com> sử dụng JavaScript để tải dữ liệu cầu thủ khi truy cập, vì vậy sử dụng `requests.get` sẽ chỉ lấy html tĩnh, không có nội dung tải động như dữ liệu cầu thủ. Để lấy dữ liệu cầu thủ trên, em sử dụng thư viện **Selenium** để giả lập trình duyệt và tải hết nội dung.

```

6  def read_url(url):
7      driver = webdriver.Safari()
8      driver.get(url)
9      time.sleep(5) # Đợi trang tải xong
10     page_source = driver.page_source
11     driver.quit()
12     soup = bs(page_source, 'html.parser')
13     return soup

```

Có thể thay bằng webdriver của các trình duyệt khác như Chrome,...

2. *Url của chứa dữ liệu các cầu thủ là <https://www.footballtransfers.com/en/values/players/most-valuable-players/playing-in-uk-premier-league> và các page của url này.*

```

47 # Đọc dữ liệu từ trang 1
48 url_page_1 = 'https://www.footballtransfers.com/en/values/players/most-valuable-players/playing-in-uk-premier-league'
49 soup_page_1 = read_url(url_page_1)
50 players_data_page_1 = extract_player_data(soup_page_1)
51 all_players_data = pd.concat([all_players_data, pd.DataFrame(players_data_page_1)], ignore_index=True)
52
53 # Đọc dữ liệu từ trang 2 đến trang 24
54 for page in range(2, 25):
55     url = f'https://www.footballtransfers.com/en/values/players/most-valuable-players/playing-in-uk-premier-league/{page}'
56     soup = read_url(url)
57     players_data = extract_player_data(soup)
58     all_players_data = pd.concat([all_players_data, pd.DataFrame(players_data)], ignore_index=True)
59     print(f"Data from page {page} added to DataFrame.")

```

3. Sau đó đọc dữ liệu các cầu thủ trong thẻ tr và lấy dữ liệu cần tìm và lưu vào file test.csv.  
 Kết quả khi mở file test.csv:

```

1 Player Name;Team;Transfer Value
2 Erling Haaland;Man City;€149.3M
3 Bukayo Saka;Arsenal;€117.5M
4 Phil Foden;Man City;€116.1M
5 Rodri;Man City;€100.4M
6 Kai Havertz;Arsenal;€95M
7 William Saliba;Arsenal;€79.5M
8 Martin Ødegaard;Arsenal;€73.2M
9 Rúben Dias;Man City;€70.5M
10 Alexander Isak;Newcastle Utd.;€69.8M
11 Declan Rice;Arsenal;€69.1M
12 Cole Palmer;Chelsea;€67.3M
13 Bruno Guimarães;Newcastle Utd.;€65.3M
14 Josko Gvardiol;Man City;€65M

```

### Phương pháp định giá cầu thủ:

- ⇒ Dựa vào bảng giá chuyển nhượng các cầu thủ, ta có thể định giá các cầu thủ dựa trên những tiêu chí:
- Phân tích hiệu suất cầu thủ
  - Yếu tố tuổi tác và tiềm năng phát triển
  - Yếu tố hợp đồng và điều khoản chuyển nhượng
  - Phân tích thương mại và giá trị truyền thông
  - Sử dụng mô hình định giá hiện đại