

## Data Preprocessing

### 1. Data Merging

**Python:**

- Merge data according to the common columns in the data files.
- Solution is provided in the Python notebook.
- Save the final files as 'Football\_data\_merged'.csv file.
- While joining tables, "outer join" can also be used.

**Tableau:**

- Put all 5 data files in single Excel files (5 worksheets).
- Upload the data in Tableau.
- Do merging according to the common columns in data files, as shown below:



### 2. Data Dictionary

In ChatGPT, write prompt as:

As a data analyst, provide data dictionary for the following data with description in tabular form.

Copy the first few samples (10-15) from the data file 'Football\_data\_merged'.csv file including column names and paste it here.

#### Solution provided by ChatGPT

Here is a data dictionary for the provided data:

Column Name	Description
appearance_id	Unique identifier for each appearance record
game_id_x	Unique identifier for each game record
player_id	Unique identifier for each player record
date_x	Date of the game appearance in the format DD-MM-YYYY
player_name_x	Player's name
competition_id_x	Unique identifier for each competition record

<b>Column Name</b>	<b>Description</b>
yellow_cards	Number of yellow cards received by the player in the game
red_cards	Number of red cards received by the player in the game
goals	Number of goals scored by the player in the game
assists	Number of assists made by the player in the game
minutes_played	Number of minutes the player played in the game
competition_id_y	Unique identifier for each competition record (duplicated)
season	Season of the game in YYYY format
round	Round of the competition
date_y	Date of the game in the format DD-MM-YYYY
home_club_goals	Number of goals scored by the home club
away_club_goals	Number of goals scored by the away club
home_club_position	Position of the home club in the league
away_club_position	Position of the away club in the league
home_club_manager_name	Manager's name of the home club
away_club_manager_name	Manager's name of the away club
stadium	Name of the stadium where the game was played
attendance	Number of spectators in the stadium
referee	Referee's name
home_club_formation	Formation used by the home club
away_club_formation	Formation used by the away club
home_club_name	Name of the home club
away_club_name	Name of the away club
aggregate	Aggregate score
competition_type	Type of competition (e.g., domestic_league)
game_event_id	Unique identifier for each game event record
date	Date of the game event in the format DD-MM-YYYY
minute	Minute of the game event

Column Name	Description
type_x	Type of game event (e.g., Substitutions, Goals, Cards)
player_id_y	Unique identifier for each player record (duplicated)
description	Description of the game event
player_in_id	Unique identifier for the player coming into the game
player_assist_id	Unique identifier for the assisting player
name	Name of the player
last_season	Last season the player participated in
current_club_id	Unique identifier for the current club of the player
player_code	Code associated with the player
country_of_birth	Country of birth of the player
date_of_birth	Date of birth of the player in the format DD-MM-YYYY
sub_position	Substitute position of the player (if applicable)
position_x	Position of the player on the field
foot	Preferred foot of the player
height_in_cm	Height of the player in centimeters
market_value_in_eur	Market value of the player in euros
highest_market_value_in_eur	Highest market value of the player in euros
contract_expiration_date	Expiration date of the player's contract
agent_name	Name of the player's agent
game_lineups_id	Unique identifier for each game lineup record
game_id_y	Unique identifier for each game record (duplicated)
type_y	Type of game lineup (e.g., substitutes)
number	Player's jersey number
player_name_y	Player's name
team_captain	Indicates if the player is the team captain
position_y	Player's position on the field

Note: The data seems to have duplicated columns (e.g., game\_id\_x and game\_id\_y) that might be a result of merging or joining datasets. Depending on your analysis, you may want to clean the data and remove redundant columns.

In football, an **assisting player** is often associated with someone who provides **assists**. An assist is a pass or a touch that directly leads to a goal scored by a teammate. The assisting player contributes to the team's attacking play by creating opportunities for others to score.

They may not necessarily be the main goal scorers themselves, but they play a crucial role in the team's offensive dynamics.

**Substitution Time:** Substitutions can typically be made at specific moments during the game:

- **Normal Playing Time:** Substitutions are usually allowed during stoppages in play, such as throw-ins, goal kicks, corner kicks, and injuries. Players can enter or leave the field during these breaks.
- **Half-Time:** Teams can also make substitutions during the half-time break.

### 3. Data Cleaning

- Remove the redundant columns using Python.
- Null value treatment (data should not be reduced drastically)
- Save the cleaned data file as ‘Football\_data\_merged\_cleaned’.csv file.
- Solution is provided in the Python notebook/
- Again, create data dictionary using ChatGPT as created above.

Here is a data dictionary for the provided data:

Field	Description
appearance_id	Unique identifier for each appearance
game_id	Identifier for the game
player_id	Identifier for the player
competition_id	Identifier for the competition
yellow_cards	Number of yellow cards received
red_cards	Number of red cards received
goals	Number of goals scored
assists	Number of assists made
minutes_played	Total minutes played in the game
Field	Description
season	Season of the game

Field	Description
round	Round of the competition
home_club_goals	Goals scored by the home club in the game
away_club_goals	Goals scored by the away club in the game
home_club_position	Position of the home club in the league
away_club_position	Position of the away club in the league
home_club_manager_name	Manager of the home club
away_club_manager_name	Manager of the away club
stadium	Stadium where the game was played
attendance	Number of spectators in the game
referee	Referee for the game
home_club_name	Name of the home club
away_club_name	Name of the away club
aggregate	Aggregate score in the competition
competition_type	Type of competition (e.g., domestic_league)
game_event_id	Identifier for the game event
date	Date of the game event
minute	Minute of the game event
type_x	Type of game event (e.g., Substitutions, Goals, Cards)
player_in_id	Identifier for the substitute player
name	Name of the player
last_season	Last season the player participated in
current_club_id	Identifier for the current club of the player
player_code	Code associated with the player
country_of_birth	Country of birth of the player
date_of_birth	Date of birth of the player
sub_position	Position of the substitute player
position_x	Position of the player (e.g., Defender, Midfielder)

foot	Preferred foot of the player
height_in_cm	Height of the player in centimeters
market_value_in_eur	Current market value of the player in Euros
highest_market_value_in_eur	Highest market value of the player in Euros
contract_expiration_date	Expiration date of the player's contract
game_lineups_id	Identifier for the game lineup
type_y	Type of lineup (e.g., substitutes)
number	Jersey number of the player
player_name	Name of the player in the lineup
team_captain	Whether the player is the team captain (0 or 1)
position_y	Position of the player in the lineup

- Note: The table includes only the unique fields from the provided data. Depending on the specific requirements of your analysis, you may need additional information or a more detailed explanation of certain fields.

#### 4. Generate Business Objectives using ChatGPT

The data structures including column names and their descriptions are provided below. As a business analyst, provide all possible business objectives for this data.

As a business analyst, you can derive several business objectives from the provided data. Here are some possible business objectives:

- 1. Performance Analysis:**
  - Evaluate and analyze player performance based on various metrics such as goals, assists, yellow cards, red cards, and minutes played.
  - Assess team performance in terms of home and away goals, club positions, and manager influence.
- 2. Player Profile and Market Value:**
  - Create player profiles, including personal information, playing position, and current market value.
  - Analyze trends in market values and contract expiration dates for players.
- 3. Team Comparison:**
  - Compare and contrast the performance of different clubs, managers, and teams in various seasons and rounds.
  - Analyze aggregate data to identify patterns in team performance.

#### **4. Attendance and Stadium Analysis:**

- Evaluate attendance trends and analyze the influence of factors such as stadiums and referees on attendance.
- Assess the impact of stadium capacity on home club goals.

#### **5. Referee Analysis:**

- Evaluate referee performance and its potential impact on match outcomes.
- Analyze the distribution of cards and goals in matches officiated by different referees.

#### **6. Substitution Patterns:**

- Study substitution patterns, including the timing and frequency of player substitutions.
- Assess the impact of substitutions on match outcomes.

#### **7. Event Analysis:**

- Analyze specific events (goals, cards, substitutions) in matches and identify patterns.
- Study the correlation between specific events and match results.

#### **8. Competition Analysis:**

- Compare and analyze data across different competitions and match types.
- Assess the influence of competition type on player and team performance.

#### **9. Player Attributes and Demographics:**

- Analyze player attributes such as foot preference, height, and country of birth.
- Study the relationship between player attributes and performance.

#### **10. Contract Management:**

- Monitor player contract expiration dates and identify players with contracts nearing expiration.
- Assess the impact of contract status on player performance.

These objectives can help guide further analysis and provide valuable insights for decisionmaking in various aspects of the football business.

## **5. Business Questions**

Provide all possible business questions for the objectives above.

Certainly! Here are specific business questions aligned with the mentioned objectives:

### **1. Performance Analysis**

1. How do the number of goals and assists correlate with total minutes played for players in a specific season?

2. What is the distribution of yellow and red cards among players, and how does this relate to their minutes played?
3. Which players had the highest impact on game outcomes based on goals and assists?
4. How does a player's performance (goals, assists) vary by competition type?
5. What are the trends in player appearances over multiple seasons, and how do they relate to performance metrics?
6. How does the home club's formation affect the number of goals scored in matches?
7. What is the average number of appearances per player by position (e.g., forward, midfielder)?
8. How do game events (e.g., goals, assists) cluster within specific minutes of the game?
9. What percentage of players receive yellow cards during specific competitions?
10. How does the team captaincy influence player performance metrics such as goals and assists?

## **2. Player Profile and Market Value**

1. What is the average market value of players by position and how does it compare to their performance metrics?
2. How does a player's age and height correlate with their market value?
3. What trends can be observed in players' market values over multiple seasons?
4. How does the current club's performance affect a player's market value?
5. Which countries of birth have players with the highest average market values?
6. How do the highest market values of players relate to their contract expiration dates?
7. What impact does a player's foot preference have on their market value?
8. How do agents influence the market value of their players?
9. What is the average time remaining on contracts for players at different market value tiers?
10. How do players' sub-positions relate to their market values?

## **3. Team Comparison**

1. How do home and away club goals compare across different competitions?
2. What patterns emerge in team performance based on the home club's formation?
3. How do attendance figures correlate with the home club's position in the league?
4. What is the average number of yellow and red cards issued per team in a season?
5. How do the win/loss ratios of teams relate to the managers' names and strategies?
6. How do average player appearances compare between top-performing and bottom-performing teams?
7. What is the relationship between a team's average age and its performance metrics?

8. How do teams with high attendance rates perform compared to those with lower attendance?
9. What are the differences in player market values between competing teams in the same league?
10. How do player substitutions affect team performance during critical match periods?

#### **4. Attendance and Stadium Analysis**

1. What factors most significantly influence attendance at home games?
2. How does the stadium capacity correlate with actual match attendance?
3. What are the attendance trends for specific teams during different rounds of a competition?
4. How do match outcomes (home club goals vs. away club goals) impact future attendance?
5. What is the average attendance across different competitions, and how does it compare by season?
6. How does the performance of the home club in previous matches affect attendance?
7. What patterns emerge in attendance based on the day of the week or time of the match?
8. How does the presence of a specific referee impact match attendance?
9. What are the attendance figures for matches with significant game events (e.g., last-minute goals)?
10. How does a team's position in the league at the end of the season correlate with overall attendance?

#### **5. Referee Analysis**

1. What is the average number of yellow and red cards issued per referee in a season?
2. How does the performance of referees affect the outcome of matches (home club goals vs. away club goals)?
3. What trends can be identified in the number of penalties awarded by different referees?
4. How does referee experience correlate with the number of fouls called during a match?
5. What is the relationship between referee decisions and game events such as goals or assists?
6. How do referee assignments vary for high-stakes matches versus regular matches?
7. What is the impact of referee nationality on game outcomes?
8. How often do referees issue cards for specific types of fouls?
9. What patterns emerge when analyzing the consistency of a referee across different competitions?
10. How does the average duration of matches vary with different referees?

#### **6. Substitution Patterns**

1. How do substitution patterns impact team performance in the second half of games?
2. What is the average number of substitutions made by teams during critical match situations?

3. How do different positions (e.g., forwards vs. defenders) affect substitution decisions?
4. How does the timing of substitutions correlate with game outcomes (win/loss)?
5. What is the relationship between a player's performance and their likelihood of being substituted?
6. How do player appearances impact substitution strategies across various competitions?
7. What patterns emerge in substitution frequency based on player age and experience?
8. How does the formation used by a team influence its substitution patterns?
9. How do teams perform when utilizing all allowed substitutions compared to those that do not?
10. What is the impact of substitutions on a team's overall scoring in matches?

## **7. Event Analysis**

1. How do specific game events (goals, assists, cards) correlate with match outcomes?
2. What patterns can be identified in game events that lead to comebacks or late-game victories?
3. How does the timing of game events (e.g., minute of goal) influence overall match dynamics?
4. What is the relationship between player assists and team goals in matches?
5. How do game events differ across various competitions (e.g., league vs. cup)?
6. What impact do red cards have on the subsequent game events for both teams?
7. How do substitutions influence the occurrence of critical game events (e.g., goals, cards)?
8. What are the trends in goal distribution across different minutes of the match?
9. How do player events contribute to overall team statistics such as possession and passing accuracy?
10. What factors contribute to the likelihood of game events occurring in critical match periods?

## **8. Competition Analysis**

1. How does team performance vary across different competition types (e.g., league vs. cup)?
2. What is the average number of goals scored per game in each competition?
3. How do player appearances differ in league matches versus cup matches?
4. What trends can be observed in the performance of teams over multiple seasons within the same competition?
5. How do attendance figures compare across various competitions?
6. What impact do external factors (e.g., weather, referee) have on match outcomes in different competitions?
7. How do player market values fluctuate during different stages of competitions?
8. What are the most common competition types associated with player transfers?
9. How does competition type affect the frequency of yellow and red cards issued?

10. What patterns emerge in team formations across different competitions?

## 9. Player Attributes and Demographics

1. How do player attributes (height, foot, position) correlate with performance metrics such as goals and assists?
2. What demographic factors (age, country of birth) are most predictive of player market value?
3. How does the average age of players in a team impact overall performance?
4. What is the relationship between a player's physical attributes (height, foot) and their playing position?
5. How do players' demographics (country, age) differ between top-performing and lower-performing teams?
6. How does the representation of different countries affect overall league performance?
7. What trends can be identified in player demographics over multiple seasons?
8. How do player attributes influence transfer values during contract negotiations?
9. What are the average physical attributes of players in different leagues?
10. How do demographic factors impact player injury rates and recovery times?

## 10. Contract Management

1. How do contract expiration dates correlate with player performance and market value?
2. What trends can be observed in contract lengths for players across different positions?
3. How does a player's performance impact contract negotiations and renewals?
4. What factors contribute to the decision-making process for contract renewals?
5. How do the average market values of players compare to the duration of their contracts?
6. How often do players with expiring contracts outperform those with long-term contracts?
7. What is the average time players spend at a club before contract renewal or transfer?
8. How does player representation by agents affect contract negotiations and outcomes?
9. What patterns emerge in contract management practices across different clubs?
10. How do performance metrics influence the timing of contract negotiations?

These questions can serve as a starting point for a detailed analysis of the football data, providing actionable insights for decision-making in various aspects of the football business.

## 6. How these questions can be answered using visualizations or EDA?

Exploratory Data Analysis (EDA) and visualizations play a crucial role in deriving insights from complex datasets. Here's how you can use visualizations and EDA techniques to answer the business questions mentioned:

### 1. Performance Analysis:

- **Visualization:** Line charts or bar graphs showing the trend of goals, assists, and minutes played over different seasons or rounds.
- **EDA:** Descriptive statistics, such as mean and standard deviation, to understand the central tendency and variability of performance metrics.

### 2. Player Profile and Market Value:

- **Visualization:** Scatter plots to visualize the relationship between player attributes (e.g., height, market value) and performance metrics.
- **EDA:** Correlation matrices to quantify the strength and direction of relationships between variables.

### 3. Team Comparison:

- **Visualization:** Stacked bar charts or radar charts to compare team performance metrics side by side.
- **EDA:** Box plots to analyze the distribution of performance metrics for different clubs.

### 4. Attendance and Stadium Analysis:

- **Visualization:** Bar charts or heatmaps to show attendance variations across different stadiums and matches.
- **EDA:** Statistical analysis to identify stadiums with significantly higher or lower attendance.

### 5. Referee Analysis:

- **Visualization:** Pie charts or bar graphs displaying the distribution of cards given by different referees.
- **EDA:** Hypothesis testing to determine if the average number of cards given by referees is significantly different.

### 6. Substitution Patterns:

- **Visualization:** Time series plots or histograms to show the distribution of substitution timings.
- **EDA:** Statistical analysis to identify patterns in the impact of substitutions on match outcomes.

### 7. Event Analysis:

- **Visualization:** Event timelines or heatmaps to illustrate the sequence and timing of key events during matches.
- **EDA:** Cross-tabulations and chi-square tests to examine the association between events and match outcomes.

### 8. Competition Analysis:

- **Visualization:** Box plots or violin plots comparing team and player performance across different competitions.
- **EDA:** ANOVA or Kruskal-Wallis tests to assess statistical differences in performance metrics between competitions.

### 9. Player Attributes and Demographics:

- **Visualization:** Bar charts or radar charts to compare player attributes across positions or countries.
- **EDA:** Cluster analysis to identify natural groupings of players based on attributes.

#### **10. Contract Management:**

- **Visualization:** Time series plots showing the distribution of contract expiration dates.
- **EDA:** Survival analysis to model the time until contract expiration and identify patterns.

#### **2. Team Captain Influence:**

- **Visualization:** Comparative charts showing the performance of team captains vs. non-captain players.
- **EDA:** Inferential statistics to determine if there are significant differences in performance metrics between team captains and others.

For each question, it's important to choose the right visualization techniques and statistical methods that best convey the insights hidden in the data. EDA provides an opportunity to explore data patterns, identify outliers, and generate hypotheses for more in-depth analysis. Visualizations enhance the interpretability of results and make it easier to communicate findings to stakeholders.

## **Solution to the analysis one by one**

### **Business Objective 1: Performance Analysis**

#### **1. Performance Analysis:**

- Evaluate and analyze player performance based on various metrics such as goals, assists, yellow cards, and minutes played.
- Assess team performance in terms of home and away goals, club positions, and manager influence.

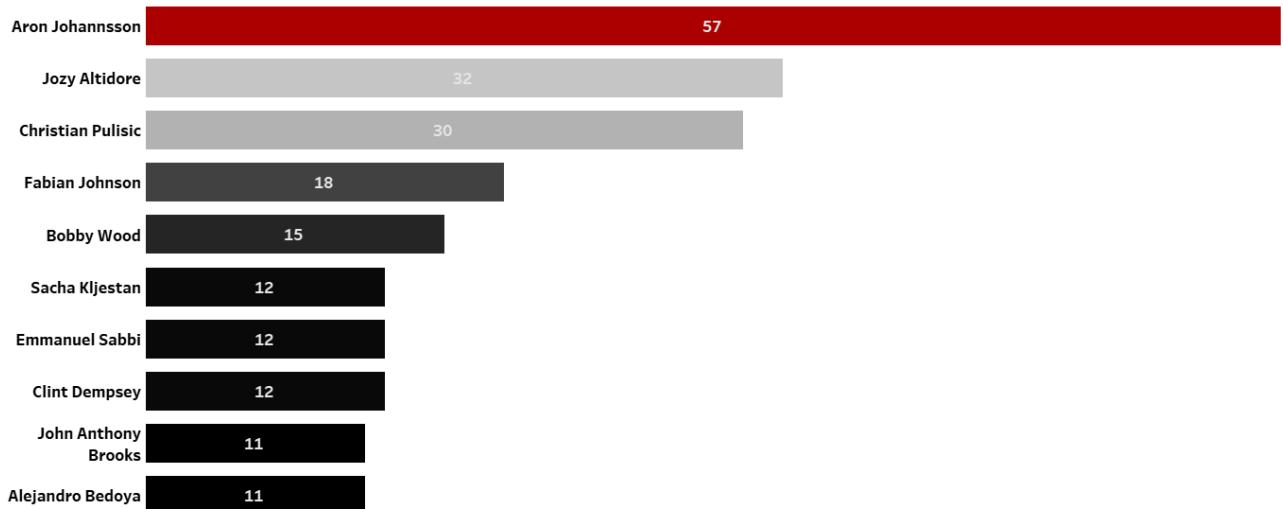
#### **Business Questions:**

- What is the average attendance at games, and can we assume that the sampling distribution of the attendance mean is approximately normal?
- What is the average market value of players, and how does the sample mean compare to the population mean?
- Can we predict whether a team will win a match based on match-related factors?

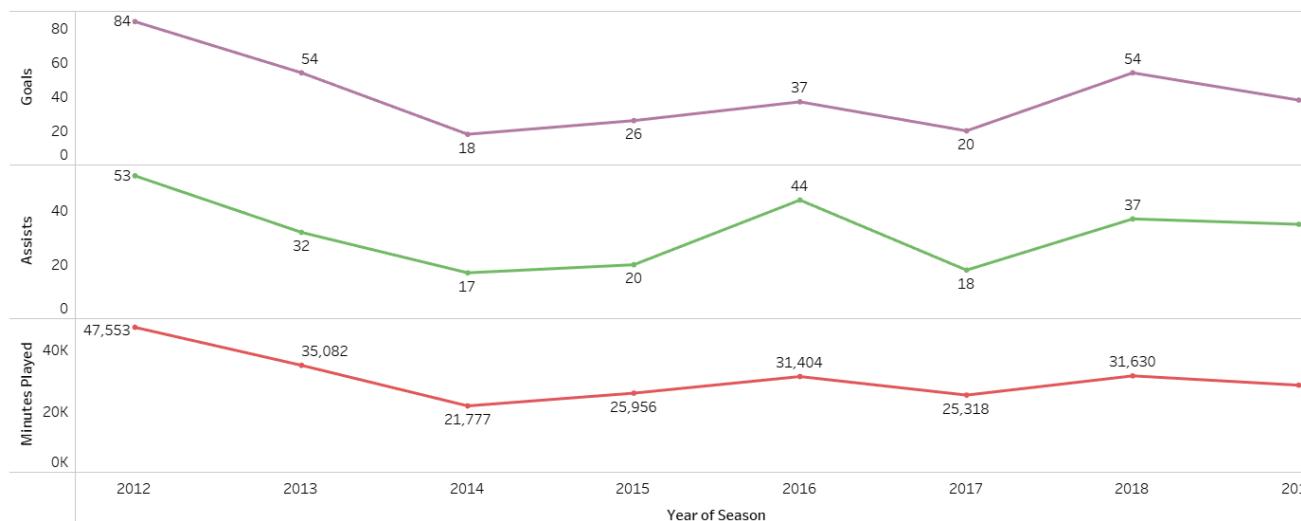
## Using Tableau:

- Evaluate and analyze player performance based on various metrics such as goals, assists, yellow cards, and minutes played.

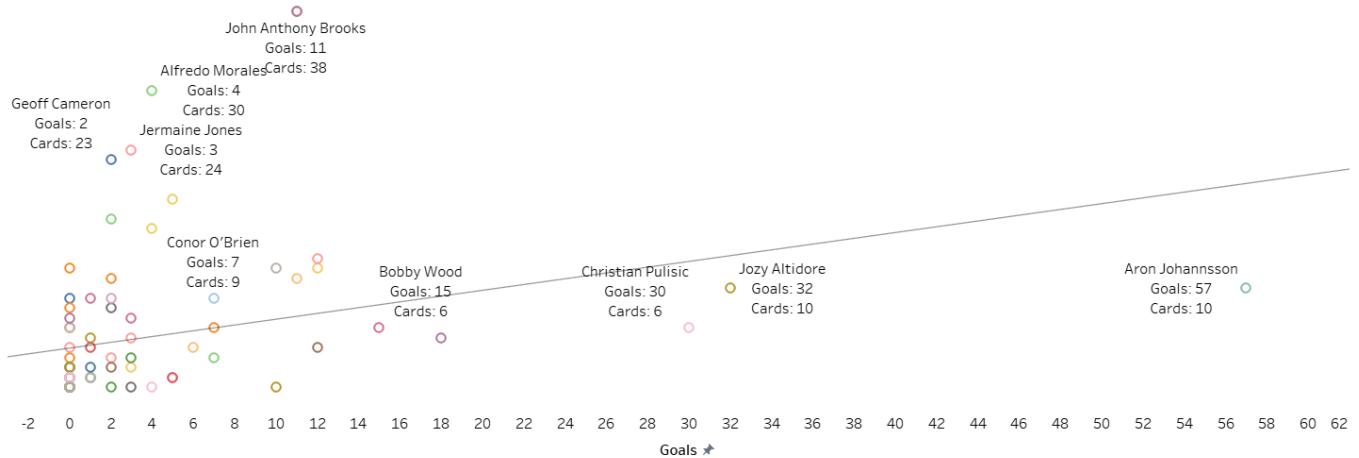
1. Player performance based on goals as follows:



2. How do player performances (goals, assists, minutes played) vary by season and competition?



3. What is the impact of cards on player performance during games .



### Using rows.com

- Assess team performance in terms of home and away goals, club positions, and manager influence.

### 1. Average home and away goals by Team:

## Average Home and Away Goals by Team

↑ ↓ ⌂ ⌂ ⌂

	A	B	C
1	home_club_name	AVG home_club_goals	AVG away_club_goals
2	1.FC Nuremberg	1.6206896552	1.2241379310
3	Hannover 96	2.4545454545	0.9090909091
4	Sport-Club Freiburg	1.55	0.7
5	Sportverein Werder Bremen von 1899	1.6804123711	1.3195876289
6	Verein fÃ/r Bewegungsspiele Stuttgart 1893	0.3	2.85
7	1. FuÃ¥ball- und Sportverein Mainz 05	0.7575757576	0.8484848485
8	Eintracht Frankfurt FuÃ¥ball AG	2.3096774194	1.3290322581
9	Borussia Verein fÃ/r LeibesÃ/bungen 1900 MÃ¶nchengladbach	1.2580645161	0
10	Verein fÃ/r LeibesÃ/bungen Wolfsburg	0.8778625954	2.4351145038
11	FC Schalke 04	1.1622807018	1.9385964912
12	TSG 1899 Hoffenheim FuÃ¥ball-Spielbetriebs GmbH	1.5185185185	1.2222222222
13	Fortuna DÃ/ sseldorf	0.5263157895	1.8771929825
14	1. FuÃ¥ball-Club KÃ¶ln	2.6060606061	2.2424242424
15	Hertha BSC	1.6521739130	1.5108695652
16	Borussia Dortmund	3.2105263158	1.6842105263
17	FC Ingolstadt 04	1.5555555556	2.5185185185
18	FC Augsburg 1907	1.9285714286	1.0714285714
19	Bayer 04 Leverkusen FuÃ¥ball	1.2678571429	2.2142857143
20	FC Bayern MÃ/nchen	5.3333333333	0.2222222222
21	Koninklijke Atletiek Associatie Gent	2.3333333333	1.1666666667
22	Yellow-Red Koninklijke Voetbalclub Mechelen	1	0
23	Koninklijke Voetbalclub Kortrijk	2	3
24	Royal Excel Mouscron (-2022)	0.5	2.5
25	KSC Lokeren (- 2020)	1.5	1.5
26	Koninklijke Voetbal Club Westerlo	0.6666666667	0.6666666667
27	Olympique Lyonnais	0.1428571429	2
28	Royal Charleroi Sporting Club	1	1
29	KV Oostende	1	0
30	Koninklijke Racing Club Genk	2	0
31	Royal Standard Club de LiÃ©ge	1	1

And so on...

## 2. Average home and away club positions by Team:

Average Home and Away Club Positions by Team     

	A	B	C
1	home_club_name	AVG home_club_position	AVG away_club_position
2	1.FC Nuremberg	13.0469348659	8.2739463602
3	Hannover 96	11.3181818182	11.5909090909
4	Sport-Club Freiburg	14.7	11.55
5	Sportverein Werder Bremen von 1	10.9419803404	6.8193079198
6	Verein fÃ/r Bewegungsspiele St	14.7	4.7
7	1. FuÃŸball- und Sportverein Mai	13.5454545455	7.6060606061
8	Eintracht Frankfurt FuÃŸball AG	8.0592648162	9.1851078154
9	Borussia Verein fÃ/r LeibesÃ/bungen Wo	9.3548387097	11.4838709677
10	Verein fÃ/r LeibesÃ/bungen Wo	10.2290076336	5.2595419847
11	FC Schalke 04	8.4796491228	9.3991228070
12	TSG 1899 Hoffenheim FuÃ Fußball-	6	6.7407407407
13	Fortuna DÃ/ sseldorf	17.0175438596	12.2105263158
14	1. FuÃ Fußball-Club KÃ¶ln	12.5454545455	5.3030303030
15	Hertha BSC	8.2554347826	11.2771739130
16	Borussia Dortmund	4.1722013656	11.2257845127

17	FC Ingolstadt 04	16.5185185185	4.0740740741
18	FC Augsburg 1907	8.5	13.2142857143
19	Bayer 04 Leverkusen FuÃŸball	7.3035714286	3.1071428571
20	FC Bayern MÃ¼nchen	1.3015873016	4.3333333333
21	Koninklijke Atletiek Associatie Ge	4.0277777778	9.5555555556
22	Yellow-Red Koninklijke Voetbalclu	11.5	5
23	Koninklijke Voetbalclub Kortrijk	5	3
24	Royal Excel Mouscron (-2022)	13.5	3
25	KSC Lokeren (- 2020)	9	4.5
26	Koninklijke Voetbal Club Westerlo	11.3333333333	7
27	Olympique Lyonnais	3.6428571429	12.6428571429
28	Royal Charleroi Sporting Club	3	6
29	KV Oostende	3	4
30	Koninklijke Racing Club Genk	9	7

And so on...

### **3. Manager Influence on Team Performance:**

## Manager Influence on Team Performance

✖️ ⏷ ⏸ 🔍 🔎

	A	B	C	D	E
1	home_club_manager_name	AVG goals	AVG assists	AVG yellow_cards	AVG red_cards
2	Dieter Hecking	0	0.1176470588	0.4901960784	0
3	Mirko Slomka	1	0	0	0
4	Christian Streich	0	0	0.3	0
5	Thomas Schaaf	0.5135135135	0	0.5135135135	0
6	Michael Wiesinger	0	0	0.2790697674	0
7	Bruno Labbadia	0.6842105263	0.1052631579	0.5438596491	0
8	Robin Dutt	0	0	1	0
9	Gertjan Verbeek	0.6	0	0.6	0
10	Thomas Tuchel	0.6129032258	0.4516129032	0.0806451613	0
11	Roger Prinzen	0	0	1	0
12	Alexander Zorniger	0	0	0.3529411765	0
13	André Schubert	0	0	0	0
14	Niko Kovac	0.0909090909	0	0.0909090909	0
15	Martin Schmidt	0.48	0	0.52	0
16	Florian Kohfeldt	0.4444444444	0	0.1805555556	0
17	David Wagner	0.1142857143	0.0095238095	0.4571428571	0
18	Alfred Schreuder	1	0	0	0
19	Adi Härtter	0.7241379310	0	0.2068965517	0
20	Uwe Rätsler	1	0	0	0
21	Markus Gisdol	0.5833333333	0.2916666667	0	0
22	Jos Luhukay	0.2857142857	0	0.2857142857	0
23	Päil Därdai	0.3291139241	0	0.1772151899	0
24	Jürgen Klopp	0.3414634146	0.6829268293	0	0
25	René Weiler	1	0	0	0
26	Markus Kauczinski	0.5185185185	0.5185185185	0	0
27	Peter Stähler	0.3594771242	0.2745098039	0	0
28	Manuel Baum	0	0	0	0
29	Friedhelm Funkel	0.6153846154	0	0.3333333333	0
30	André Breitenreiter	0	0	0.4642857143	0
31	Domenico Tedesco	0	0.1237113402	0.1237113402	0

And so on...

### Interpretations (Analysis 1):

- The total number of unique game IDs is 336.
- The average number of goals scored is 0.40.
- The maximum number of away club goals is 7.
- Players Christian (Id: 315779), Emmanuel (ID: 381187), Jordon (ID: 355369) are the top goal scorers.
- Players Christian (Id: 315779), Jonathan (ID: 542776), Jordon (ID: 355369) have the highest assists.
- Players Christian (Id: 315779) has the highest minutes played.
- Team has highest average goals under the influence of Home club manager named 'Sean Dyche'

Q. What is the probability of a player scoring a goal, given that they played at least 60 minutes in a match?

```
# Filter appearances where players played at least 60 minutes
played_60_plus = appearances[appearances['minutes_played'] >= 60]
```

```
# Calculate the number of games where the player played 60+ minutes and scored
goal_and_60_plus = played_60_plus[played_60_plus['goals'] > 0].shape[0]
```

```
# Calculate conditional probability: P(Goal | Minutes Played >= 60)
p_goal_given_60_plus = goal_and_60_plus / played_60_plus.shape[0]
```

```
print("Probability of scoring given 60+ minutes played is: ", p_goal_given_60_plus)
```

**Probability of scoring given 60+ minutes played is: 0.09471281856219095**

Q. What is the joint probability of a match having more than 2 goals and more than 30,000 spectators?

```
# Create a new column for total goals in the match
games['total_goals'] = games['home_club_goals'] + games['away_club_goals']
```

```
# Filter matches with more than 2 goals and more than 30,000 spectators
```

```

high_scoring_and_high_attendance = games[(games['total_goals'] > 2) & (games['attendance'] > 30000)]

# Calculate joint probability
p_high_scoring_and_high_attendance = high_scoring_and_high_attendance.shape[0] / games.shape[0]

print("Joint probability of a match with >2 goals and >30,000 attendance:
",p_high_scoring_and_high_attendance)

```

**Joint probability of a match with >2 goals and >30,000 attendance: 0.2347001223990208**

Q What is the average attendance at games, and can we assume that the sampling distribution of the attendance mean is approximately normal?

Solution:

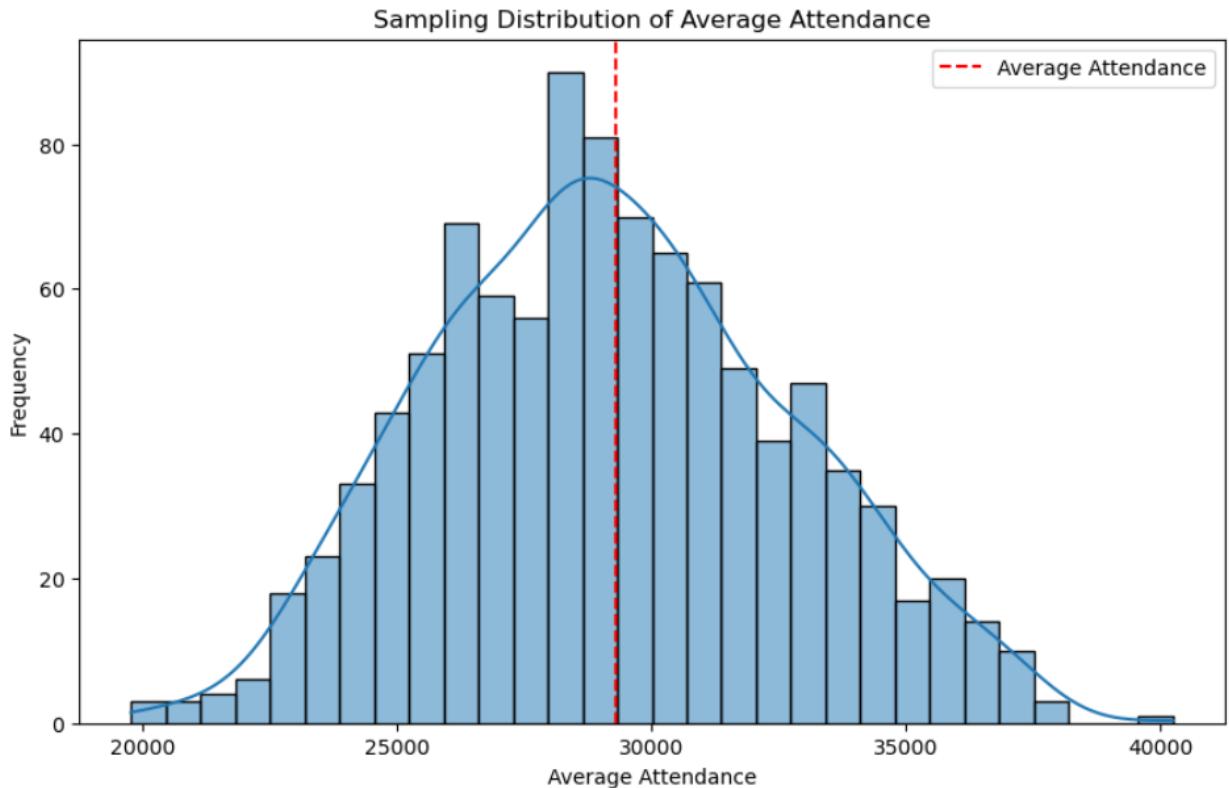
```

# Import the libraries
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from scipy import stats

# Calculate average attendance
avg_attendance = games['attendance'].mean()

# Plotting the distribution of sample means
plt.figure(figsize=(10, 6))
sns.histplot(samples, bins=30, kde=True)
plt.axvline(x=avg_attendance, color='red', linestyle='--', label='Average Attendance')
plt.title('Sampling Distribution of Average Attendance')
plt.xlabel('Average Attendance')
plt.ylabel('Frequency')
plt.legend()
plt.show()

```



Q. Can we predict if a player will receive a red card based on their performance metrics?

Predict if a player will get a red card based on:

- Minutes played
- Goals scored
- Yellow cards received

### Logistic Regression for Classification

```
# Import the libraries
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import
    mean_squared_error, mean_absolute_error, r2_score,
    confusion_matrix, classification_report
import matplotlib.pyplot as plt

# Select features and target variable
X = appearances[['minutes_played', 'goals', 'yellow_cards']]
y = appearances['red_cards']
```

```

# Scale the numerical features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
# Train-test split
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y,
test_size=0.3, random_state=42)
# Build and train logistic regression model with class weight
adjustment
model = LogisticRegression(class_weight='balanced',
random_state=42)
model.fit(X_train, y_train)
# Predictions and evaluation
y_pred = model.predict(X_test)
# Confusion Matrix and Classification Report
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test,
y_pred))
# ROC Score
from sklearn.metrics import roc_auc_score
roc = roc_auc_score(y_test, model.predict_proba(X_test)[:,1])

```

Interpretation: Model Performance: The model has an overall accuracy of 73%, but the high accuracy primarily reflects the model's effectiveness in identifying players who do not receive red cards. Classification Report: The model struggles to predict red cards (class 1), achieving a recall of 0.00 for players who do receive red cards, meaning it fails to identify red card instances. ROC-AUC Score: The ROC-AUC score of 0.693 suggests limited ability to differentiate between players who will and will not receive a red card.

## **Business Objective 2: Player Profile and Market Value**

<b>2. Player Profile and Market Value:</b>	
--	--

- Create player profiles, including personal information, playing position, and current market value.
- Analyze trends in market values and contract expiration dates for players.

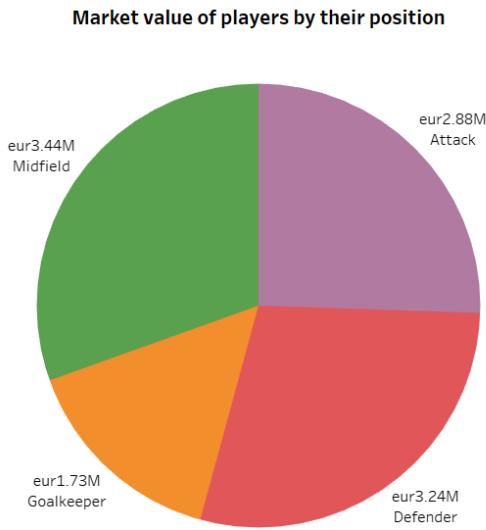
## **Business Questions:**

- What is the relationship between a player's position and their market value or highest market value?
- How do player attributes country of birth correlate with market value?

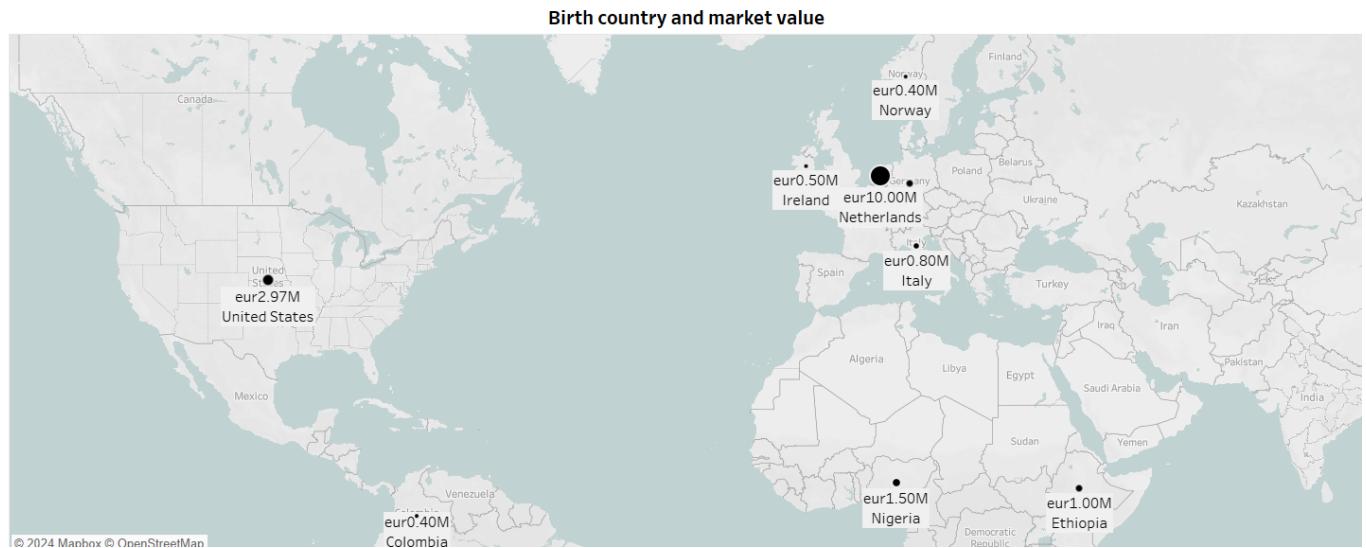
- Are there players whose market values are significantly higher or lower than expected based on their performance?

## Using Tableau:

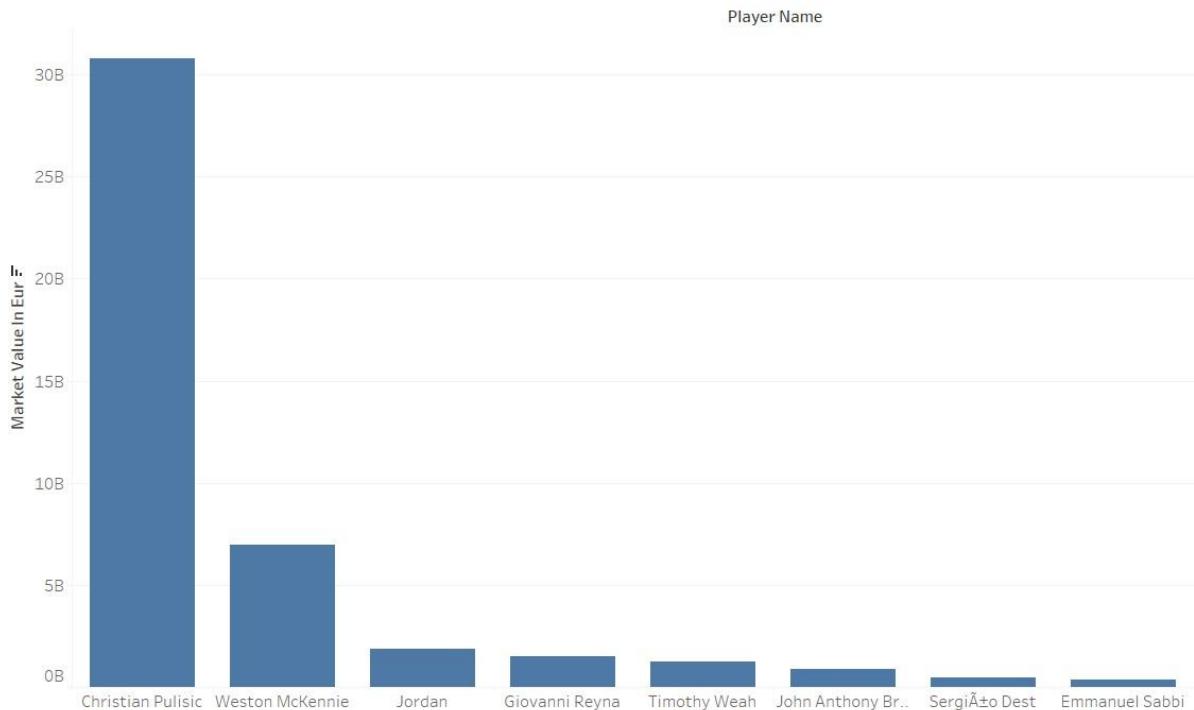
What is the relationship between a player's position and their market value or highest market value?



How do player attributes country of birth correlate with market value?



## Analyze trends in market values for players:



### Interpretations (Analysis 2):

- Midfield position has the highest avg market value. Defender have the 2nd most highest market value . then attack is at 3rd number. while Goalkeeper have comparatively less avg market value Players: Christian, Weston and Jordan have the highest market value.
- The contract expiration date with the highest market value is in the year 2027.
  - Netherlands have highest average market value(10 M)
  - and united states have average market value of 2.97 M
  - Colombia has the lowest avg market value 0.40 M

Q. Predict player's market value based on the goals scored by them using linear regression.

```
Data scaling
scaler = StandardScaler()
df1 = df
df1['goals'] = df1['goals'].astype(float)
#df1['assists'] = df1['assists'].astype(float)
df1['market_value_in_eur'] =
df1['market_value_in_eur'].astype(float)
df2 = df1[['goals','market_value_in_eur']]
cols =
df2.columns
df_scaled = scaler.fit_transform(df2)
df_scaled = pd.DataFrame(df_scaled,
columns=cols)
```

```
df_scaled.head()
```

```
X = df_scaled[['goals']] y =
df_scaled['market_value_in_eur']
```

```
scaler = StandardScaler() df1 = df df1['goals'] =  
df1['goals'].astype(float) #df1['assists'] = df1['assists'].astype(float)  
df1['market_value_in_eur'] = df1['market_value_in_eur'].astype(float)  
df2 = df1[['goals','market_value_in_eur']] cols = df2.columns  
df_scaled = scaler.fit_transform(df2) df_scaled =  
pd.DataFrame(df_scaled, columns=cols) df_scaled.head()
```

```
X = df_scaled[['goals']] y =  
df_scaled['market_value_in_eur']
```

Training model, prediction and performance evaluation

```
# Create and fit the model model  
= LinearRegression()  
model.fit(X, y)
```

```
# Predict market value predicted_market_value  
= model.predict(X)  
print(predicted_market_value)
```

```
# Performance evaluation  
MSE=mean_squared_error(y, predicted_market_value)  
MAE=mean_absolute_error(y, predicted_market_value)  
RMSE = mean_squared_error(y, predicted_market_value, squared=False)  
print(MSE, MAE, RMSE)
```

```
# R2 Score from sklearn.metrics import  
r2_score r2 = r2_score(y,  
predicted_market_value) print(f"The R2  
score of the model is", r2)
```

Q. Predict player's market value based on significant features such as height, minutes played, goals etc. using multiple linear regression.

Solution:

Use various features from the given data for the prediction of player's market value such as player's height, minutes played, goals, others and perform feature selection through Multicollinearity. Then make prediction using multiple linear regression.

```
scaler = StandardScaler() df1 = df df1['goals'] = df1['goals'].astype(float)
df1['assists'] = df1['assists'].astype(float) df1['height_in_cm'] =
df1['height_in_cm'].astype(float) df1['minutes_played'] =
df1['minutes_played'].astype(float) df1['market_value_in_eur'] =
df1['market_value_in_eur'].astype(float) df2 =
df1[['goals','market_value_in_eur','assists','height_in_cm','minutes_played']] cols =
df2.columns df_scaled = scaler.fit_transform(df2)

df_scaled = pd.DataFrame(df_scaled, columns=cols)
df_scaled.head()
```

```
X = df_scaled[['goals','assists','height_in_cm','minutes_played']] y
= df_scaled['market_value_in_eur']
```

Checking multicollinearity using heatmap and by removing features (independent variables) that are highly correlated with each other.

Which value should represent high multicollinearity?

If the cutoff value is 0.7 which means if two features have a correlation value of 0.7 or above, one of the features will be ignored.

The next question is, which feature should be ignored or selected out of the two highly correlated features?

Answer: The feature that has high correlation with the dependent variable should be selected.

```
# Selecting the cutoff value as 0.7
```

```
abs(df_scaled.corr())>=0.7
```

Checking multicollinearity for the given dataset

```
# Plotting heatmap
```

```
fig, ax = plt.subplots(1, 1, figsize=(12, 8))
```

```
ax = sns.heatmap(df_scaled.corr(), annot=True, cmap="YlGnBu")
```

Predicting player's market value using selected features and applying multiple linear regression

Selected features through Multicollinearity in previous problem -  
'goals','assists','height\_in\_cm','minutes\_played'.

Let's find correlation of these features with target label which is 'market\_value\_in\_eur'.

of features with target value.

```
# To show the correlation values of features with target value.
```

```
sorted_corr=df_scaled.corr()[['market_value_in_eur']].sort_values(by='market_value_in_eur', ascending=False) abs(sorted_corr)
```

1. One feature: 'height\_in\_cm'

It has the highest correlation value with the dependent variable.

Data Splitting, training, and predicting the model height\_in\_cm

```
X1=np.array(df_scaled['height_in_cm']).reshape((-1, 1))
```

```
y=np.array(df_scaled['market_value_in_eur'])
```

```
X_train1,X_test1,y_train1,y_test1=train_test_split(X1,y,test_size=0.3,random_state=200)
```

```
s_model1 = LinearRegression().fit(X_train1, y_train1)
```

```
s_r_sq1=s_model1.score(X_train1, y_train1) y_pred1 = s_model1.predict(X_test1)
```

```
#print('predicted response:', y_pred, sep='\n') print('Intercept: \n', s_model1.intercept_)
```

```
print('slope:', s_model1.coef_) Intercept:
```

```
0.007735824099060942
```

```
slope: [-0.37937118]
```

Evaluating the model performance

```
MSE1=mean_squared_error(y_test1, y_pred1)
MAE1=mean_absolute_error(y_test1,y_pred1)
RMSE1 = mean_squared_error(y_test1, y_pred1, squared=False)
print('MSE = ', MSE1)

print('RMSE = ', RMSE1)
print('MAE = ', MAE1)

# R2 Score r2_1 = r2_score(y_test1,
y_pred1) print(f"The R2 score of the model
is", r2_1)
```

2. Two features: 'height\_in\_cm', 'assists'

```
# Two features
X2 = df_scaled[['assists', 'height_in_cm']].values.reshape(-1,2)

#x1 = X2[:, 0]
#x2 = X2[:, 1]
```

Splitting data and training model

```
X_train2,X_test2,y_train2,y_test2=train_test_split(X2,y,test_size=0.3,random_state=200)
model_mul1 = LinearRegression().fit(X_train2, y_train2)
```

```
print('Intercept: \n', model_mul1.intercept_)
print('slope:', model_mul1.coef_) Intercept:
0.008966529307864827 slope: [
0.10945648 -0.36274408]
```

Evaluating the model performance

```
MSE2=mean_squared_error(y_test2, model_mul1.predict(X_test2))
RMSE2 = mean_squared_error(y_test2, model_mul1.predict(X_test2), squared=False)
MAE2= mean_absolute_error(y_test2, model_mul1.predict(X_test2)) print('MSE = ',
MSE2) print('RMSE = ', RMSE2) print('MAE = ', MAE2)
```

```
# R2 Score r2_2 = r2_score(y_test2,  
model_mull.predict(X_test2)) print(f"The R2 score of  
the model is", r2_2)
```

3. Three features: 'assists', 'goals', 'height\_in\_cm'  
minutes\_played  
# Three features  
X3 = df\_scaled[['assists', 'goals', 'height\_in\_cm']].values.reshape(-1,3)

Splitting data and training model

```
X_train3,X_test3,y_train3,y_test3=train_test_split(X3,y,test_size=0.3,random_state=200)  
model_mul2 = LinearRegression().fit(X_train3, y_train3)
```

```
print('Intercept: \n', model_mul2.intercept_) print('slope:',  
model_mul2.coef_)  
Intercept:  
0.009574706680227377 slope: [ 0.10196891  
0.0716885 -0.35998573]
```

Evaluating the model performance

```
MSE3=mean_squared_error(y_test3, model_mul2.predict(X_test3))  
RMSE3 = mean_squared_error(y_test3, model_mul2.predict(X_test3), squared=False)  
MAE3= mean_absolute_error(y_test3, model_mul2.predict(X_test3)) print('MSE = ',  
MSE3) print('RMSE = ', RMSE3) print('MAE = ', MAE3)  
# R2 Score r2_3 = r2_score(y_test3,  
model_mul2.predict(X_test3)) print(f"The R2 score of  
the model is", r2_3)
```

4. Four features: 'assists', 'goals', 'minutes\_played', 'height\_in\_cm'  
'global\_radiation', 'humidity', 'pressure', 'cloud\_cover'  
# Four features

```
X4 = df_scaled[['assists', 'goals', 'minutes_played', 'height_in_cm']].values.reshape(-1,4)
```

```
X_test4
```

```
X_train4,X_test4,y_train4,y_test4=train_test_split(X4,y,test_size=0.3,random_state=200)
model_mul3 = LinearRegression().fit(X_train4, y_train4)
```

```
print('Intercept: \n', model_mul3.intercept_) print('slope: ',
model_mul3.coef_)
```

```
Intercept:
```

```
0.0096372118783709 slope: [ 0.10101133  0.06873862
0.0156774 -0.35890366]
```

```
MSE4=mean_squared_error(y_test4, model_mul3.predict(X_test4))
```

```
RMSE4 = mean_squared_error(y_test4, model_mul3.predict(X_test4), squared=False)
```

```
MAE4= mean_absolute_error(y_test4, model_mul3.predict(X_test4)) print('MSE = ',
MSE4) print('RMSE = ', RMSE4) print('MAE = ', MAE4)
```

```
# R2 Score r2_4 = r2_score(y_test4,
model_mul3.predict(X_test4)) print(f"The R2 score of
the model is", r2_4)
```

```
Prediction with four features: 'assists', 'goals', 'minutes_played', 'height_in_cm'
```

```
# Predict the target value predicted_value =
model_mul3.predict(X_test4)
```

```
# Print the predicted value
print("Predicted value:", predicted_value)
```

### **Business Objective 3: Team Comparison**

**Business Objectives:**

### **3. Team Comparison:**

- Compare and contrast the performance of different clubs, managers, and teams in various seasons and rounds.
- Analyze aggregate data to identify patterns in team performance.

### **Business Questions:**

- How do different clubs and managers influence team performance in terms of goals and points?
- What are the trends in home and away performance for different teams?
- Is there a correlation between club position and specific performance metrics?

### **Using rows.com:**

- **How do different clubs and managers influence team performance in terms of goals?**

## Club and Manager Impact on Team Performance (Goals) <> ✨

	A	B	C
1	home_club_name	home_club_manager_name	SUM goals
2	Burnley Football Club	Sean Dyche	126
3	Borussia Dortmund	Lucien Favre	94
4	Fodbold Club Nordsjælland	Kasper Hjulmand	85
5	Hobro IK	Peter Sørensen	81
6	Stade Rennais Football C	Julien Stéphan	72
7	Borussia Dortmund	Thomas Tuchel	56
8	Chelsea Football Club	Frank Lampard	56
9	Hobro IK	Allan Kuhn	44
10	Eintracht Frankfurt Fußball	Adi Hütter	42
11	Borussia Dortmund	Peter Stöger	42
12	Koninklijke Atletiek Asso	Hein Vanhaezebrouck	40
13	Verein für Leibesübungen	Bruno Labbadia	39
14	Odense Boldklub	Jakob Michelsen	33
15	Sportverein Werder Bre	Florian Kohfeldt	32
16	Verein für Bewegung	Jürgen Kramny	28

17	Hamburger SV	Markus Gisdol	28
18	Borussia Dortmund	Peter Bosz	28
19	Verein fÃ / r LeibesÃ / b	Andries Jonker	28
20	Watford FC	Quique SÃinchez Flores	28
21	Arsenal Football Club	Mikel Arteta	28
22	Hertha BSC	Jos Luhukay	26
23	Hertha BSC	PÃil DÃirdai	26
24	Fortuna DÃ / sseldorf	Friedhelm Funkel	24
25	Olympique Lyonnais	Bruno GÃ©nÃ©sio	24
26	SÃ¶nderjyskE	Claus NÃrgaard	22
27	Paris Saint-Germain Foo	Thomas Tuchel	20
28	Saint Mirren Football Clu	Oran Kearney	20
29	Eintracht Frankfurt FuÃ	Thomas Schaaf	19
30	1.FC Nuremberg	Gertjan Verbeek	18

And so on...

- **What are the trends in home and away performance for different teams?**

**Sorted (descending order) by AVG home\_club\_goals:**

## Home and Away Performance by Team

✖️ ⏷ ⏸ 🔍

	A	B	C
1	home_club_name	AVG home_club_goals	AVG away_club_goals
2	Manchester City Football Club	6.6153846154	0.0769230769
3	Liverpool Football Club	6	4.5
4	FC Bayern MÃ¼nchen	5.3333333333	0.2222222222
5	Eindhovense Voetbalvereniging Ph	5	0
6	AFC Ajax Amsterdam	4.6052631579	0
7	Feyenoord Rotterdam	4	1
8	Alkmaar Zaanstreek	4	2
9	Willem II Tilburg	3.9230769231	0.1538461538
10	Paris Saint-Germain Football Club	3.5	0
11	The Celtic Football Club	3.5	0
12	Borussia Dortmund	3.2105263158	1.6842105263
13	AS Saint-Ã©tienne	3	0
14	Futbolniy Klub Dynamo Kyiv	3	1
15	Tottenham Hotspur Football Club	3	0
16	Sheffield United Football Club	3	0

And so on...

Sorted (descending order) by AVG away\_club\_goals:

## Home and Away Performance by Team

⇅ ⏷ ↑ ↴ 🔍

	A	B	C
1	home_club_name	AVG home_club_goals	AVG away_club_goals
2	SC Paderborn 07	1	5
3	Liverpool Football Club	6	4.5
4	Burnley Football Club	2	4
5	Sparta Rotterdam	1	4
6	Hamburger SV	0.66666666667	3.66666666667
7	Koninklijke Voetbalclub Kortrijk	2	3
8	Norwich City	2	3
9	Arminia Bielefeld	2	3
10	Silkeborg Idrætsforening	2	3
11	Fulham Football Club	1.66666666667	3
12	Football Club København	1	3
13	EA Guingamp	1	3
14	Sunderland AFC	0	3
15	Verein für Bewegungsspiele Stu	0.3	2.85
16	AC Horsens	2.6071428571	2.6071428571

And so on...

- Correlation between club position and performance metrics:**  
 The correlation between minutes played and home club position is 0.06.  
 The correlation between assists and home club position is -0.06  
 The correlation between goals and home club position is 0.02  
 The correlation between yellow cards and home club position is 0.09

### Interpretations (Analysis 3):

- Maximum number of goals i.e. 126 have been scored by the team under the influence of Burnley Football Club (home\_club\_name) and Sean Dyche (home\_club\_manager\_name).
- Manchester City Football Club (home\_club\_name) scored maximum average home club goals.

- SC Paderborn 07 (home\_club\_name) scored maximum average away club goals.
- There is a positive correlation of home club position with performance metrics: minutes played, goals and yellow cards while it has negative correlation with assists.

Q. Using a 5% significance level, determine whether there is a significant difference in the average goals scored by the home clubs and the away clubs. Assume sample size = 15, goals are normally distributed and the population variances are approximately equal.

Solution: Hypothesis Two - Sample t-Test

Establish the null and alternate hypothesis

H0: Difference between average goals scored by the home clubs and the away clubs is zero.

Ha: Difference between average goals scored by the home clubs and the away clubs is not equal to zero.

```
#  
# Group data by home/away team and calculate average goals  
home_avg_goals = data['home_club_goals'] away_avg_goals  
= data['away_club_goals']  
#print(home_avg_goals)  
#print(away_avg_goals)
```

Set the value of alpha ( $\alpha$ )

It is given that a 5% level of significance to be used to test hypothesis.

$\alpha = 0.05$

This test is a two-tailed test, each of the two rejection regions has an area of 0.025.

Establish the decision rule

i. If  $p\_value < \alpha$  : Rejection of Null Hypothesis (H0)

ii. If  $-t\text{-critical} > t\text{-statistic} > +t\text{-critical}$  : Rejection of Null Hypothesis(H0)

Analyze the data Calculate

sample statistic n = 15

```
home_avg_goals_sample = home_avg_goals.sample(n, random_state=1)
x1 = st.mean(home_avg_goals_sample) away_avg_goals_sample =
away_avg_goals.sample(n, random_state=1) x2 =
st.mean(away_avg_goals_sample)
v1=(st.stdev(home_avg_goals_sample))**2
v2=(st.stdev(away_avg_goals_sample))**2
```

```
n1=len(home_avg_goals_sample)
n2=len(away_avg_goals_sample) dfr= n1+n2-2
print("Sample mean of home_avg_goals =",x1)
print("Sample mean of away_avg_goals =",x2)
print("Sample variance of home_avg_goals
      =",v1) print("Sample variance of
away_avg_goals =",v2) print("Sample size of
home_avg_goals =",n1) print("Sample size of
away_avg_goals =",n2) print('Degrees of
freedom =',dfr)
```

Sample mean of home\_avg\_goals = 2.533333333333333

Sample mean of away\_avg\_goals = 1.4

Sample variance of home\_avg\_goals = 5.552380952380953

Sample variance of away\_avg\_goals = 0.6857142857142856

Sample size of home\_avg\_goals = 15

Sample size of away\_avg\_goals = 15

Degrees of freedom = 28

Calculate t-statistic and t-critical

r

```
signal = abs(x1-x2) noise = math.sqrt(v1*(n1-1) + v2*(n2-  
1))*math.sqrt(1/n1 + 1/n2) t_statistic=signal/noise  
print('t-statistic =',t_statistic)
```

```
p_value=t.sf(abs(t_statistic),dfr)*2 print("The  
p_value is ", p_value)
```

```
t_critical = t.ppf(1-0.025, dfr) print('t-  
critical =',t_critical)
```

## **Business Objective 4: Attendance and Stadium Analysis**

### **Business Objectives:**

#### **4. Attendance and Stadium Analysis:**

- Evaluate attendance trends and analyze the influence of factors such as stadiums and referees on attendance.
- Assess the impact of stadium capacity on home club goals.

### **Business Questions:**

- How does stadium capacity affect attendance and home club goals?
- Are there specific stadiums that consistently attract higher attendance?
- Does the choice of stadium impact the overall match experience for fans?

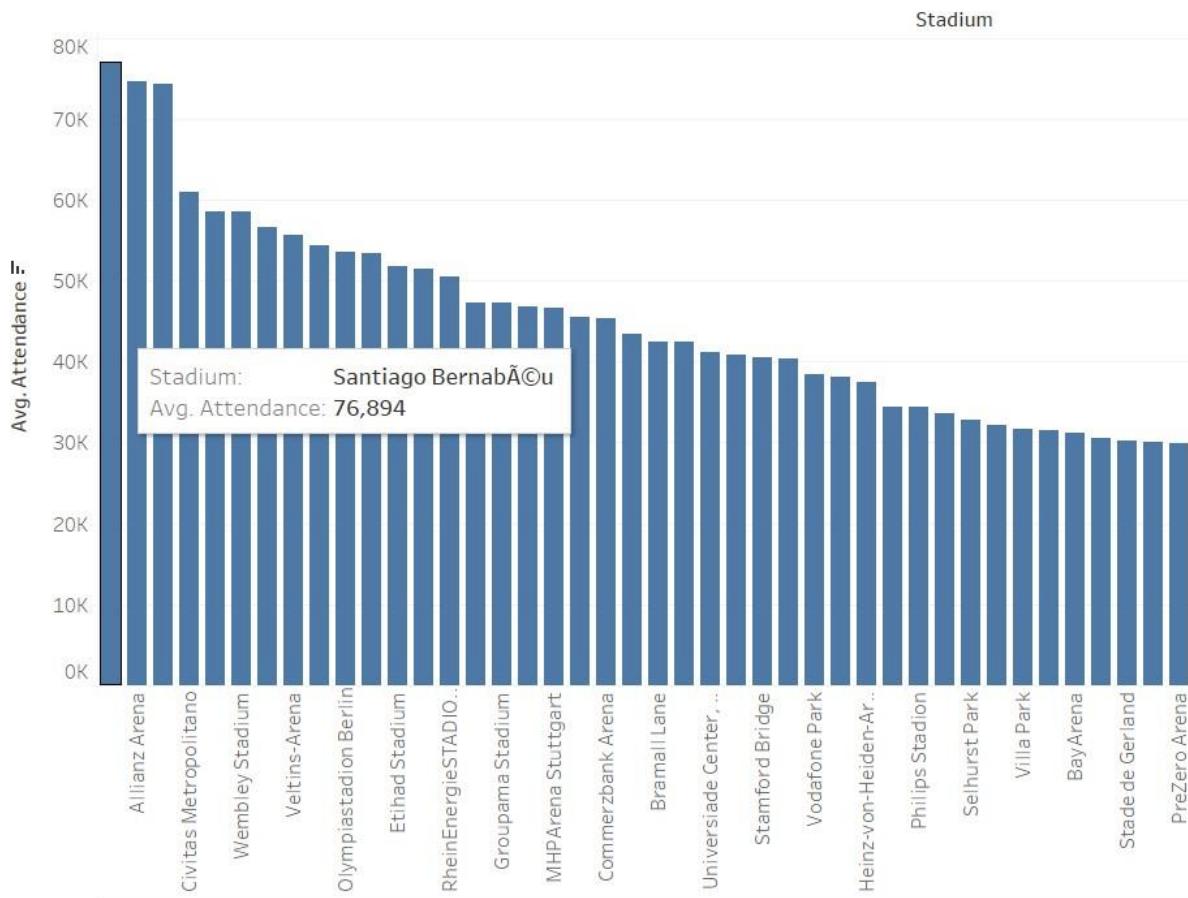
- **Stadium effect on attendance:**

**Using rows.com and Tableau both:**

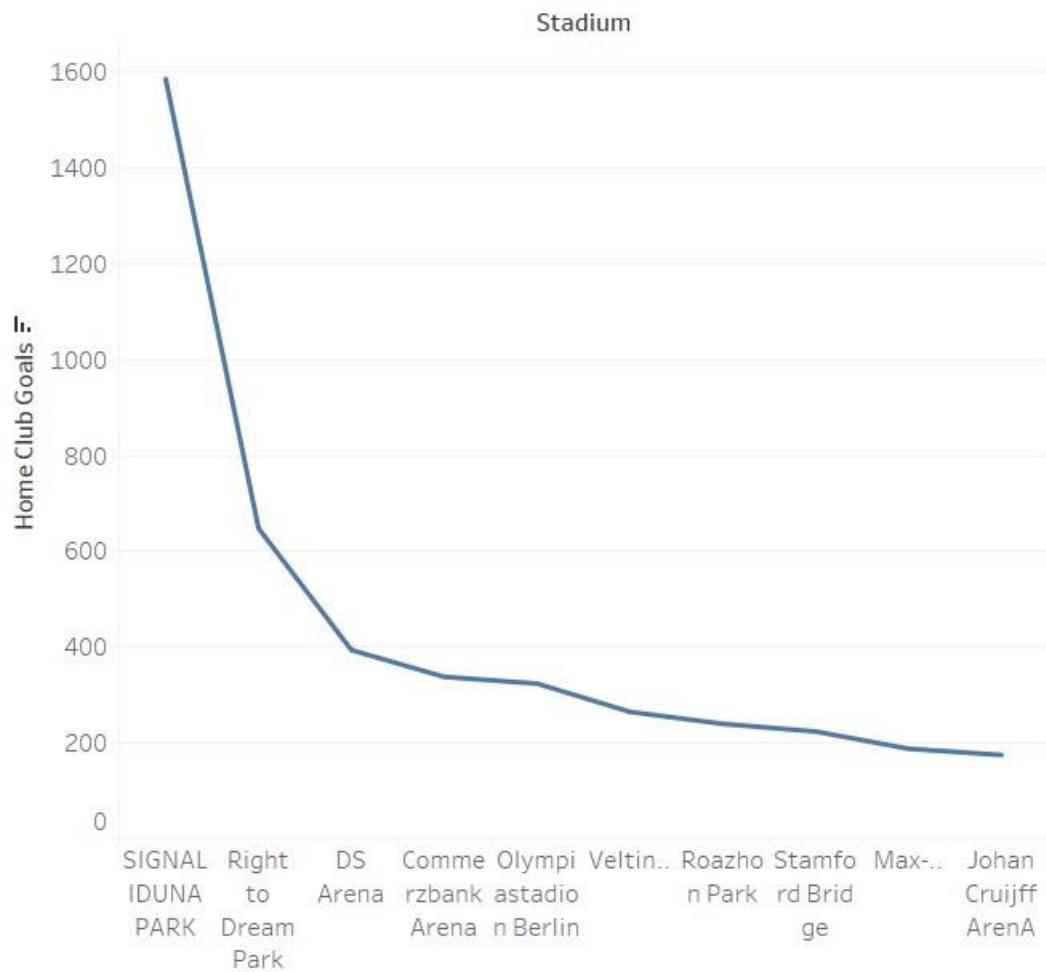
## Stadium Capacity and Attendance

	A	B
1	stadium	AVG attendance
2	Santiago BernabÃ©u	76894
3	Allianz Arena	74666.6666666667
4	SIGNAL IDUNA PARK	74249.1862348178
5	Civitas Metropolitano	61023
6	Mercedes-Benz Arena	58605.0588235294
7	Wembley Stadium	58494.8451612903
8	Celtic Park	56542.5
9	Veltins-Arena	55568.4343902790
10	Volksparkstadion	54308
11	Olympiastadion Berlin	53537.7365853659
12	Johan Cruijff ArenA	53419.7631578947
13	Etihad Stadium	51736.7692307692
14	Stadion im Borussia-Park	51409.9677419355
15	RheinEnergieSTADION	50502.3413078150
16	Parc des Princes	47289

And so on..



- **Stadium effect on home club goals?**



## Stadium and Number of Matches

	A	B	+
1	stadium	COUNT stadium	
2	SIGNAL IDUNA PARK	494	
3	DS Arena	280	
4	Veltins-Arena	228	
5	Right to Dream Park	225	
6	Olympiastadion Berlin	205	
7	Stamford Bridge	140	
8	Commerzbank Arena	135	
9	Volkswagen Arena	131	
10	Max-Morlock-Stadion	116	
11	Wohninvest-Weserstadion	97	
12	Roazhon Park	84	
13	Nature Energy Park	77	
14	Ghelamco Arena	72	
15	MERKUR SPIEL-ARENA	57	
16	BayArena	56	

- Are there specific stadiums that consistently attract higher attendance?

*Answer:* Yes, The stadiums such as ‘Santiago BernabAcu’, ‘Allianz Arena’, ‘Signal Iduna Park’ have very high average attendance as compared to other stadiums.

### Interpretations (Analysis 4):

- The stadium ‘Santiago BernabAcu’ has maximum average attendance, i.e. 76894 and stadium ‘HelsingA r Stadion’ has minimum average attendance, i.e. 1119.
- Specific stadiums consistently attract higher attendance. Example, the stadiums such as ‘Santiago BernabAcu’, ‘Allianz Arena’, ‘Signal Iduna Park’ have very high average attendance as compared to other stadiums.
- The stadium ‘Signal Iduna Park’ has maximum total number of home club goals and maximum number of matches.

Q. Categorize matches into high, moderate and low attendance based on match characteristics and other significant factors. Solution: KNN Classification

```
df_temp = df df_temp[df_temp["attendance"] < 20000] = 0  
df_temp[(df_temp['attendance'] >= 20000) & (df['attendance'] < 45000)] = 1  
df_temp[df_temp["attendance"] >= 45000] = 2
```

```
df_temp["attendance"]
```

```
# label_encoder object knows how to understand word labels.
```

```
label_encoder = preprocessing.LabelEncoder()
```

```
# Encode labels comp_encode=
```

```
label_encoder.fit_transform(df_temp[['competition_type']])  
print(comp_encode) sea_encode=  
label_encoder.fit_transform(df_temp[['season']]) print(comp_encode)  
print(label_encoder.classes_)
```

```
# Define features and target variable
```

```
X = df_temp[['goals', 'competition_type', 'season']] y =  
df_temp["attendance"] # High, moderate or low attendance
```

```
from sklearn.model_selection import train_test_split X_train,  
X_test, y_train, y_test= train_test_split(X, y,  
test_size= 0.2,  
shuffle= True, #shuffle the data to avoid bias  
random_state= 0)
```

```
y_test
```

Finding Optimum K value

```
from sklearn import neighbors
```

```

error_rate = [] for
i in range(1,5):
    clf_food = neighbors.KNeighborsClassifier(n_neighbors=i)
    clf_food.fit(X_train,y_train)    pred_i =
    clf_food.predict(X_test)
    error_rate.append(np.mean(pred_i != y_test))

plt.figure(figsize=(10,6))
plt.plot(range(1,5),error_rate,color='blue', linestyle='dashed',
marker='o',markerfacecolor='red', markersize=10)
plt.title('Error Rate vs. K Value') plt.xlabel('K') plt.ylabel('Error Rate') print("Minimum
error:-",min(error_rate),"at K =",error_rate.index(min(error_rate))+1)

```

n=3922

```

k=round(np.sqrt(10),0).astype(int)
print("K = ", k)
Building Model and performance evaluation
y

```

```

from sklearn import neighbors

# learning the classifier clf =
neighbors.KNeighborsClassifier(k)
clf.fit(X, y)
KNeighborsClassifier(n_neighbors=3)
# Model Properties
print(clf.classes_)
print(clf.n_neighbors)

# Using the trained model to predict the outcome of the X_Test dataset
y_pred = clf.predict(X_test) y_pred[1:10]

```

```

# average : string, [None, 'binary' (default), 'micro', 'macro', 'weighted']

# This parameter is required for multiclass/multilabel targets. If None, the scores for each
# class are returned.

from sklearn.metrics import recall_score
from sklearn.metrics import f1_score from
sklearn.metrics import accuracy_score from
sklearn.metrics import precision_score

print('accuracy:', accuracy_score(y_test, y_pred)) print('recall:',

recall_score(y_test, y_pred, average='micro')) print('f1-score:',

f1_score(y_test, y_pred, average='micro')) print('precision:',

precision_score(y_test, y_pred, average='micro'))

```

## **Business Objective 5: Referee Analysis**

### **Business Objectives:**

#### **5. Referee Analysis:**

- Evaluate referee performance and its potential impact on match outcomes.
- Analyze the distribution of cards and goals in matches officiated by different referees.

### **Business Questions:**

- What is the correlation between referee decisions (cards) and match outcomes?
- Are there referees associated with more controversial or high-scoring matches?

- **Correlation between referee decisions (cards) and match outcomes: Using rows.com, Python etc.**

In football (soccer), **yellow cards and red cards** are disciplinary actions issued by the match referee to players who commit fouls or engage in unsporting behavior. These cards are used to maintain discipline on the field and ensure fair play.

A yellow card is a caution or warning given to a player for various reasons and A red card is a more severe punishment and results in the player being sent off the field.

There is no Red card in cleaned data.

## Yellow Cards and Match Outcome

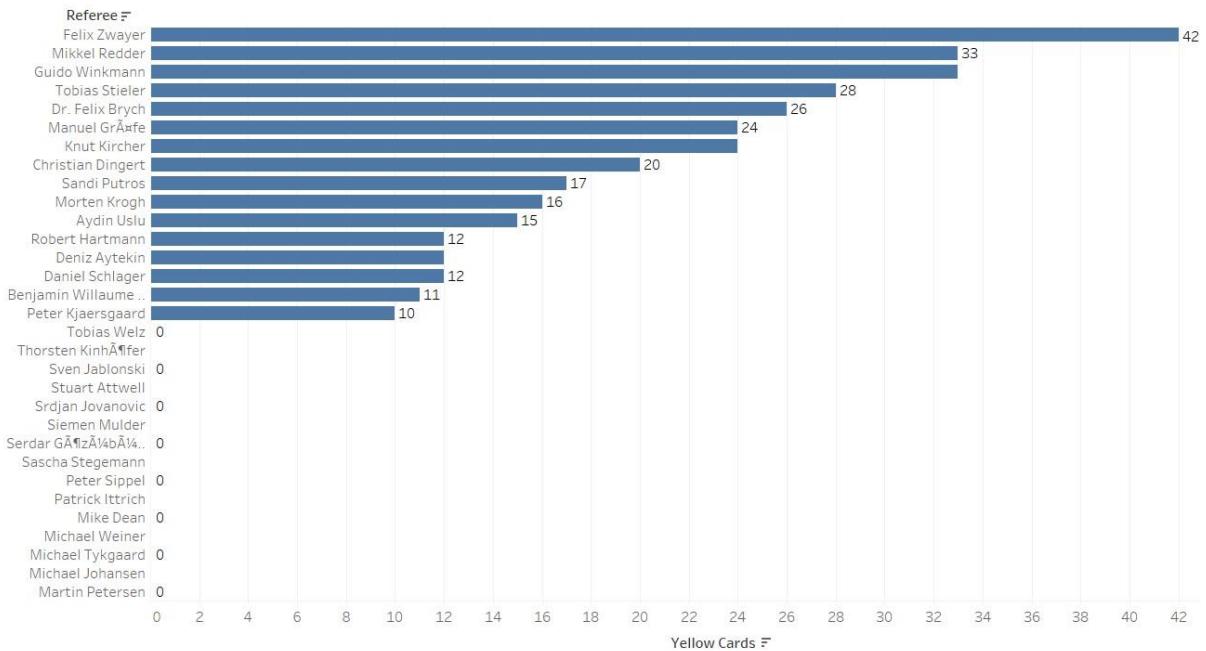
	A	B	C	
1	yellow_cards	home_club_goals	away_club_goals	
2	1	3	0	
3	1	3	0	
4	1	3	0	
5	1	3	0	
6	1	3	0	
7	1	3	0	
8	1	1	0	
9	1	1	0	
10	1	1	0	
11	1	1	0	
12	1	1	0	
13	1	1	0	
14	1	1	1	
15	1	1	1	
16	1	1	1	

Python Result:

	game_id	player_id	yellow_cards
game_id	1.000000	0.516769	-0.102141
player_id	0.516769	1.000000	-0.094057
yellow_cards	-0.102141	-0.094057	1.000000
red_cards	NaN	NaN	NaN
goals	0.122913	0.043862	0.031877
assists	0.036331	0.165656	-0.094127
minutes_played	0.095915	-0.046593	0.121012
season	0.990047	0.531405	-0.111281
home_club_goals	-0.048219	0.050035	-0.069751
away_club_goals	0.001400	-0.023951	-0.005039

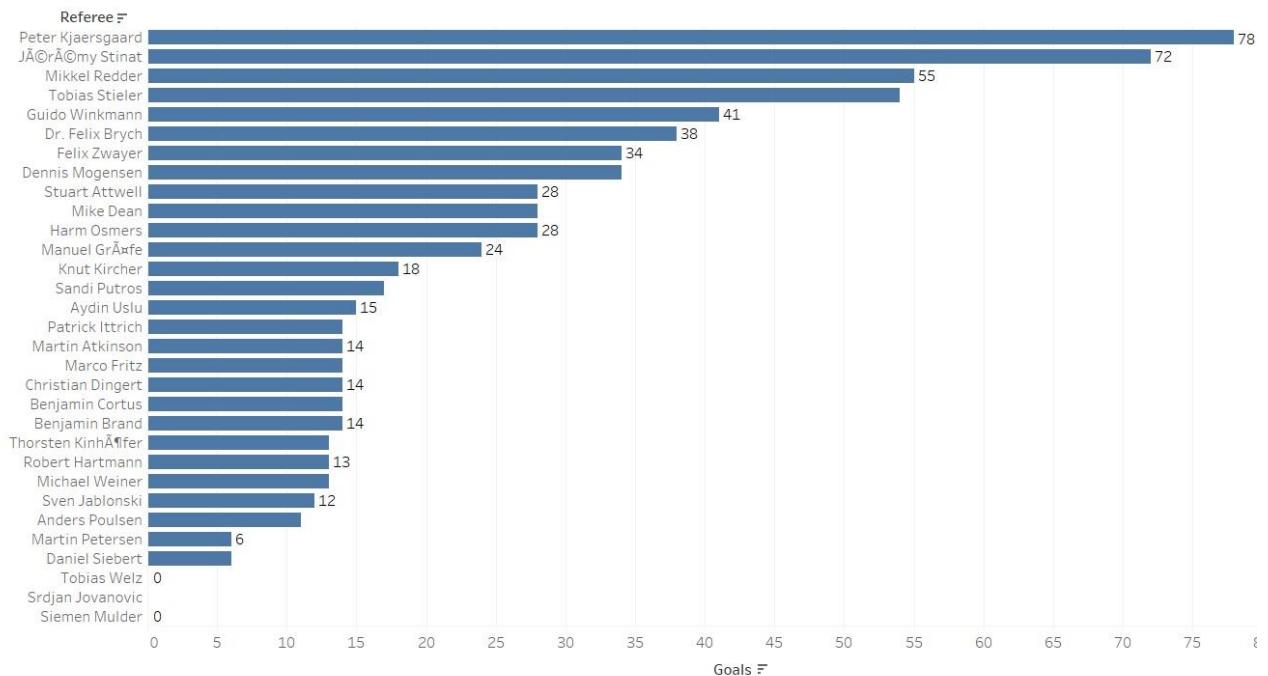
Yellow cards has correlation with ‘goals’ = 0.03, with ‘home\_club\_goals’ = -0.069 and with ‘away\_club\_goals’ = -0.005. This shows yellow cards have not much impact on match outcomes (goals).

- Distribution of cards in matches officiated by different referees.



Maximum yellow cards in matches officiated by referees ‘Felix’, ‘Mikkel’, and ‘Guido’.

- Distribution of goals in matches officiated by different referees.



Maximum goals scored in matches officiated by referees ‘Peter Kjaersgaard’ and ‘Jacracmy Stinat’.

- Are there referees associated with more controversial or high-scoring matches?

Answer: Yes. As we can see from above graphs that referees ‘Peter Kjaersgaard’ and ‘Jacracmy Stinat’ are associated with maximum goals scored in matches while referees ‘Felix’, ‘Mikkel’, and ‘Guido’ are associated with maximum yellow cards in matches.

### Interpretations (Analysis 5):

- Yellow cards has correlation with ‘goals’ = 0.03, with ‘home\_club\_goals’ = -0.069 and with ‘away\_club\_goals’ = -0.005. This shows yellow cards have not much impact on match outcomes (goals).
- Maximum goals scored in matches officiated by referees ‘Peter Kjaersgaard’ and ‘Jacracmy Stinat’.
- Maximum yellow cards in matches officiated by referees ‘Felix’, ‘Mikkel’, and ‘Guido’.
- Few referees (mentioned above) are associated with controversial and few referees (mentioned above) are associated with high-scoring matches.

### Business Objective 6: Substitution Patterns

#### **Business Objectives:**

- |                                  |  |
|----------------------------------|--|
| <b>6. Substitution Patterns:</b> |  |
|----------------------------------|--|

- Study substitution patterns, including the timing and frequency of player substitutions.
- Assess the impact of substitutions on match outcomes.

## Business Questions:

What is the impact of player substitutions on match outcomes?

- Are there specific patterns in the timing and frequency of substitutions across different teams?
- Do certain managers exhibit more effective substitution strategies?

Q1 : What is the impact of player substitutions on match outcomes?

Using Rows -

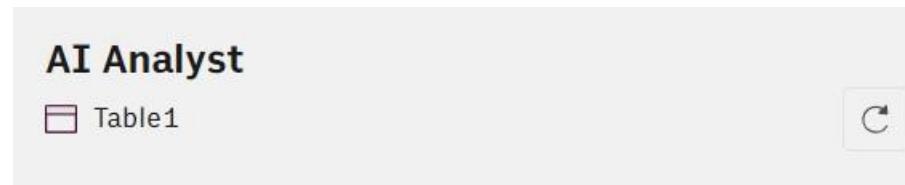
Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	70
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.



## Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**What is the impact of player substitutions on match outcomes?**

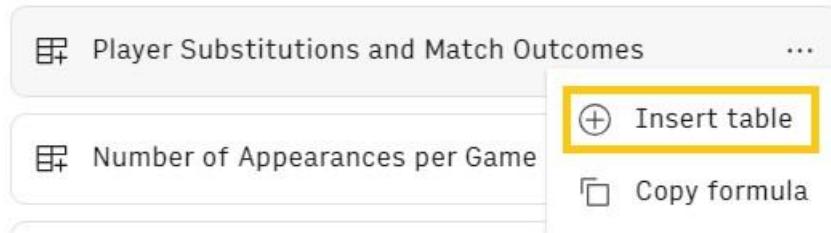
Click on enter option.

What is the impact of player substitutions on match outcomes?

Step 2 :

In deep dives you can see one new table is created with name Player Substitutions and Match Outcomes. Click on insert table option.

### Deep dives



Step 3 –

Below dataset we can see Player Substitutions and Match Outcomes Table.

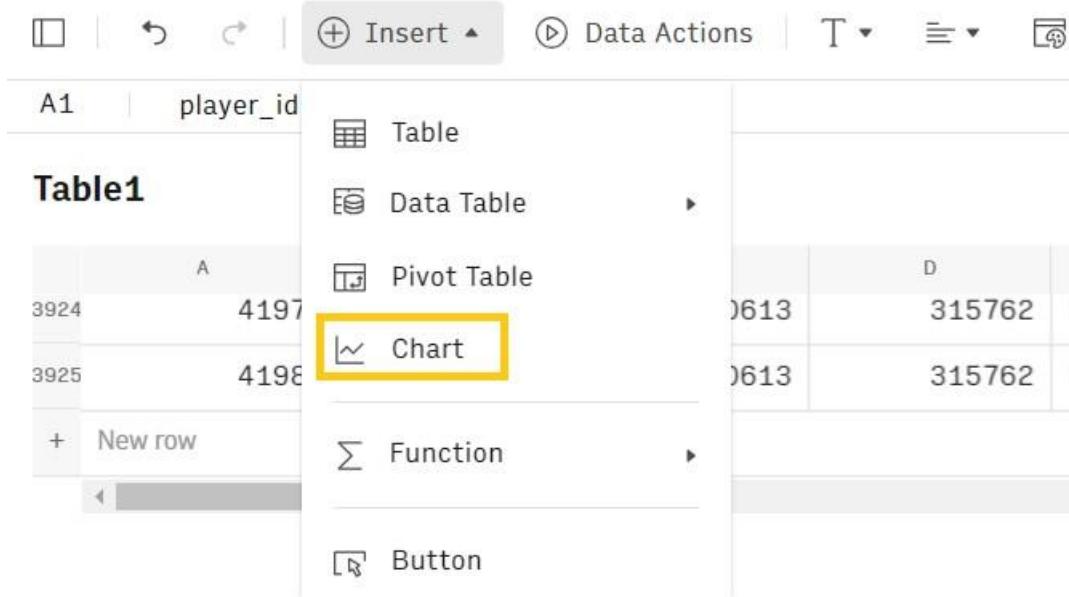
### Player Substitutions and Match Outcomes

	A	B	C	D	E	F	+
1	player_id	player_in_id	home_club_goa	away_club_goa	home_club_na	away_club_na	
2	49723	60957		1		2	1.FC Nurember
3	49723	60957		1		2	1.FC Nurember
4	49723	60957		1		2	1.FC Nurember
5	49723	60957		1		2	1.FC Nurember
6	49723	60957		1		2	1.FC Nurember

Step 4 – Filter data for player\_id 49723 by selecting sort option.

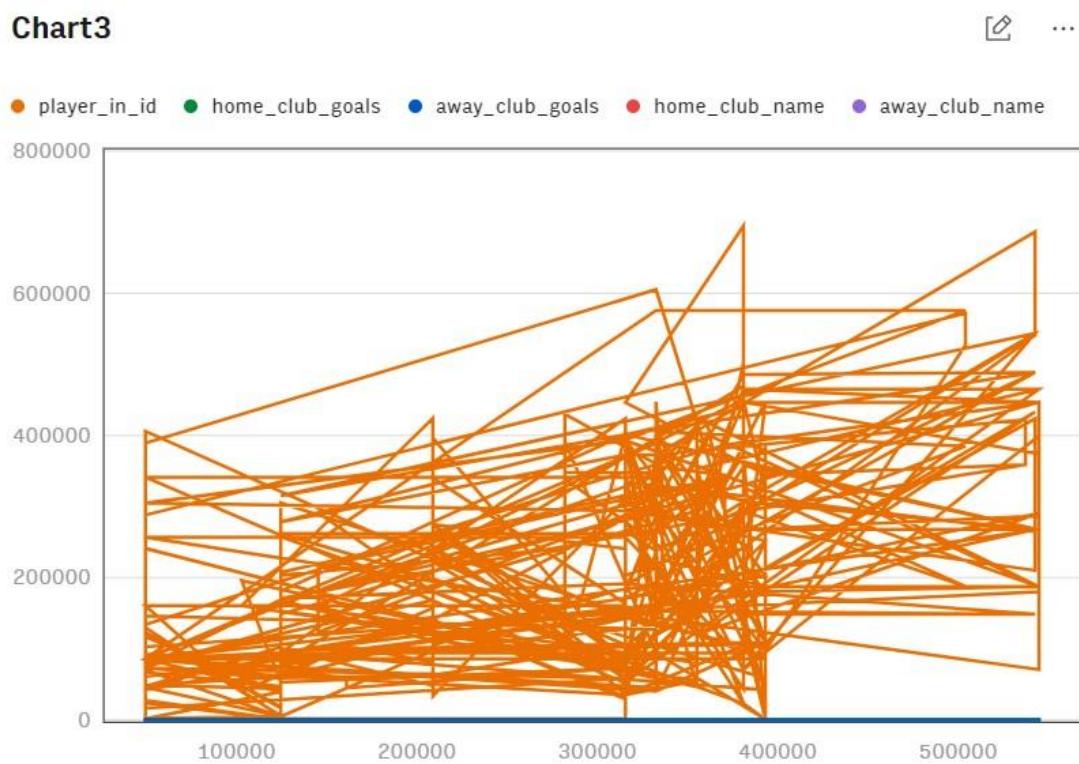
	A	B	C	D	E	F	+
1	player_id	player_in_id	home_club_goa	away_club_goa	home_club_na	away_club_na	
2	49723	60957		1		2	1.FC Nurember
3	49723	60957		1		2	1.FC Nurember
4	49723	60957		1		2	1.FC Nurember
5	49723	60957		1		2	1.FC Nurember

Step 5 - Select table. Click on insert option and select chart.

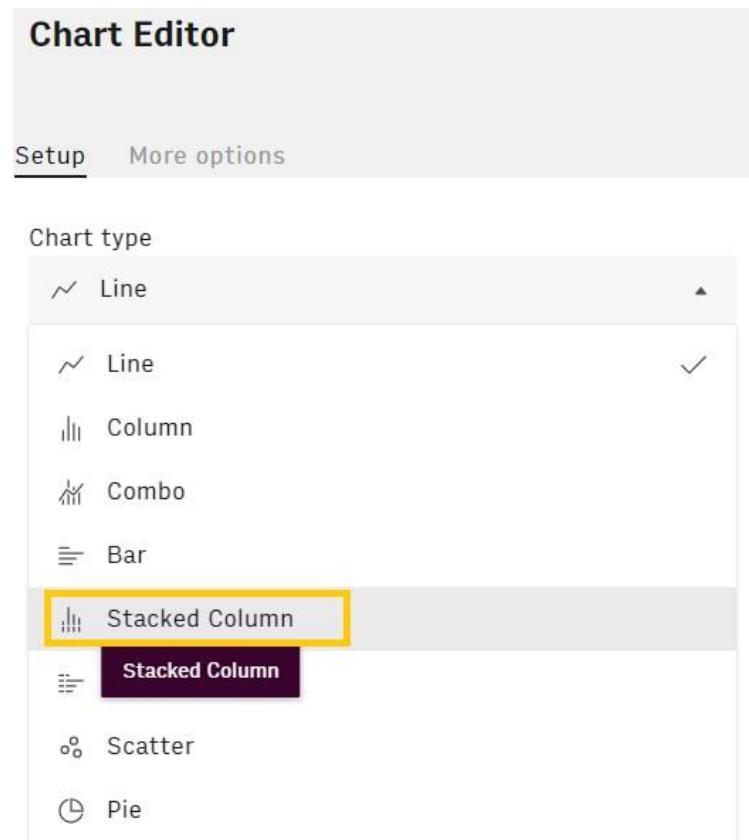


Note - Screenshot is taken with only first 6 options in insert menu.

The line chart will create as below –



Step 6 - Edit this line chart from Chart Editor option. Select Chart type as Stacked column.



Step 7 – Put in main axis as Player\_in\_id. Delete home\_club\_name and away\_club\_name.

Also vertical left axis put only player\_id,home\_club\_goals and away\_club\_goals.

#### Main axis

player\_id

#### Vertical left axis

player\_in\_id



home\_club\_goals



away\_club\_goals



home\_club\_name

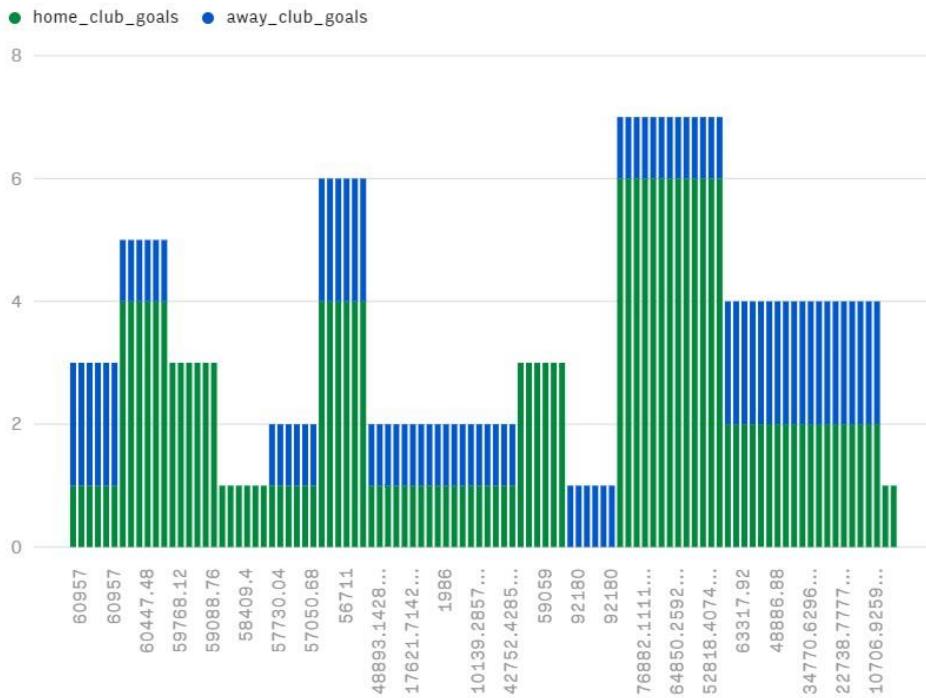


away\_club\_name



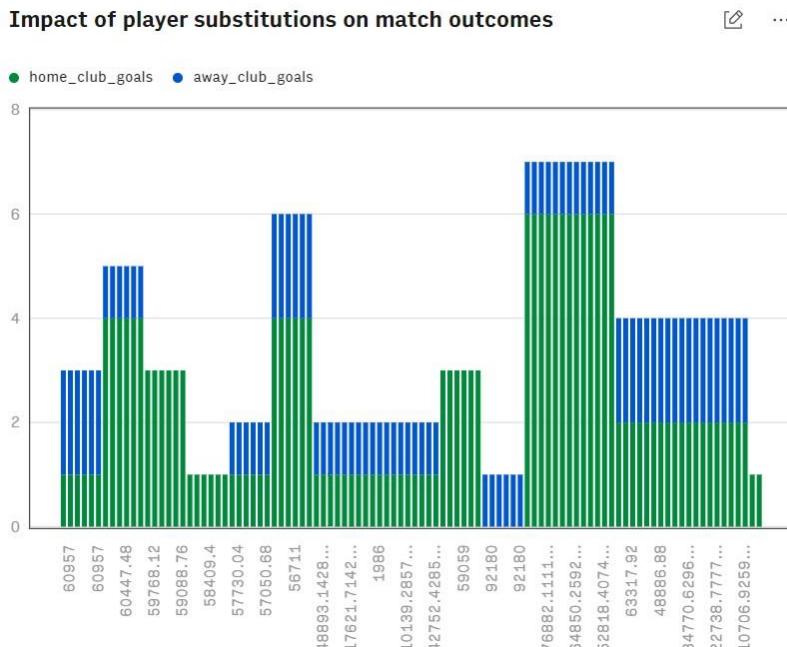
You can able to see the stacked bar as –

## Chart2



Step 8 – By double clicking on title chart 2, can edit title as **Impact of player substitutions on match outcomes**.

Result –



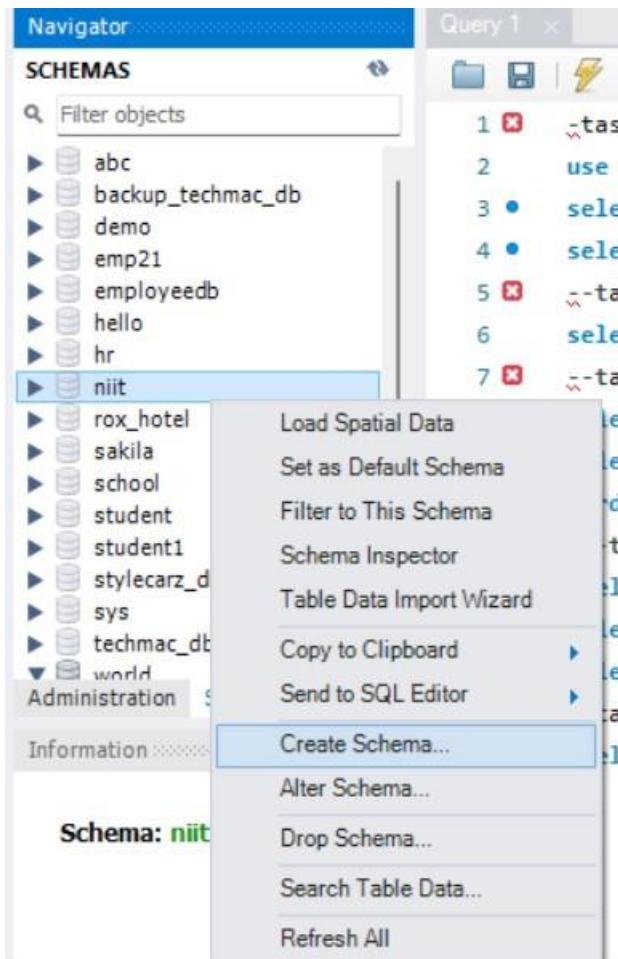
Interpretation –

From obtained result we can see that if player is replaced then replacement is impacting on goal.

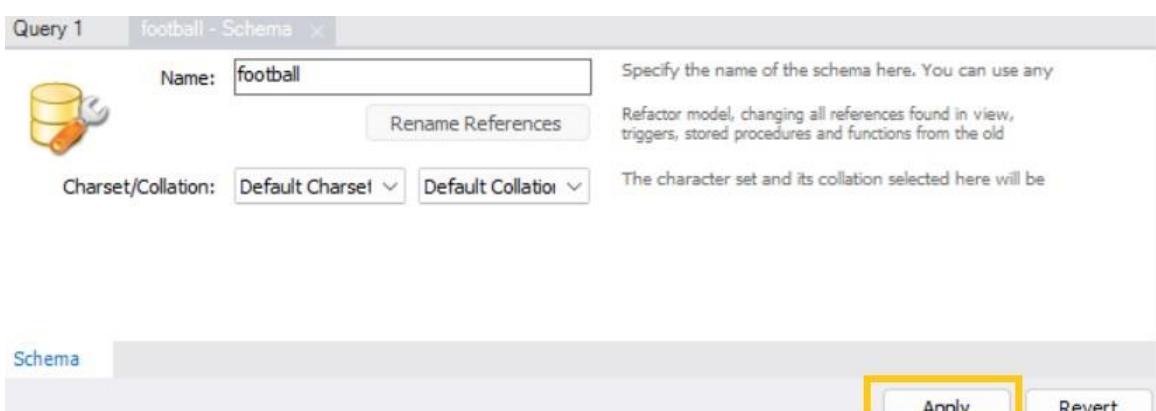
Using SQL –

Step 1 –

Right click on any schema and select Create Schema option.

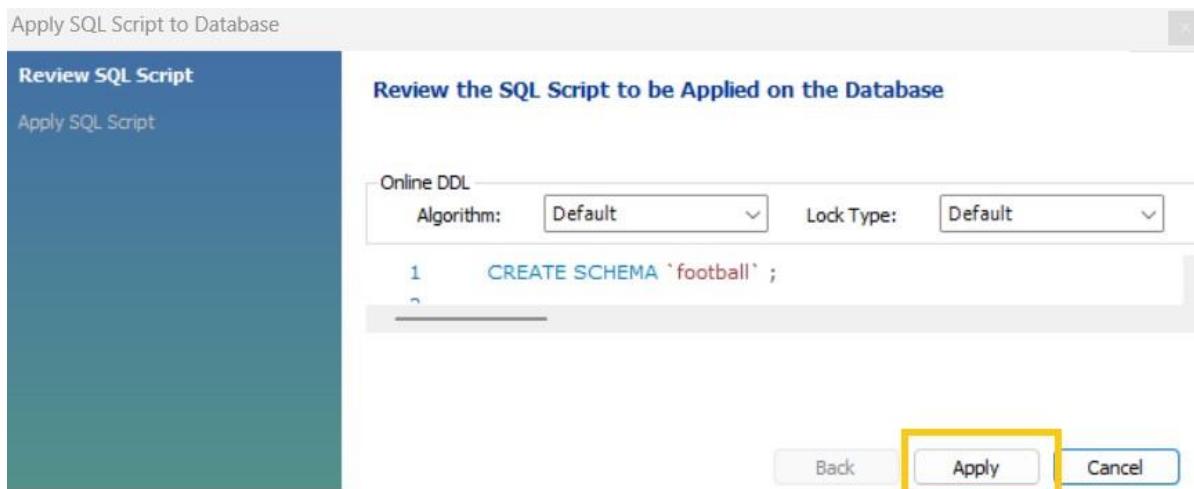


Create schema with name football and click on apply.



Step 3 –

Click on apply in Apply SQL Script database window.



Step 4 –

Click on finish. You can see the football schema is created.

- ▶ abc
- ▶ backup\_techmac\_db
- ▶ demo
- ▶ emp21
- ▶ employeedb
- ▶ **football**
- ▶ hello
- ▶ hr
- ▶ niit
- ▶ rox\_hotel
- ▶ sakila
- ▶ school
- ▶ student
- ▶ student1
- ▶ stylecarz\_db

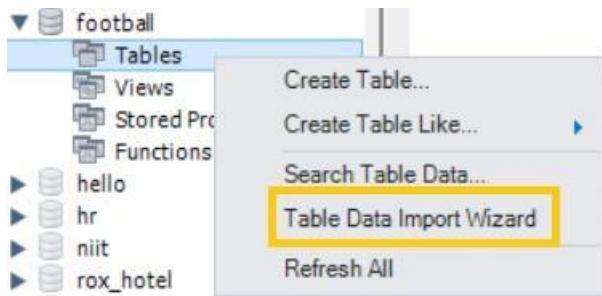
Step 5 –

Open Schema by clicking on triangle as –

- ▼ **football**
  - Tables
  - Views
  - Stored Procedures**
  - Functions

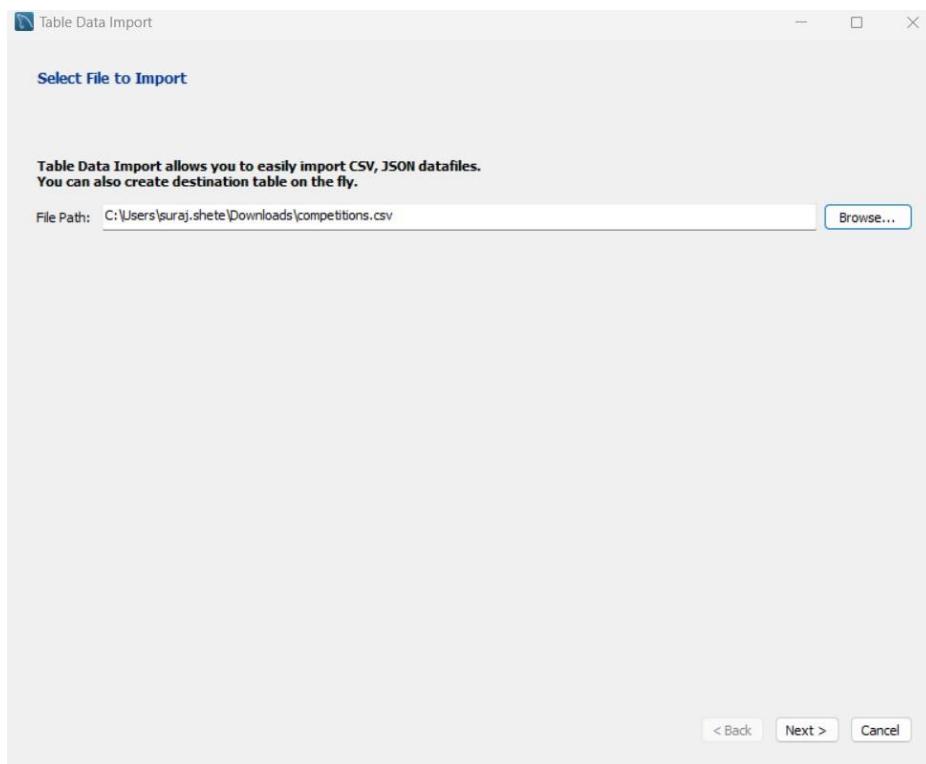
Step 6 –

Right click on Tables option and select Table Data Import Wizard.



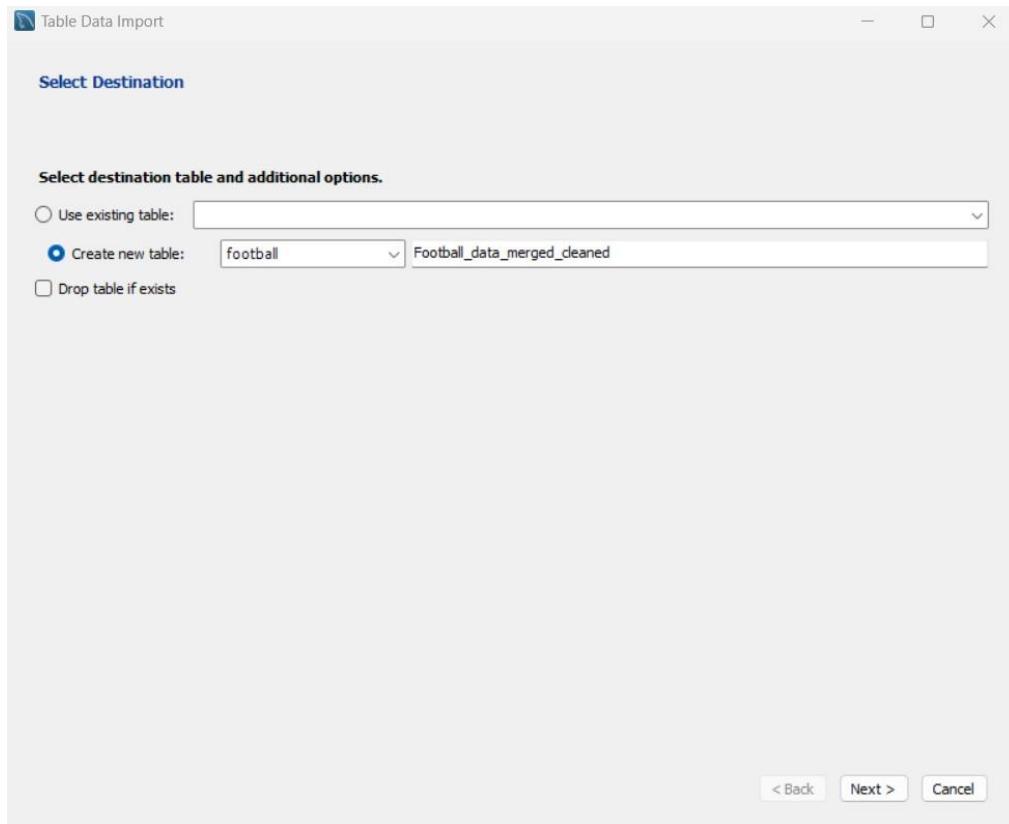
### Step 7 –

Browse the football data file and click on next.



### Step 8 –

Select new table option and click on Next more two times.



Your football data is successfully uploaded in MySQL.

Step 9 - Open one new query editor. Write following query.

Using Python –

Step 1 –

Open Jupyter notebook and read football\_data\_merged\_cleaned file using pandas library.

Run the following code in jupyter Notebook – import pandas as pd  
import matplotlib.pyplot as plt

```
# Assume df is your DataFrame containing football match summary data
```

```
# Identify matches with substitutions matches_with_subs = df[df['type_x'] == 'Substitutions']['game_id'].unique()
```

```

# Create a new column indicating whether a match had substitutions or not
df['has_substitutions'] = df['game_id'].isin(matches_with_subs)

# Calculate average goals for matches with and without substitutions avg_goals_with_subs
= df[df['has_substitutions']]['home_club_goals'].mean() avg_goals_without_subs =
df[~df['has_substitutions']]['home_club_goals'].mean()

# Calculate average goals conceded for matches with and without substitutions
avg_goals_conceded_with_subs = df[df['has_substitutions']]['away_club_goals'].mean()
avg_goals_conceded_without_subs = df[~df['has_substitutions']]['away_club_goals'].mean()

# Create a bar chart to visualize the impact labels = ['Matches with
Substitutions', 'Matches without Substitutions'] goals_data =
[avg_goals_with_subs, avg_goals_without_subs]
goals_conceded_data = [avg_goals_conceded_with_subs,
avg_goals_conceded_without_subs]

fig, ax = plt.subplots(figsize=(10, 6))
bar_width = 0.35 opacity
= 0.8

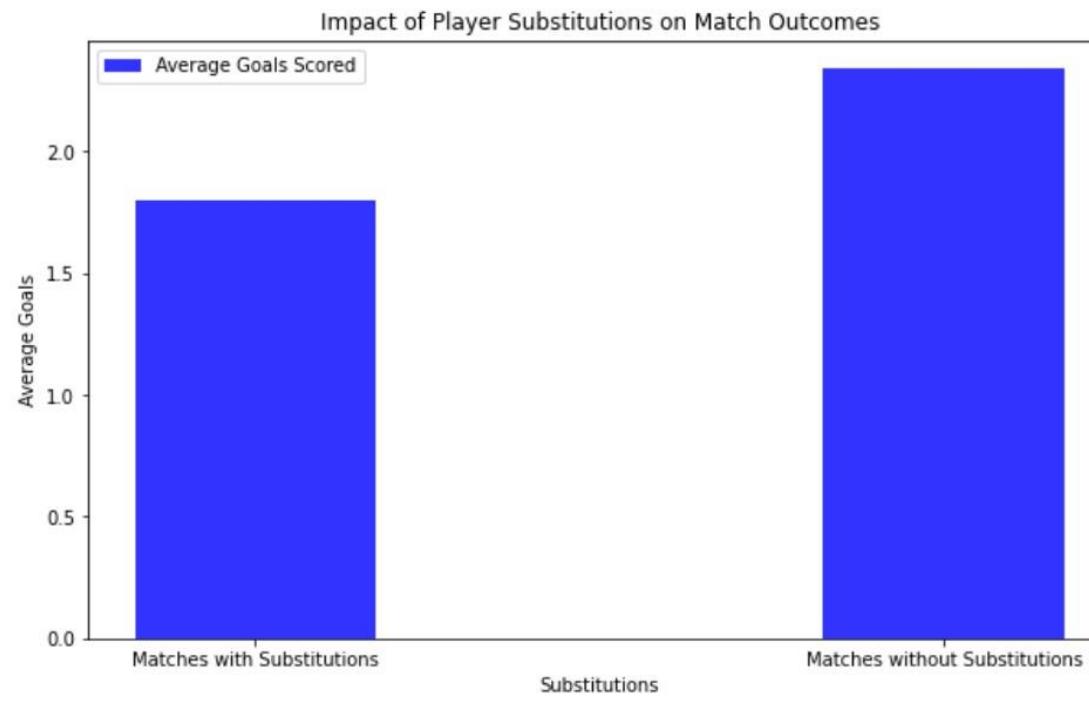
rects1 = ax.bar(labels, goals_data, bar_width, alpha=opacity, color='b', label='Average Goals
Scored')

ax.set_xlabel('Substitutions') ax.set_ylabel('Average Goals')
ax.set_title('Impact of Player Substitutions on Match Outcomes')
ax.legend()

plt.show()

```

## Result –



Q2 . Are there specific patterns in the timing and frequency of substitutions across different teams?

Solution –

Using Rows

- Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	70
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Are there specific patterns in the timing and frequency of substitutions across different teams?**

Click on enter option.

Are there specific patterns in the timing and frequency of substitutions across different teams?



Step 2 :

In deep dives you can see one new table is created with name Substitution timing by Team.  
Click on insert table option.

### Deep dives

Substitution Timing by Team

...

Insert table

Copy formula

Substitution Frequency by Team

Step 3 –

Below dataset we can see Substitution timing by Team Table.

	A	B	C	+
1	home_club_na	minute	COUNT player_i	
2	Borussia Dortm	90	70	
3	Borussia Dortm	77	68	
4	FC Schalke 04	90	49	
5	Fodbold Club No	84	45	
6	Bayer 04 Leverk	46	28	

Step 4 –

Select table. Click on insert option and select Pivot Table.

The screenshot shows a spreadsheet interface with a table of data. The 'Insert' button in the toolbar is highlighted. A dropdown menu is open under 'Insert', showing options like 'Table', 'Data Table', and 'Pivot Table'. The 'Pivot Table' option is selected and highlighted with a yellow box. Below the table, there's a row with cells containing 'home\_club\_na' and 'Borussia Dortm', which are also highlighted with yellow boxes.

Step 5 –

In pivot Table option on right side, click on Next Step.

## Pivot Table

Source Reference [\(?\)](#)

```
= 'Substitution Timing by Team'!A1:C354
```

e.g. A1

Next step

### Step 6 –

In pivot Table option on right side, select home\_club\_name in rows field and average minutes in values field.

## Pivot Table

The screenshot shows the configuration interface for a Pivot Table. At the top, there is a source reference box containing the formula `= 'Substitution Timing by Team'!A1:C354`. Below this, the 'Rows' section is set to `home_club_name`. The 'Columns' section is currently empty, showing a placeholder `Select value`. In the 'Values' section, there is a search bar with the placeholder `Q`. Under the 'Values' heading, three items are listed: `home_club_name`, `minute` (which is highlighted in grey), and `COUNT player_in_id`. Each item has a `+ Add` button to its right.

Values

**minute**

AVERAGE

Add

You can see the Pivot table for home\_club wise average minutes.

### Pivot Table

	A	B
1	home_club_na	
2	1. FuÃ¶ball- un	53.66667
3	1. FuÃ¶ball-Clu	68
4	1.FC Nurember	51.14286
5	AC Horsens	70.5
6	AFC Ajax Amste	76.66667

Step 7 –

Select Pivot Table, Click on insert option and select chart.

Insert

Table

Data Table

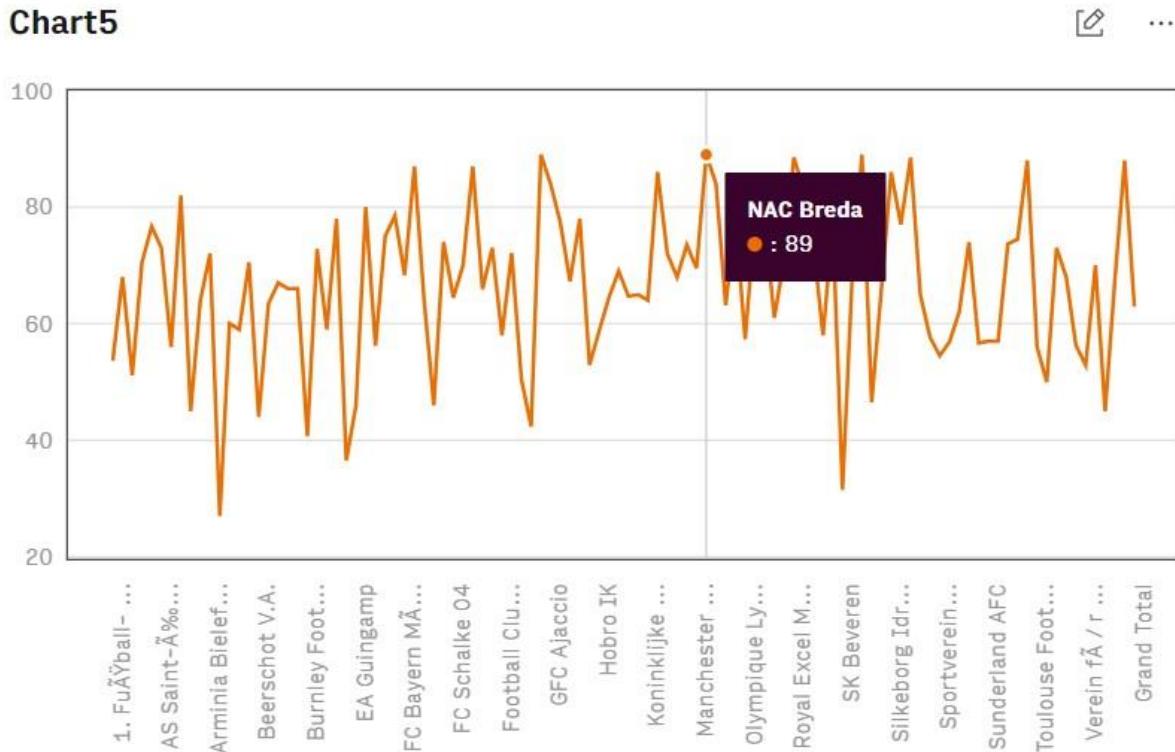
Pivot Table

Chart

Function

Button

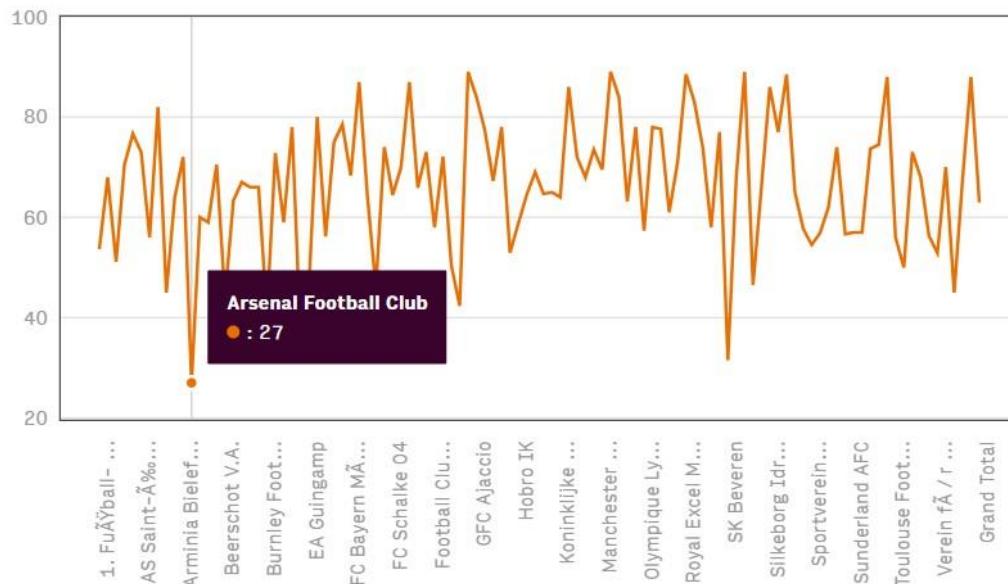
The obtained result is like –



Step 8 – By double clicking on title chart1, can edit title as Team wise average substitution time

## Team wise average substitution time

...



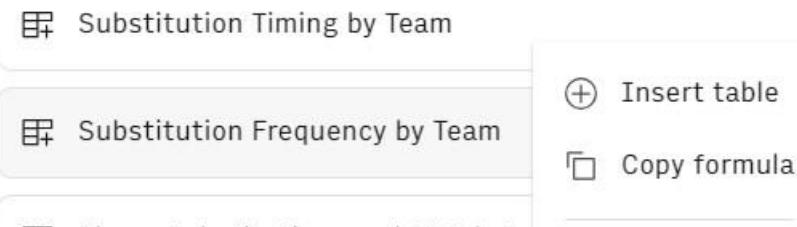
Interpretation –

Team NAC breda have maximum substitution time i.e. 89 minutes whereas team Arsenal Football Club have minimum substitution time i.e. 27 minutes.

Step 9 :

In deep dives you can see one new table is created with name Substitution Frequency by Team. Click on insert table option.

### Deep dives



Step 10 –

Below dataset we can see Substitution Frequency by Team Table. Sort table with Count of players in descending order.

**Substitution Frequency by Team**

The screenshot shows a table titled "Substitution Frequency by Team". The columns are labeled "A" and "B". The first column, "home\_club\_na", contains team names, and the second column, "COUNT player\_i", contains numerical values representing the count of substitutions. The table is sorted by the count in descending order. A sorting panel on the right indicates the current sort key is "COUNT player\_in\_i..." and the order is "Z → A".

A	B
1 home_club_na	COUNT player_i
2 1.FC Nurember	116
3 Hannover 96	22
4 Sport-Club Frei	20
5 Sportverein Wer	97
6 Verein fÃ / r Be	40

We can see the sorted table as -

**Substitution Frequency by Team**

The screenshot shows a table titled "Substitution Frequency by Team". The columns are labeled "A" and "B". The first column, "home\_club\_na", contains team names, and the second column, "COUNT player\_i", contains numerical values representing the count of substitutions. The table is sorted by the count in descending order. A sorting panel on the right indicates the current sort key is "COUNT player\_in\_i..." and the order is "Z → A".

A	B
1 home_club_na	COUNT player_i
2 Borussia Dortmund	494
3 Hobro IK	280
4 FC Schalke 04	228
5 Fodbold Club No	225
6 Hertha BSC	184

□ | ↪ | ⌂ | Insert ▲ | Data Actions | T ▾ | ⌂ ▾

A1	home_clu	
----	----------	--

**Table1**

	A	D
3924	4197	0613
3925	4198	0613
+ New row		3157

Table Data Table Pivot Table Chart Function Button Input field Dropdown list Date picker Checkbox Link

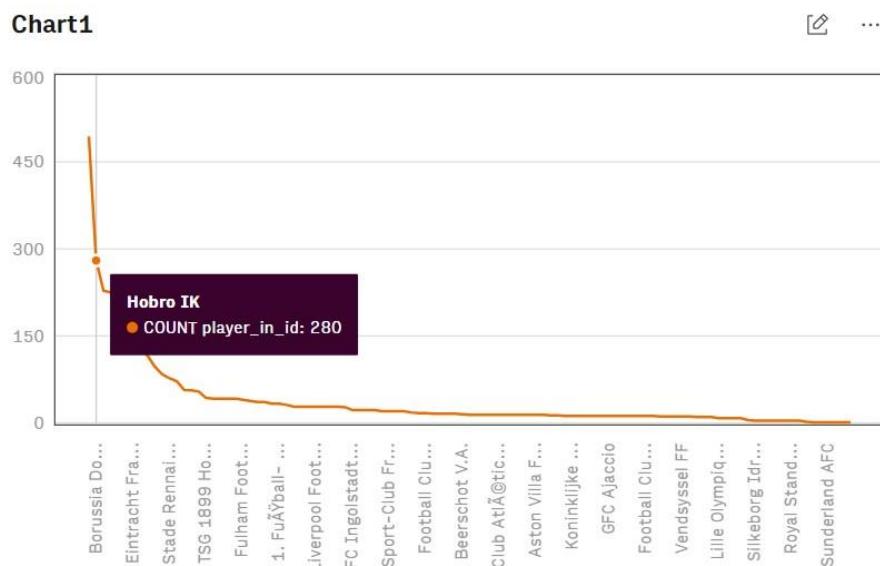
**Substitution Fre**

	A
1	<b>home_club_na</b>
2	Borussia Dortm
3	Hobro IK
4	FC Schalke 04
5	Fodbold Club No

Clear action element

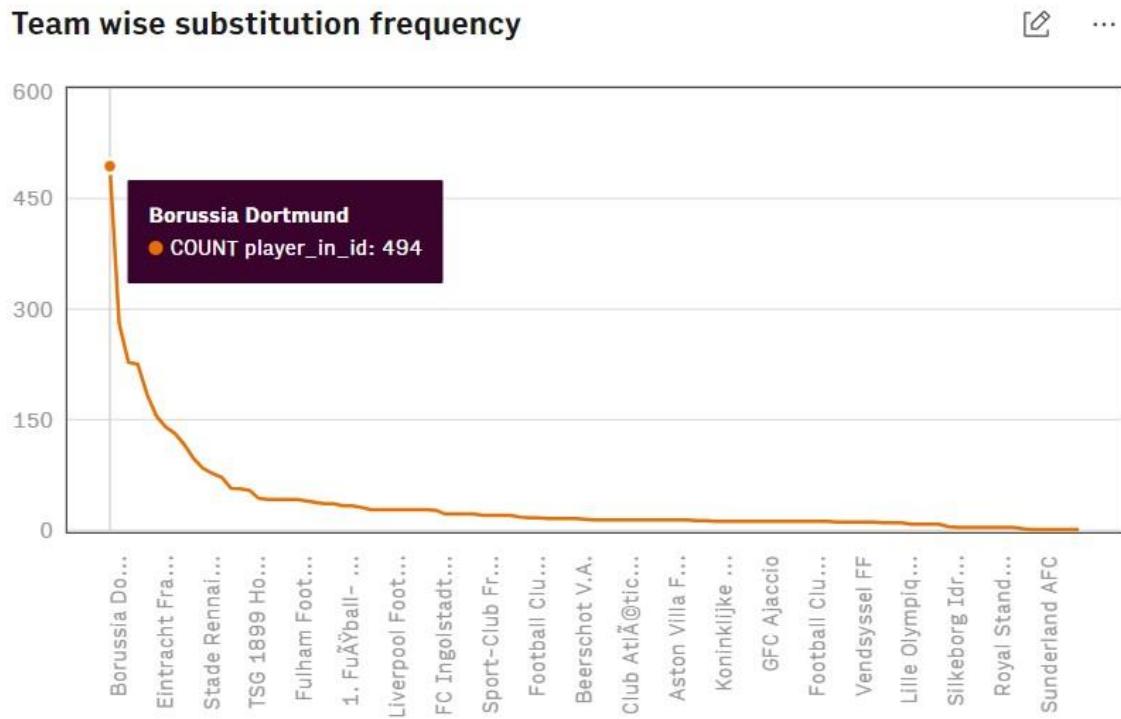
Step 11 –

Select table. Click on insert option and select chart.



Step 12 – By double clicking on title chart 1, can edit title as **Team wise substitution frequency**

Team wise substitution frequency



Interpretation –

Team Borussia Dortmund have maximum player substitution frequency i.e. 494.s

Using Python –

### 1. Team wise average substitution time

In Jupyter Notebook write following code and run–

```
# Identify substitutions
substitutions = df[df['type_x'] == 'Substitutions']

# Calculate team-wise average substitution time
team_avg_sub_time = substitutions.groupby('home_club_name')['minute'].mean()
print(team_avg_sub_time)

# Create bar chart
plt.figure(figsize=(10, 6))
plt.barh(team_avg_sub_time.index, team_avg_sub_time.values, color='blue')
```

```

plt.xlabel('Team') plt.ylabel('Average Substitution Time (Minutes)')
plt.title('Team-Wise Average Substitution Time in Football
Matches') plt.xticks(rotation=45, ha='right') plt.tight_layout()
plt.show()

```

Result –

home_club_name	
1. Fußball- und Sportverein Mainz 05	62.800000
1. Fußball-Club Kägln	69.300000
1.FC Nuremberg	55.000000
AC Horsens	71.000000
AFC Ajax Amsterdam	79.500000
	...
Verein für Bewegungsspiele Stuttgart 1893	67.000000
Verein für Leibesübungen Wolfsburg	70.765957
Watford FC	77.500000
Willem II Tilburg	58.000000
Yellow-Red Koninklijke Voetbalclub Mechelen	67.500000
Name: minute, Length: 93, dtype: float64	

---

## 2. Team wise substitute frequency

In Jupyter Notebook write following code and run–

```
import pandas as pd import matplotlib.pyplot as plt
```

```

# Assume df is your DataFrame containing football match summary data

# Filter only substitution events substitution_df =
df[df['type_x'] == 'Substitutions'] # Count the
substitution frequency for each team

team_substitution_count =
substitution_df.groupby(['home_club_name']).size().reset_index(name='substitution_count')

# Sort the DataFrame in descending order by index
team_substitution_count = team_substitution_count.sort_values(by='substitution_count',
ascending=False) team_substitution_count

```

Result –

	home_club_name	substitution_count
14	Borussia Dortmund	328
31	FC Schalke 04	216
45	Hobro IK	178
34	Fodbold Club Nordsjælland	174
18	Chelsea Football Club	112
...	...	...
67	SC Paderborn 07	1
32	Feyenoord Rotterdam	1
28	FC Groningen	1
23	Eindhovense Voetbalvereniging Philips Sport Ve...	1
73	Southampton FC	1

Interpretation –

Team Borussia Dortmund have maximum player substitution frequency.

---

Q3 . Do certain managers exhibit more effective substitution strategies?

Solution –

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70
3	1	2231426_497	2231426	49723 L1	0	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Do certain managers exhibit more effective substitution strategies?**

Click on enter option.

Do certain managers exhibit more effective substitution strategies?



Step 2 -

In deep dives you can see one new table is created with name Manager Substitution Effectiveness Ranking. Click on insert table option.

### Deep dives

Manager Substitution Effectiveness Ranking ...

+ Insert table

Substitution Frequency by Team

Copy formula

Step 3:

Below dataset we can see Manager Substitution Effectiveness Ranking Table.

## Manager Substitution Effectiveness Ranking

	A	B	C	D	
1	home_club_ma	COUNT player_i	SUM assists	SUM goals	
2	Lucien Favre	161	67	94	
3	Thomas Tuchel	124	56	76	
4	Peter StÅ¶ger	153	42	55	
5	Julien StÅ©pha	72	36	72	
6	Kasper Hjulman	209	34	85	

Step 4 – Sort Brand wise Manager Substitution Effectiveness Ranking data by sum of goals in descending order and click on sort option

Manager Substitution Effectiveness Ranking

	A	B	C	D	
1	home_club_ma	COUNT player_i	SUM assists	SUM goals	
2	Lucien Favre	161	67	94	
3	Thomas Tuchel	124	56	76	
4	Peter StÅ¶ger	153	42	55	
5	Julien StÅ©pha	72	36	72	
6	Kasper Hjulman	209	34	85	
7	Erik ten Hag	38	30	0	
8	Andries Jonker	28	28	28	

Sort by ②  
 SUM goals (D1) ▾ Z → A ▾

Step 5–

Select sorted table. Click on insert option and select chart.

Manager Substitution Effectiveness Ranking

	A	B	C	D	
1	home_club_ma	COUNT player_i	SUM assists	SUM goals	
2	Sean Dyche	126	0	126	
3	Lucien Favre	94	67	94	
4	Kasper Hjulman	85	34	85	

Insert ▾ Data Actions ▾ T ▾ E ▾ F ▾ R ▾

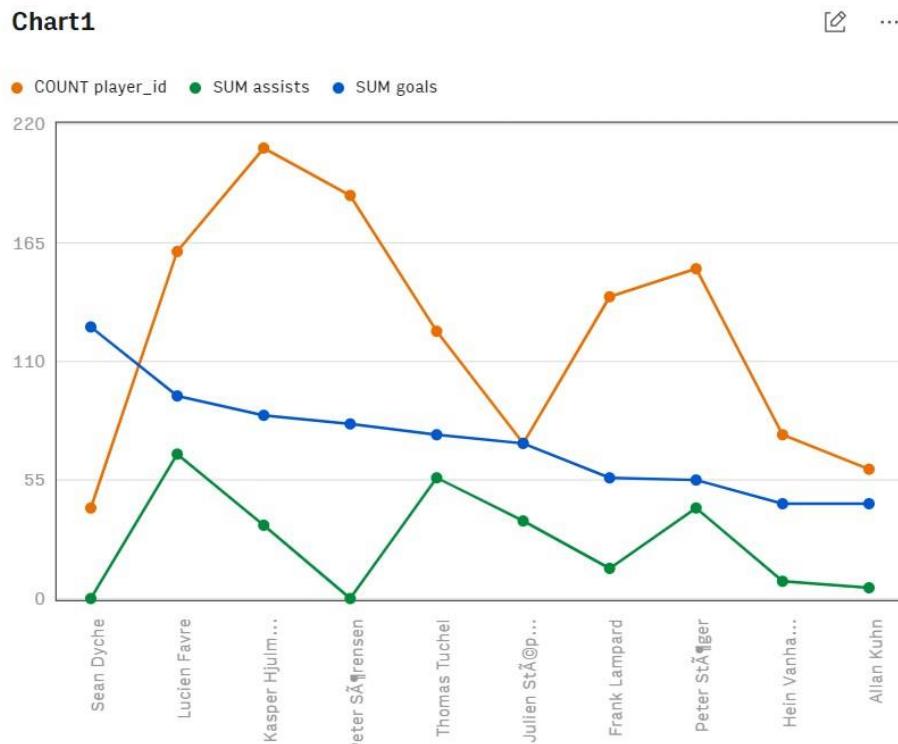
A1 | home\_club\_ma

Manager Substitution Effectiveness Ranking

	A	B	C	D	
1	home_club_ma	COUNT player_i	SUM assists	SUM goals	
2	Sean Dyche	126	0	126	
3	Lucien Favre	94	67	94	
4	Kasper Hjulman	85	34	85	

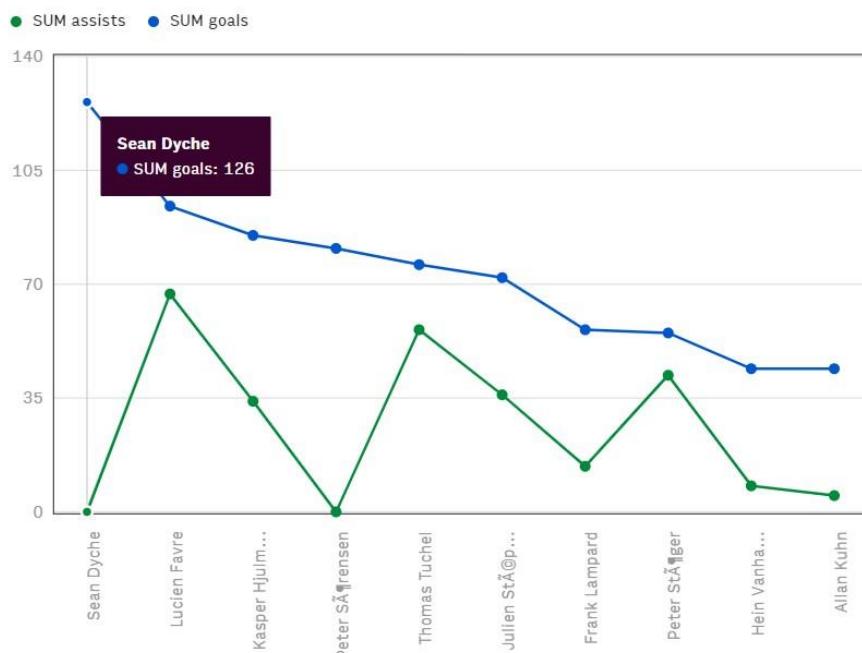
- Table
- Data Table
- Pivot Table
- Chart
- Function
- Button

You can see the result as -

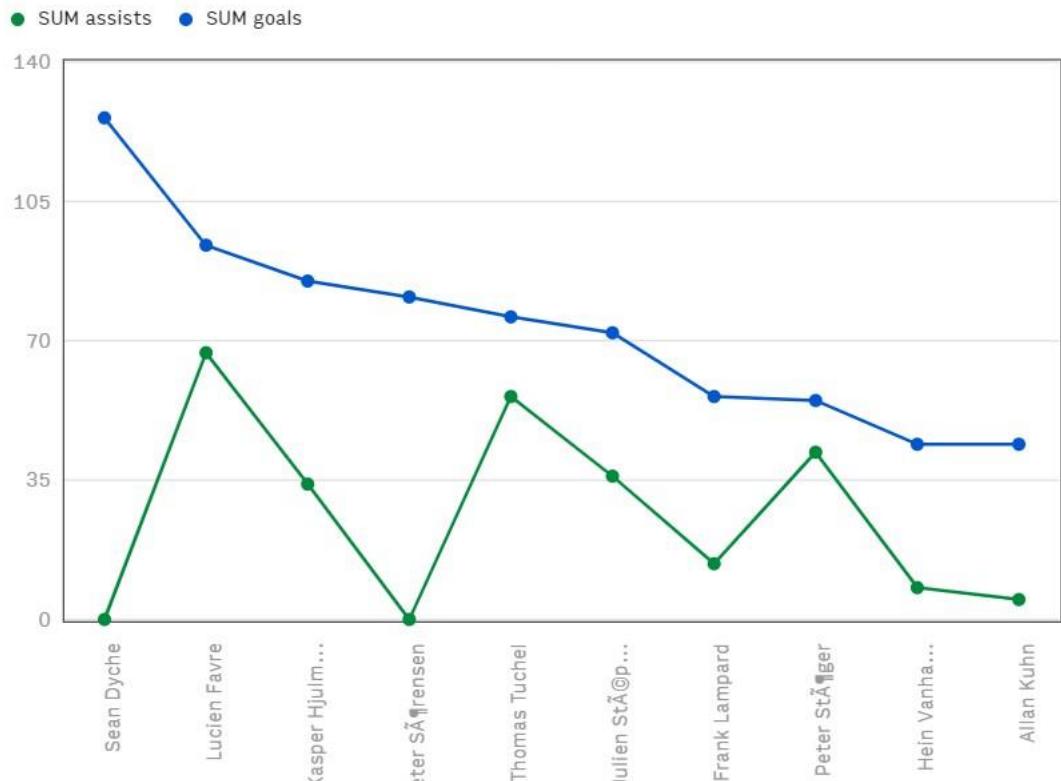


Step 6 –

Remove count of player\_id from chart editor.

**Chart1**

Step 7 – By double clicking on title chart1, can edit title as **Top 10 managers who used effective sub.**

**Top 10 managers who used effective sub.**

Interpretation –

Sean Dyche is a manager with high effectiveness followed by Lucien Favre.

Using SQL

Run the following query in sql –

-- Manager effectiveness

```
SELECT home_club_manager_name, COUNT(player_id), SUM(assists), SUM(goals) from football_data_merged_cleaned
```

```
WHERE player_id IS NOT NULL AND player_in_id IS NOT NULL GROUP BY home_club_manager_name ORDER BY SUM(goals) DESC
```

Result –

home_club_manager_name	COUNT(player_id)	SUM(assists)	SUM(goals)
Sean Dyche	42	0	126
Julien StÃ©phan	72	36	72
Lucien Favre	112	42	70
Thomas Tuchel	104	56	56
Frank Lampard	140	14	56
Peter StÃ¶ger	153	42	55
Hein Vanhaezebrouck	76	8	44
Bruno Labbadia	57	6	39

Interpretation –

Sean Dyche is a manager with high effectiveness followed by Lucien Favre.

---

---

### **Business Objective 7: Event Analysis**

Q1. How do specific events (goals, cards, substitutions) correlate with match results?

Q2. Are there particular events that consistently lead to changes in momentum during matches?

Q3. What is the impact of key events on fan engagement and overall match excitement?

Q1.

How do specific events (goals, cards, substitutions) correlate with match results? Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	70
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.

**AI Analyst**

Table1

Ask AI a question about your data...

### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**How do specific events (goals, cards, substitutions) correlate with match results?**  
Click on enter option.

How do specific events (goals, cards, substitutions) correlate with match results?

In Quick insights you can see the result for correlation between goals and substitution is 0.02.

### Quick insights

The correlation between goals and substitutions is 0.02.

The correlation between cards and match results is 0.00

Interpretation – correlation between goals and substitution is 0.02.

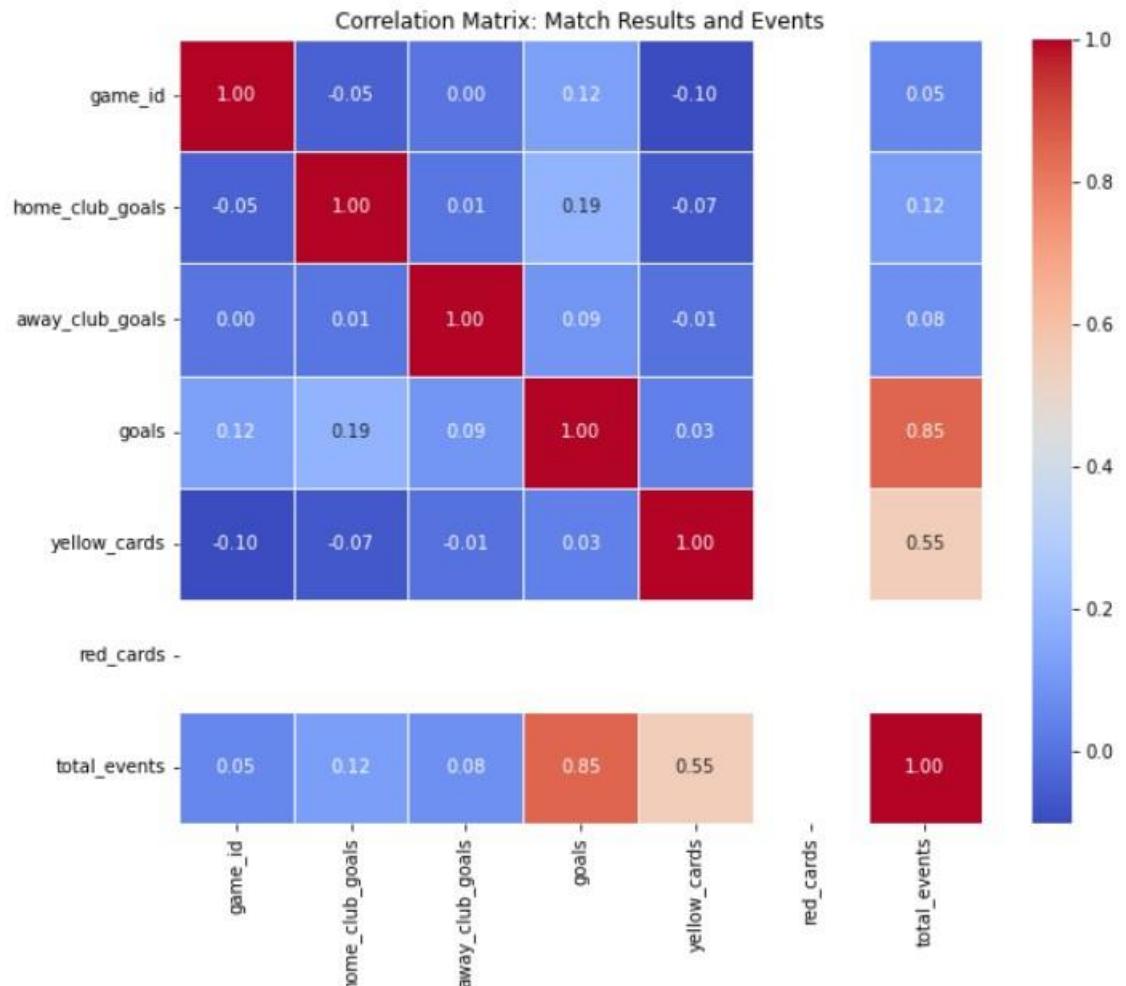
Using Python –

Run the following code in jupyter

```
Notebook- import pandas as pd import  
seaborn as sns import matplotlib.pyplot as  
plt
```

```
# Assume df is your DataFrame containing football match summary data  
  
# Create a new DataFrame with relevant columns  
events_df = df[['game_id', 'home_club_goals', 'away_club_goals', 'goals', 'yellow_cards',  
'red_cards']]  
  
# Calculate total events (goals, yellow cards, red cards) for each match  
events_df['total_events'] = events_df['goals'] + events_df['yellow_cards'] +  
events_df['red_cards']  
  
# Calculate correlation matrix correlation_matrix  
= events_df.corr()  
  
# Visualize the correlation matrix using a heatmap  
plt.figure(figsize=(10, 8))  
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt='.2f', linewidths=.5)  
plt.title('Correlation Matrix: Match Results and Events') plt.show()
```

Result –



Interpretation –

Goals and cards does not have lot much correlation.

---

Q2 . Are there particular events that consistently lead to changes in momentum during matches? Solution – Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	0
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Are there particular events that consistently lead to changes in momentum during matches?**

Click on enter option.

Are there particular events that consistently lead to changes in momentum during matches?



Step 2 :

In deep dives you can see one new table is created with name Assists leading to Changes to momentum. Click on insert table option.

### Deep dives

Assists Leading to Changes in Momentum

Goals Leading to Changes in Momentum

Events Leading to Changes in Momentum



Step 3 –

Below dataset we can see Assists leading to Changes to momentum Table.

### Assists Leading to Changes in Momentum

A	B	+
1	COUNT game_e	
2	17	
3	17	
4	17	
5	15	
6	15	

Step 4 –

Select table. Click on insert option and select chart.

The screenshot shows a data analysis interface with a toolbar at the top. In the center, there is a table titled "Top 10 manager" with columns "game\_e" and "Julien". Below the table, there is another section titled "Assists Leading" with a dropdown menu. A context menu is open over the "Julien" column, with the "Insert" option highlighted. Under "Insert", the "Chart" option is selected. Other options in the "Insert" menu include "Table", "Data Table", "Pivot Table", and "Function". The "Assists Leading" section also has a dropdown menu with options like "Function", "Button", etc.

The line chart will create as below –

## Chart1



Interpretation -

Three game events have maximum number of games i.e. 17.

Using SQL-

Write following query in Mysql

-- Game events momentum

```
SELECT game_event_id, COUNT(game_event_id) from football_data_merged_cleaned
WHERE assists > 0 GROUP BY game_event_id ORDER BY COUNT(game_event_id)
DESC;
```

Result –

game_event_id	COUNT(game_event_id)
64f295d405cea500d04aa2271bd8ad45	14
b6a161b22e38d9f2bc9595ced30fe819	14
b3b3319cc00c4eb21eb03453b90c54f4	14
bf2b6e9b485d340094a575bca8dfdb66	14
d5e1adccdd77e3f37618cde9d5ad6651	14
1bfd96ef65479b97e7e17b18ec34bcb3	14
f63704ad7aa32bd8975ac906e1d53384	14
a0553a4a6ed2d47e9129a5e516b31a54	14

Q3.

What is the impact of key events on fan engagement and overall match excitement?

Solution –

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70

AI analyst popup will open.

The screenshot shows the AI Analyst interface. At the top, there is a table titled "Table1" with the following data:

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70
3	1	2231426_497	2231426	49723 L1	0	0	0	0	70

Below the table, there is a section titled "Quick insights" containing the following text:

- The total number of appearance IDs is 3,924.
- The total number of unique game IDs is 336.
- The highest number of goals scored is 3.
- The average minutes played per game is 69.96.

Type here prompt as –

What is the impact of key events on fan engagement and overall match excitement?

Click on enter option.

What is the impact of key events on fan engagement and overall match excitement?

Step 2 :

In deep dives you can see one new table is created with name Match Excitement vs Key Events and Fan Engagement vs Key Events. Click on insert table option for Match Excitement vs Key Events.

## Deep dives

➡ Red Cards vs Key Events

➡ Yellow Cards vs Key Events

➡ Assists vs Key Events

➡ Match Excitement vs Key Events

➡ Fan Engagement vs Key Events

Step 3 –

Below dataset we can see Match Excitement vs Key Events Table.

Match Excitement vs Key Events

1	A	B	+
1	attendance	COUNT goals	
2	81360	210	
3	42100	61	
4	81365	57	
5	51500	46	
6	50000	44	

Step 4 –

Change chart type in chart editor as column

## Chart Editor

Setup More options

Chart type

Column

Stacking

None

Step 5 – By double clicking on title chart 1, can edit title as **Match excitement - Goal count wise**

Step 6 – In more options under Chart editor, Change data labels as top.

## Chart Editor

Setup More options

Min

Max

Title

Show vertical left axis



### LEGEND AND LABELS

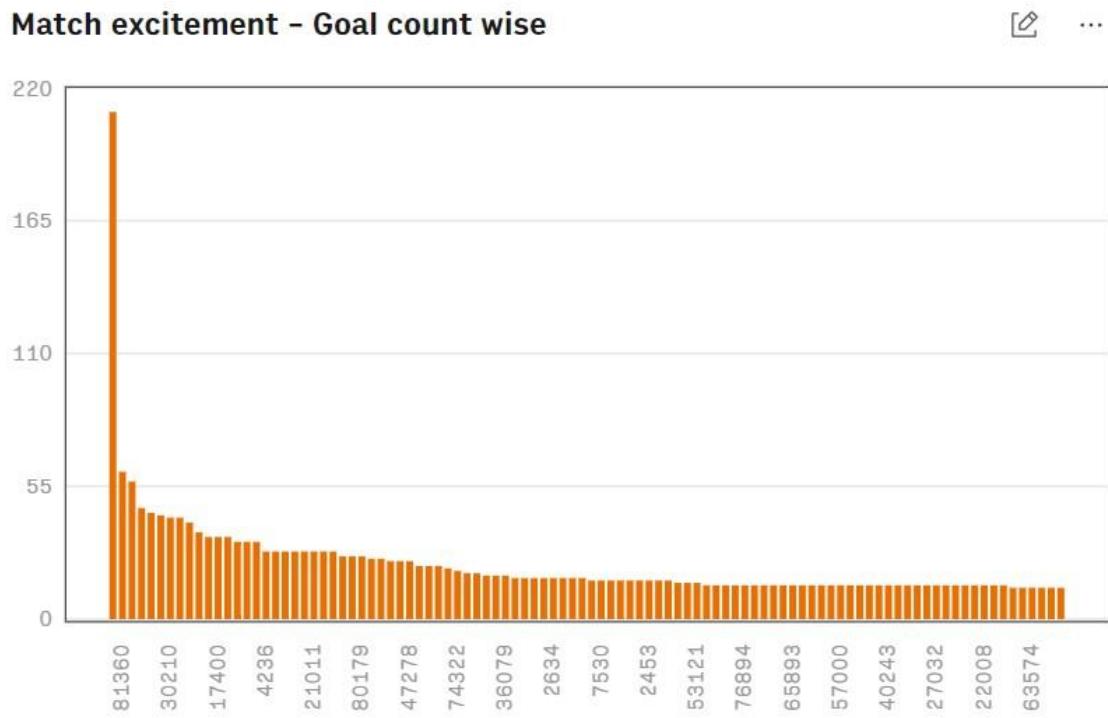
Legend position

Auto

Data labels

None

Result -



Interpretation – We can see that as number of goals in match increases attendance of peoples (match excitement) increases.

For Fan Engagement vs key events follow the same steps as above and we can see the same result.

### Fan Engagement vs Key Events

A	B	C	D	E	F
1 attendance	COUNT competition_type				
2 81365		57			
3 81360		210			
4 81359		14			
5 81000		28			
6 80667		13			

Fan Engagement vs Key Events

	A	B	+
1	attendance	COUNT competition_type	
2	81365	57	
3	81360	210	
4	81359	14	
5	81000	28	
6	80667	13	
7	80365	14	
8	80196	26	

Sort by ?

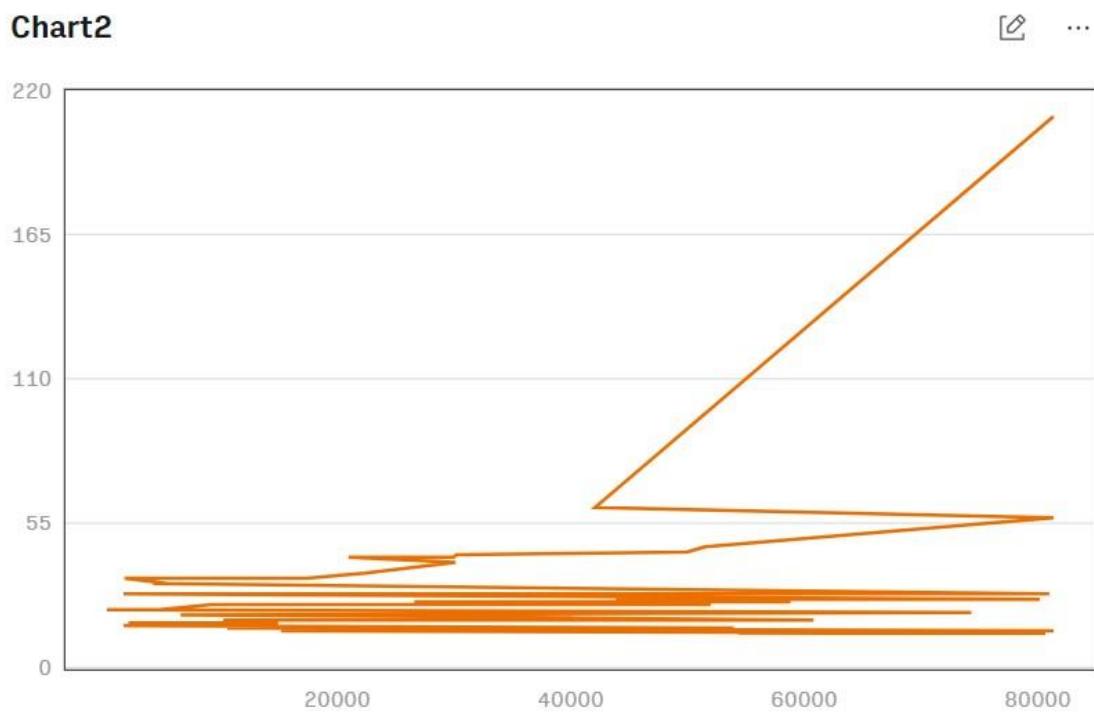
COUNT competition\_type ▾ Z → A ▾

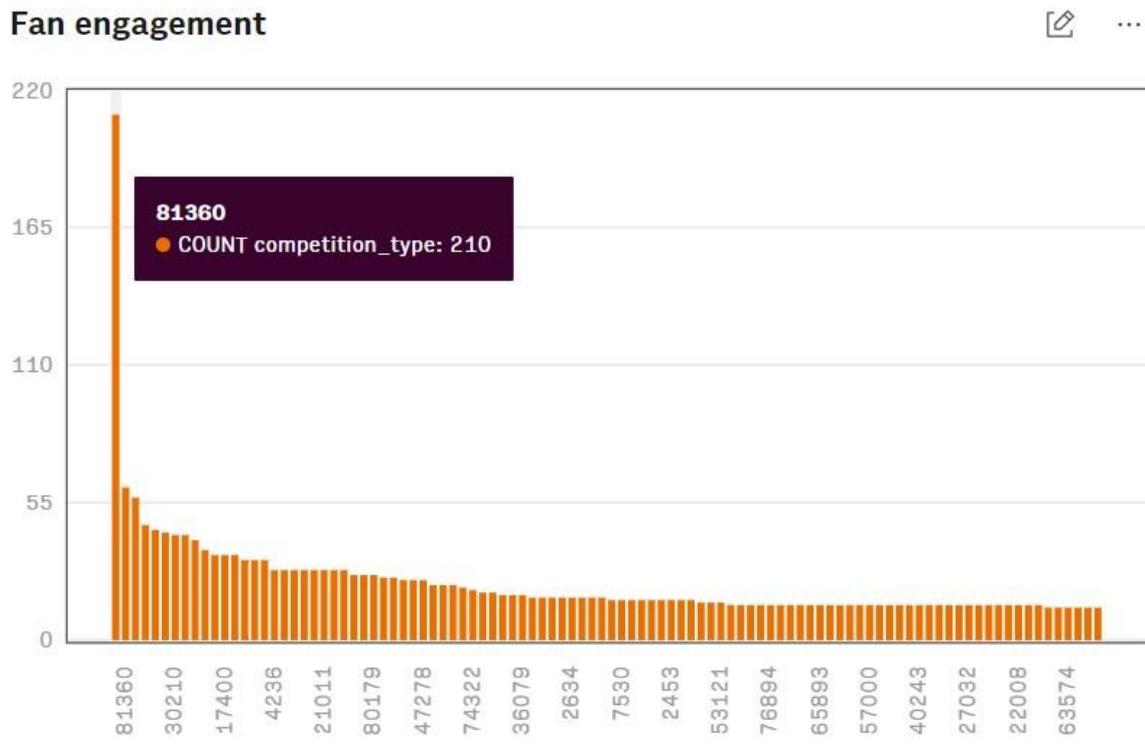
Sort

□ | ↺ ↻ | + Insert ▲ ◎ Data Actions | T ▾

A1	attendance	
Events Leading		<span>Table</span> <span>Data Table</span> <span>Pivot Table</span> <span>&lt;&gt;</span>
Match Exciteme		<span>Chart</span> <span>Function</span> <span>Up</span> <span>Down</span> <span>Search</span>
1	attendance	
2	81360	<span>Button</span>
3	42100	<span>Input field</span>
4	81365	<span>Dropdown list</span>
5	51500	<span>Date picker</span>
6	50000	<span>Checkbox</span>
7	30210	<span>Link</span>
8	30000	
9	20975	<span>Clear action element</span>
10	30150	40
11	22430	36

A1	attendance	Table
1	attendance	Data Table
97	63574	Pivot Table
98	56881	Chart
99	56376	Function
100	55617	Button
101	54376	Input field
102	51953	Dropdown list
103	49500	Date picker
104	48721	Checkbox





Interpretation – We can see that as number of competition types in event increases attendance of peoples (match excitement) increases.

Using SQL-

Run the following query in sql to find match excitement

```
-- match excitement goal wise
```

```
SELECT attendance, COUNT(goals) from football_data_merged_cleaned
WHERE goals IS NOT NULL GROUP BY attendance ORDER BY count(goals) DESC;
```

Result –

	attendance	COUNT(goals)
▶	81360	210
	81365	56
	42100	47
	50000	44
	51500	44
	30210	42
	30000	42
	20975	42

Interpretation – We can see that as number of goals in match increases attendance of peoples (match excitement) increases.

---

Q. Is there a significant difference in the average number of goals scored by players of different preferred foot (left or right)? Select 35 random samples of the average number of goals scored by players from the given data. Use the 5% significance level for the test.

Solution: Two-Tailed Hypothesis z-test

Establish the null and alternate hypothesis

H0: there is no difference in average number of goals scored by players of left foot and right foot.

Ha: there is significance difference in average number of goals scored by players of left foot and right foot.

The test needs to be conducted for the value which is greater than or less than the difference of population means. So, we have to perform the two-tailed test.

Calculate Sample statistic df =

```
pd.read_csv('Football_data_merged_cleaned.csv') data  
= df
```

right

```
# Calculate total goals for left-footed and right-footed players  
left_foot_goals = df[df['foot'] == 'left'] right_foot_goals =  
df[df['foot'] == 'right'] len(left_foot_goals)  
len(right_foot_goals)
```

```
left_foot_goals = left_foot_goals.sample(n=35, random_state=1, replace=True)  
right_foot_goals = right_foot_goals.sample(n=35, random_state=1, replace=True)
```

#Sample 1: Left lef =

```
left_foot_goals['goals'] n1
```

```
= len(lef)
```

```
s_mean1 = st.mean(lef)
```

```
s_sd1 = st.stdev(lef)
```

```
s_var1 = s_sd1**2
```

```
print('For Left foot:')

print('Mean =' +str(s_mean1))

print('Standard Deviation =' +str(s_sd1))

print('Sample Size =' +str(n1))

print('Variance =' +str(s_var1))
```

```
#Sample 2: Right righ =
right_foot_goals['goals'] n2 =
len(right_foot_goals['goals']) s_mean2 =
st.mean(right_foot_goals['goals']) s_sd2 =
st.stdev(right_foot_goals['goals']) s_var2 =
s_sd2**2
```

```
print("\nFor right foot:)

print('Mean =' +str(s_mean2))

print('Standard Deviation =' +str(s_sd2))

print('Sample Size =' +str(n2))

print('Variance =' +str(s_var2))
```

Set the value of alpha

It is given that a 5% level of significance to be used to test hypothesis.

alpha ( $\alpha$ ) = 0.05

This test is a two-tailed test, each of the two rejection regions has an area of .025.

Establish the decision rule

- i. If  $p\text{-value} < \alpha$  : Rejection of Null Hypothesis( $H_0$ )
- ii. If  $-z\text{-critical} > z\text{-statistic} > +z\text{-critical}$  : Rejection of Null Hypothesis( $H_0$ )

```
Analyze the data alpha=0.05 z_statistics = ((s_mean1 - s_mean2)- 0) /  
(math.sqrt(s_var1/n1 + s_var2/n2))  
print("The Z statistics is ", z_statistics)
```

```
p_value = norm.sf(abs(z_statistics))*2 #two tailed test print("The  
p_value is "+str(p_value))
```

```
z_critical = norm.ppf(1 - alpha/2) # two tailed test print("The  
z-critical value is "+str(z_critical))
```

### **Business Objective 8: Competition Analysis:**

How does team and player performance differ across various competitions (domestic league, cup matches, etc.)?

Are there teams that consistently perform better in specific types of competitions? Does the intensity of play vary between domestic and international competitions?

Q1 . How does team and player performance differ across various competitions (domestic league, cup matches, etc.)? Solution – using rows Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	0
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**How does team and player performance differ across various competitions (domestic league, cup matches, etc.)?** Click on enter option.

How does team and player performance differ across various competitions (domestic league, cup matches, etc.)?



Step 2 – In deep dives Average Gals Scored Across Competitions will appear.

### Deep dives

Average Club Position Across Competitions

Average Goals Scored Across Competitions

Average Minutes Played Across Competitions

Player Performance Across Competitions

Step 3 - Click on insert table option.

## Deep dives

Average Club Position Across Competitions

Average Goals Scored Across Competitions ...

Average Minutes Played Across Competitions Insert table

Copy formula

Step 4 – Select Table and insert Chart.

**Average Goals Scored Across Competitions**

	A	B	C	+
1	competition_id	AVG home_club	AVG away_club	
2	L1	1.8085598612	1.5702718334	
3	DFB	1.9761904762	2.5158730159	
4	EL	1.9876543210	1.3333333333	
5	BE1	1.9230769231	1.3589743590	
6	CL	2.4458598726	0.8980891720	

A1 competition\_id

Average Goals S

	A
1	competition_id
2	L1
3	DFB
4	EL
5	BE1

competition\_id

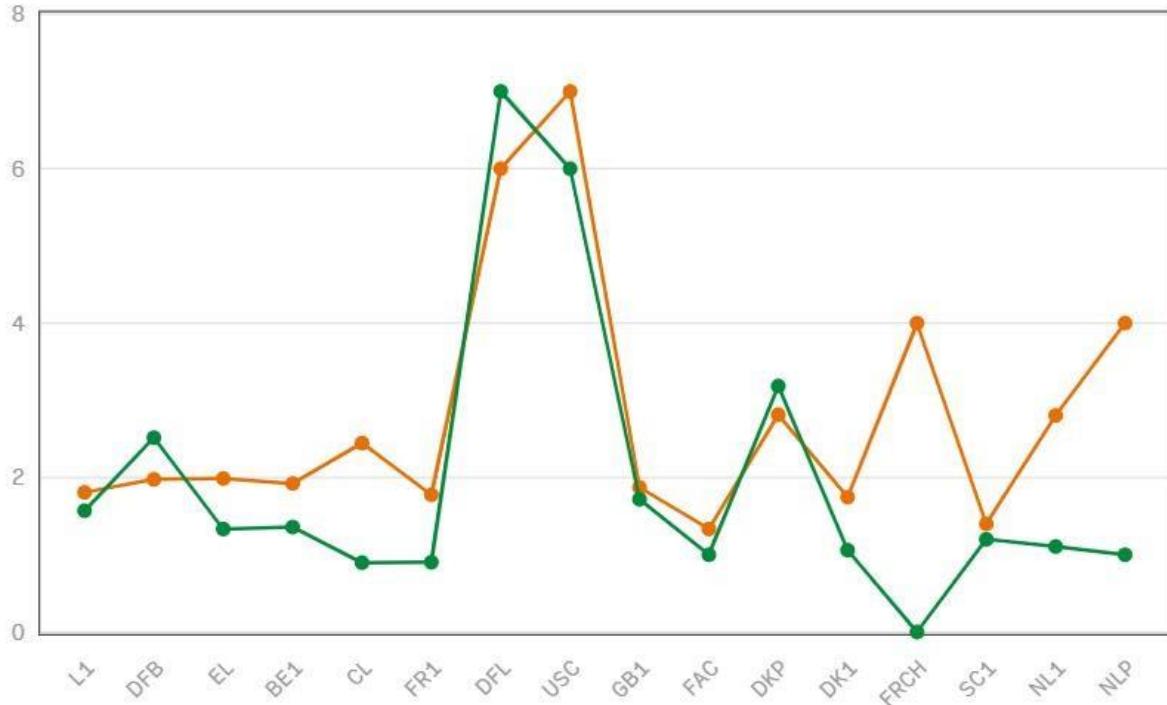
- Table
- Data Table
- Pivot Table
- Chart
- Function
- Button
- Input field
- Dropdown list

You can see the result as below -

**Chart2**



● AVG home\_club\_g... ● AVG away\_club\_g...

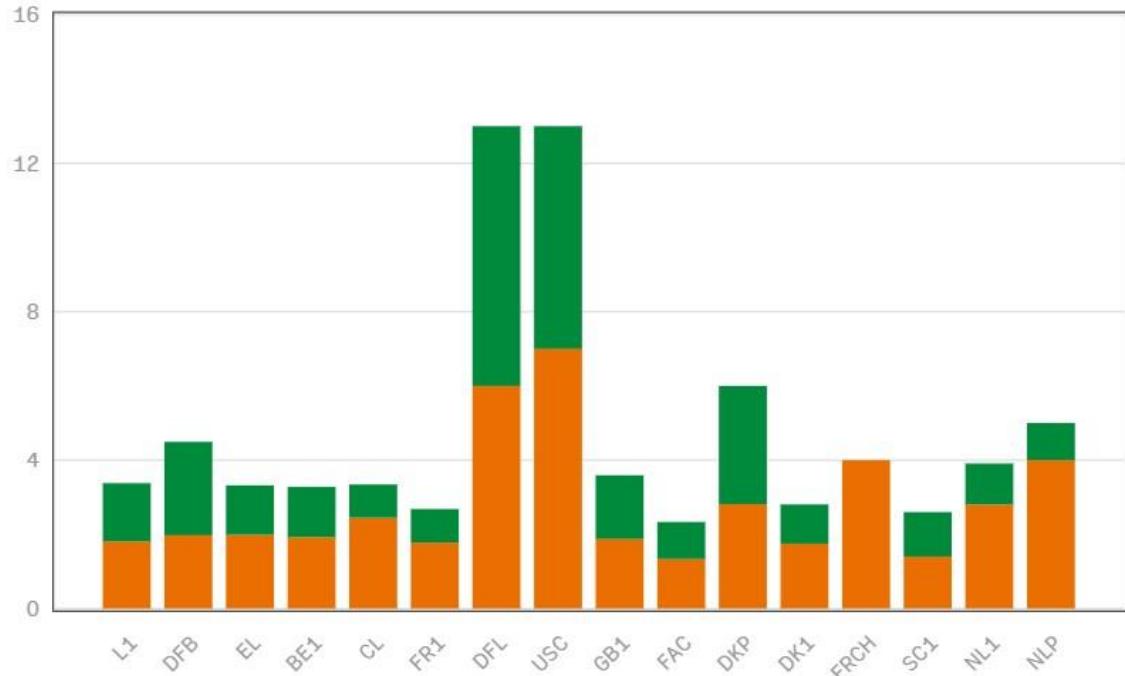


Change the chart title as **Competition wise average Goal scores**

**competition wise average Goal Scores**



● AVG home\_club\_g... ● AVG away\_club\_g...



Interpretation –

USC and DFL competitions have average maximum number of goals followed by DKP and NLP competitions.

Using SQL-

Write following query in MySQL workbench –

-- competition wise average group score

```
SELECT competition_id, AVG(home_club_goals), AVG(away_club_goals) from football_data_merged_cleaned GROUP BY competition_id;
```



	competition_id	AVG(home_club_goals)	AVG(away_club_goals)
▶	L1	1.9539	1.5359
	DFB	1.8586	2.5455
	EL	1.9877	1.3333
	BE1	1.7586	1.1379
	CL	2.2833	0.7667
	FR1	1.7500	0.8500
	DFL	6.0000	7.0000
	USC	7.0000	6.0000

Using Python –

```
import pandas as pd
```

```
# Assuming df is your DataFrame containing football match summary data
```

```
# Group by competition and calculate average goal scores
```

```
competition_avg_goals = df.groupby('competition_id')[['home_club_goals', 'away_club_goals']].mean()
```

```
# Calculate total average goals for each competition
```

```
competition_avg_goals['total_avg_goals'] = competition_avg_goals.mean(axis=1)
```

```
# Display the result print(competition_avg_goals)
```

Result –

competition_id	home_club_goals	away_club_goals	total_avg_goals
BE1	1.923077	1.358974	1.641026
CL	2.445860	0.898089	1.671975
DFB	1.976190	2.515873	2.246032
DFL	6.000000	7.000000	6.500000
DK1	1.746367	1.059445	1.402906
DKP	2.814815	3.185185	3.000000
EL	1.987654	1.333333	1.660494
FAC	1.333333	1.000000	1.166667
FR1	1.777778	0.903704	1.340741
FRCH	4.000000	0.000000	2.000000
GB1	1.872521	1.719547	1.796034
L1	1.808560	1.570272	1.689416
NL1	2.803279	1.106557	1.954918
NLP	4.000000	1.000000	2.500000
SC1	1.400000	1.200000	1.300000
USC	7.000000	6.000000	6.500000

Interpretation –

USC and DFL competitions have average maximum number of goals followed by DKP and NLP competitions.

---

Q2 Are there teams that consistently perform better in specific types of competitions?

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70
3	1	2231426_497	2231426	49723 L1	0	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Are there teams that consistently perform better in specific types of competitions?**

Click on enter option.

Are there teams that consistently perform better  
in specific types of competitions?



In deep dives you can see the Team performance by Competition Type Table is appeared.

### Deep dives

Team Performance by Competition Type

...

Insert table

Number of Appearances per Game

Copy formula

Click on Insert Table option.

## Team Performance by Competition Type

A	B	C	+
1 home_club_na	competition_ty	AVG goals	
2 Burnley Football	domestic_leagu	3	
3 1.FC Nurember	domestic_cup	1	
4 Verein fÃ / r Lei	international_c	1	
5 FC Bayern MÃ /	domestic_cup	1	
6 Royal Sporting	domestic_leagu	1	

Then insert chart.

□ | ↪ ↤ | + Insert ▲ ⌂ Data Actions | ↴

A1 | home\_clu

Team Performance

Table

Data Table

Pivot Table

Chart

Function

Button

1

1

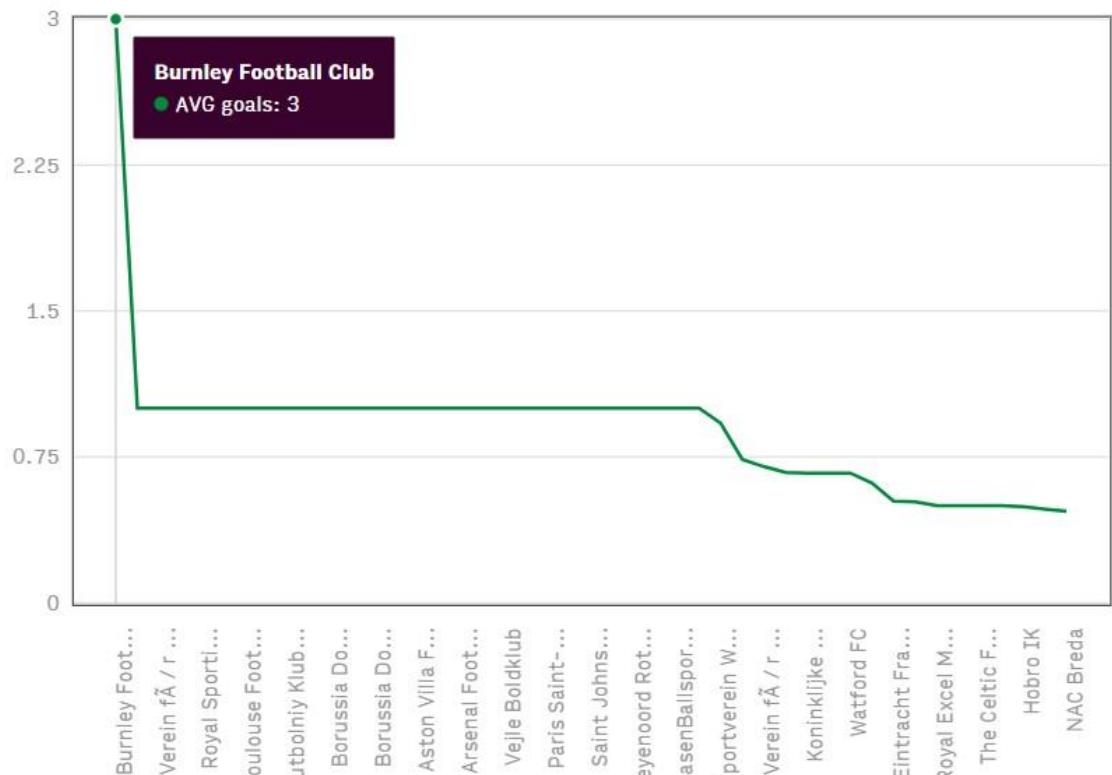
1

1

## Chart2

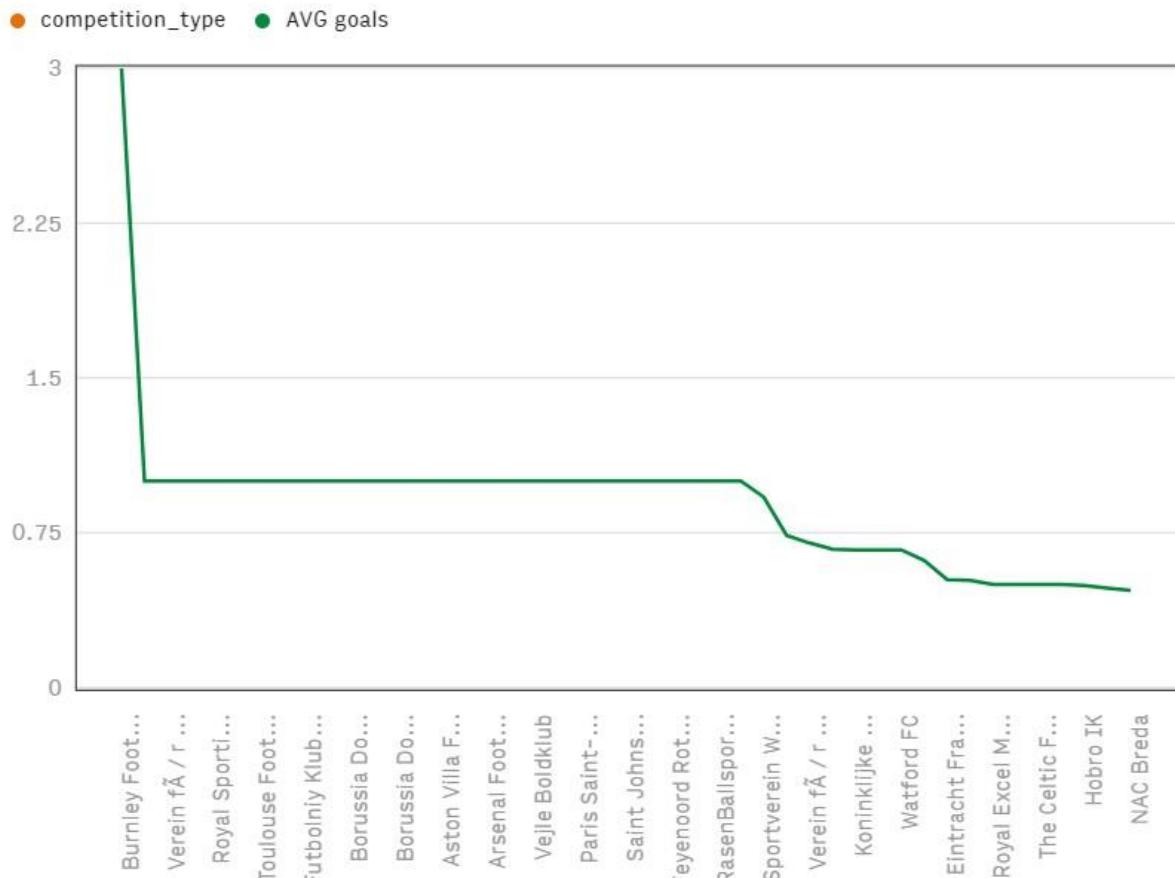
...

● competition\_type ● AVG goals



## competition wise performance

...



Python –

```
import pandas as pd import
```

```
matplotlib.pyplot as plt
```

```
# Assume df is your DataFrame containing football match summary data
```

```
# Calculate a performance metric (e.g., total_goals + total_assists) for each player
```

```
df['performance_metric'] = df['goals'] + df['assists']
```

```
# Group by competition and player, then calculate the total performance metric
```

```
competition_top_performers = df.groupby(['competition_type',  
'player_name'])['performance_metric'].sum().reset_index()
```

```
# Find the top performer in each competition
```

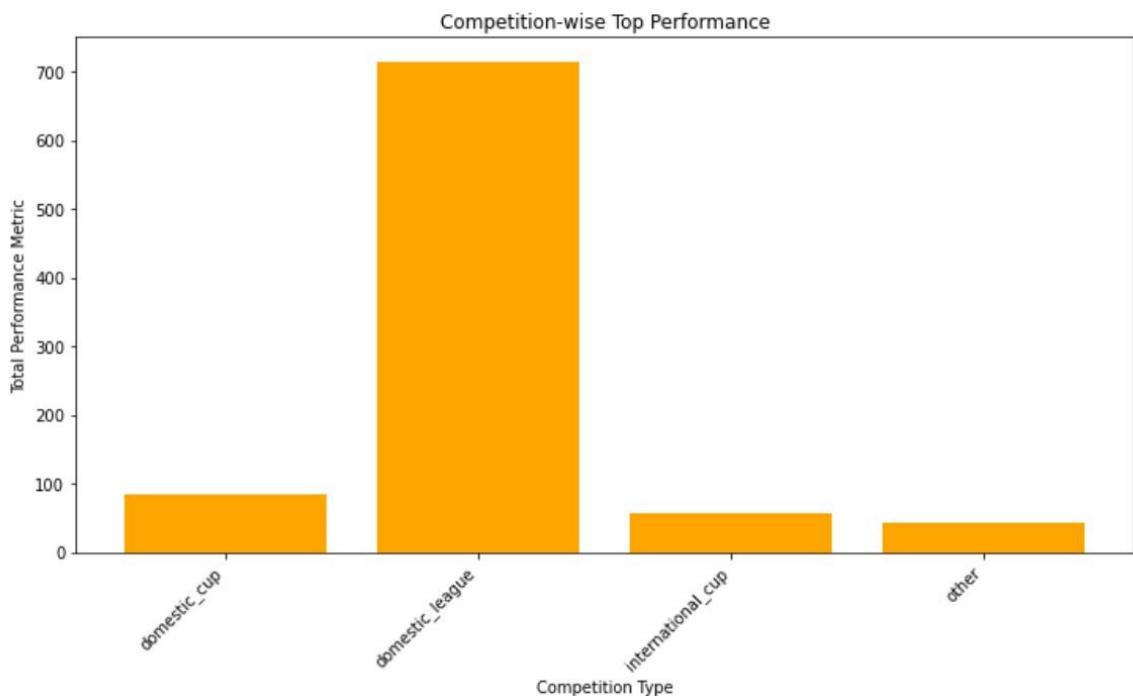
```

top_performers =
competition_top_performers.loc[competition_top_performers.groupby('competition_type')
['performance_metric'].idxmax()]

# Plot the results
plt.figure(figsize=(12, 6))

plt.bar(top_performers['competition_type'], top_performers['performance_metric'],
color='orange') plt.title('Competition-wise Top Performance') plt.xlabel('Competition
Type') plt.ylabel('Total Performance Metric') plt.xticks(rotation=45, ha='right')
plt.show()

```



Sql –

```

-- competition wise top performance
SELECT competition_type,home_club_name ,AVG(goals) from
football_data_merged_cleaned
WHERE goals IS NOT NULL GROUP BY competition_type ORDER BY AVG(goals)
DESC;

```

	competition_type	home_club_name	AVG(goals)
▶	other	Borussia Dortmund	0.6667
	domestic_cup	Eintracht Frankfurt Fußball AG	0.5177
	domestic_league	1.FC Nuremberg	0.4348
	international_cup	Verein für Leibesübungen Wolfsburg	0.3731

Interpretation –

Other competitions have maximum average goals whereas Domestic cup competition have maximum number of goals.

---

Q3 .

Does the intensity of play vary between domestic and international competitions?

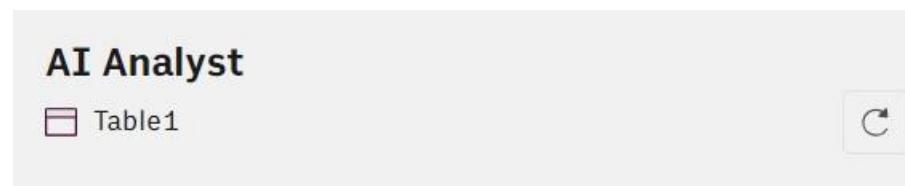
Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70
3	1	2231426_497	2231426	49723 L1	0	0	0	0	70

AI analyst popup will open.



Ask AI a question about your data...

### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Does the intensity of play vary between domestic and international competitions?**

Click on enter option.

Does the intensity of play vary between domestic and international competitions?

In deep dives, Intensity of Play in different Competitions table will appear, click on insert table option.

### Deep dives

Number of Games in Different Competitions

Intensity of Play in Different Competitions ...

Team Performance by Competition

+ Insert table

Copy formula

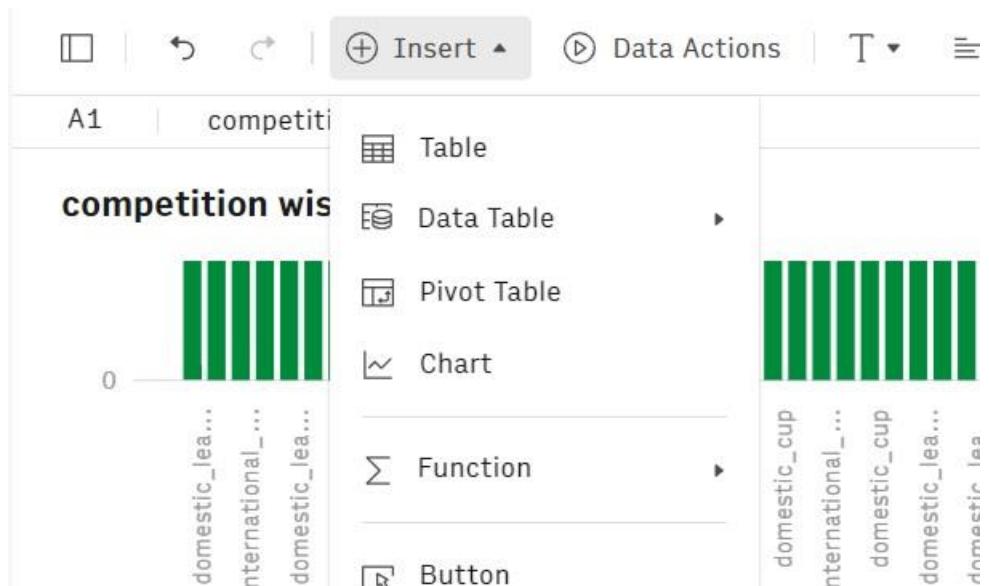
Number of Appearances per Game

You can see table as-

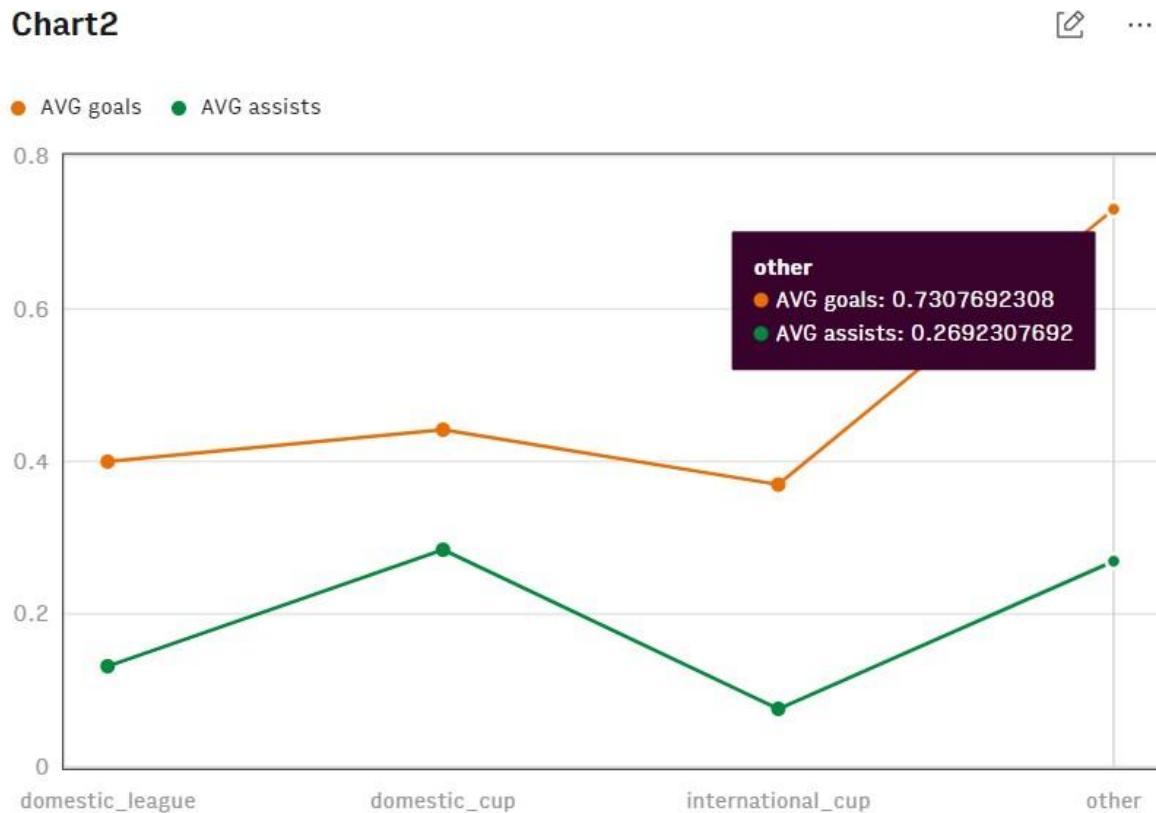
### Intensity of Play in Different Competitions

1	A	B	C	+
1	competition_ty	AVG goals	AVG assists	
2	domestic_leagu	0.3997672389	0.1315100378	
3	domestic_cup	0.4416243655	0.2842639594	
4	international_c	0.3697478992	0.0756302521	
5	other	0.7307692308	0.2692307692	

Select table and insert chart.



You can see the line chart as below -



Rename the chart title as competition wise play intensity

## competition wise play intensity

...



**SQL -**

```
-- competition wise play intensity
```

```
SELECT competition_type, AVG(goals), AVG(assists) from football_data_merged_cleaned
WHERE player_id IS NOT NULL GROUP BY competition_type
```

	competition_type	AVG(goals)	AVG(assists)
▶	domestic_league	0.4348	0.1342
	domestic_cup	0.5177	0.1986
	international_cup	0.3731	0.0896
	other	0.6667	0.3333

**Python** import pandas as pd

```
import matplotlib.pyplot as plt
```

```
# Assume df is your DataFrame containing football match summary data
```

```
# Calculate average goals for each competition
```

```
competition_avg_goals = df.groupby('competition_type')['home_club_goals',
```

```

'away_club_goals'].mean().reset_index()

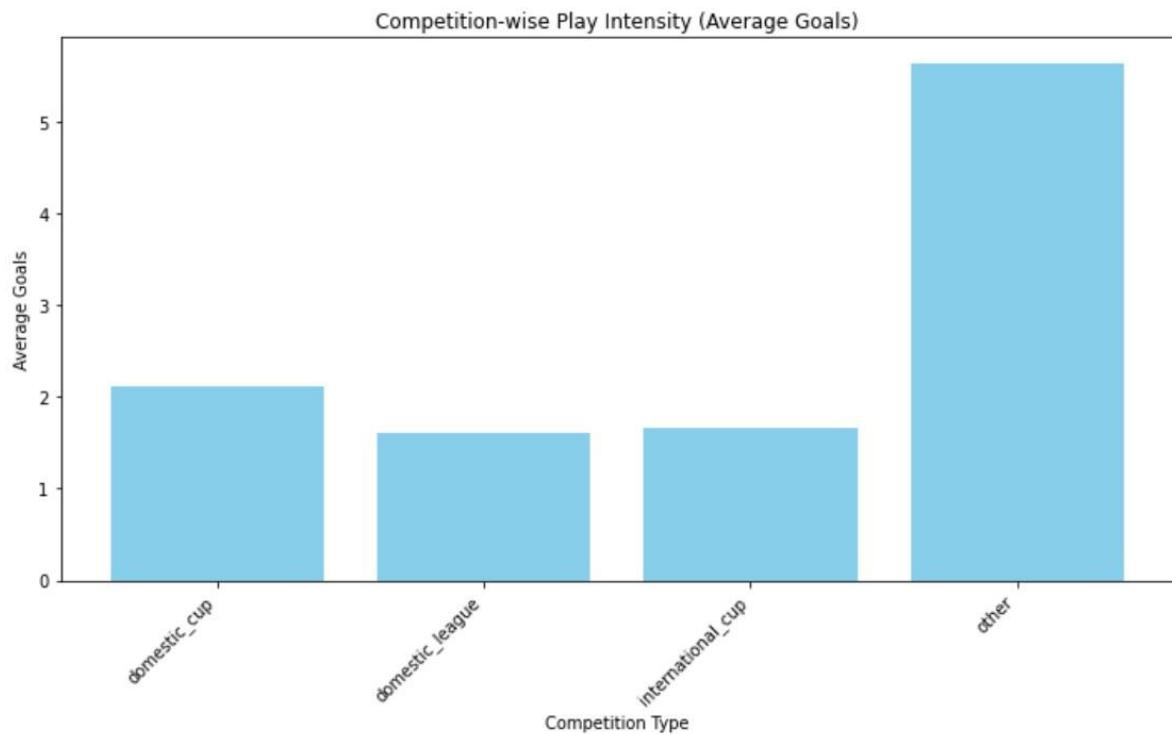
competition_avg_goals['average_goals'] = (competition_avg_goals['home_club_goals'] +
competition_avg_goals['away_club_goals']) / 2

# Plot the results
plt.figure(figsize=(12, 6))

plt.bar(competition_avg_goals['competition_type'], competition_avg_goals['average_goals'],
color='skyblue') plt.title('Competition-wise Play Intensity (Average Goals)')
plt.xlabel('Competition Type') plt.ylabel('Average Goals') plt.xticks(rotation=45, ha='right')

plt.show()

```



Interpretation –

Other competitions have maximum average goals whereas

### **Business Objective 9: Player Attributes and Demographics:**

What role do player attributes (foot preference, height) play in their on-field performance?

- ❑ Are there trends in player demographics (country of birth) and their impact on team success?
- ❑ How do player attributes contribute to specific roles on the field?

Q1 . What role do player attributes (foot preference, height) play in their on-field performance?

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
0	2231426_497	2231426	49723	L1	0	0	0	0	70
1	2231426_497	2231426	49723	L1	0	0	0	0	70

AI analyst popup will open.

**AI Analyst**

Table1

Ask AI a question about your data...

**Quick insights**

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**What role do player attributes (foot preference, height) play in their on-field performance?** Click on enter option.

What role do player attributes (foot preference, height) play in their on-field performance?

In deep dives you can see a table with name Average Performance by Foot Preference. Click on Insert Table option.

## Deep dives

Average Performance by Foot Preference ...

Number of Games in Different Com  Insert table  Copy formula

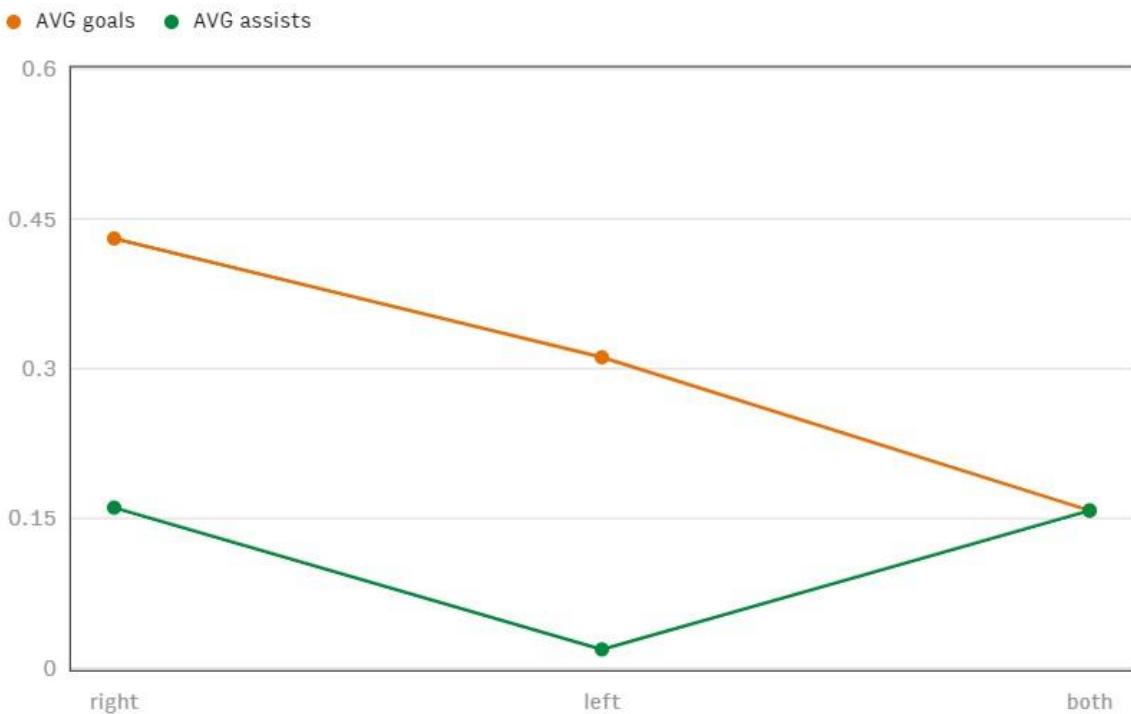
Average Performance by Foot Preference table will appear. Select table and insert chart.

### Average Performance by Foot Preference

	A	B	C	+
1	foot	AVG goals	AVG assists	
2	right	0.4304075235	0.1608150470	
3	left	0.3114241002	0.0187793427	
4	both	0.1578947368	0.1578947368	

Line chart will look like as -

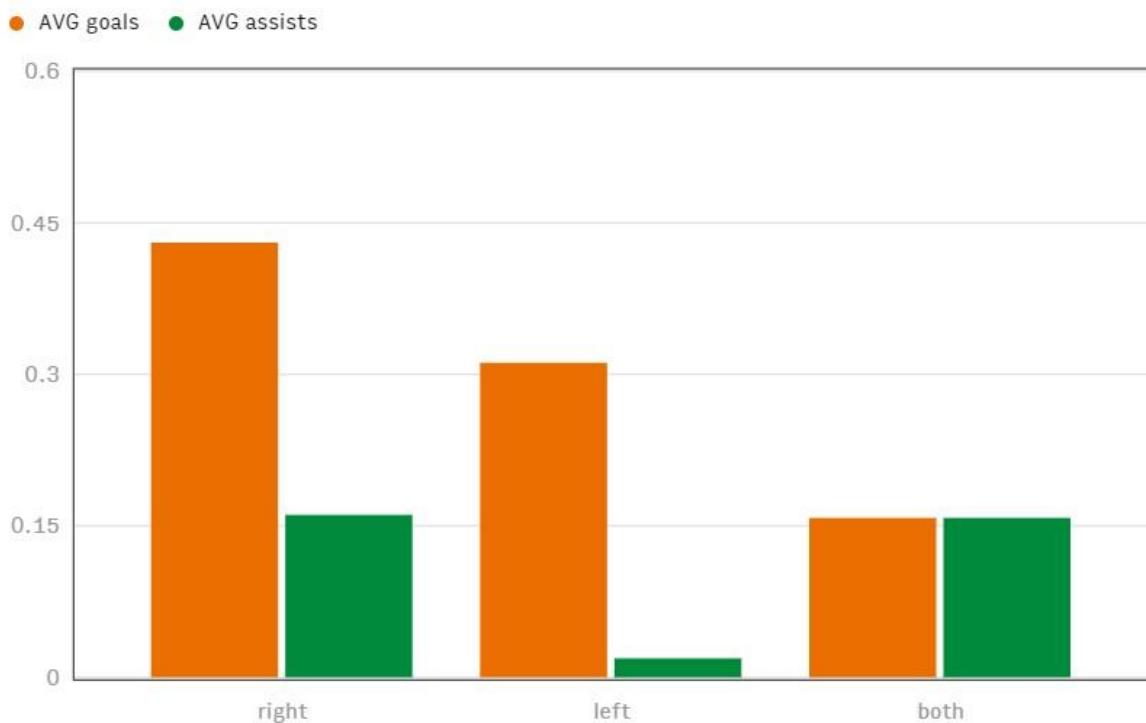
### Chart2



Rename the title of chart as - Average performance by feet

## Average performance by feet

 ...



SQL-

In MySQL workbench run the following query.

-- Average performance by feet

```
SELECT foot, AVG(goals), AVG(assists) from football_data_merged_cleaned
```

```
WHERE player_id IS NOT NULL AND foot IS NOT NULL GROUP BY foot;
```

Result -

Result Grid     Filter Rows: <input type="text"/>			
	foot	AVG(goals)	AVG(assists)
▶	right	0.4653	0.1728
▶	left	0.3497	0.0211

Python –

In python run the following code in jupyter notebook.

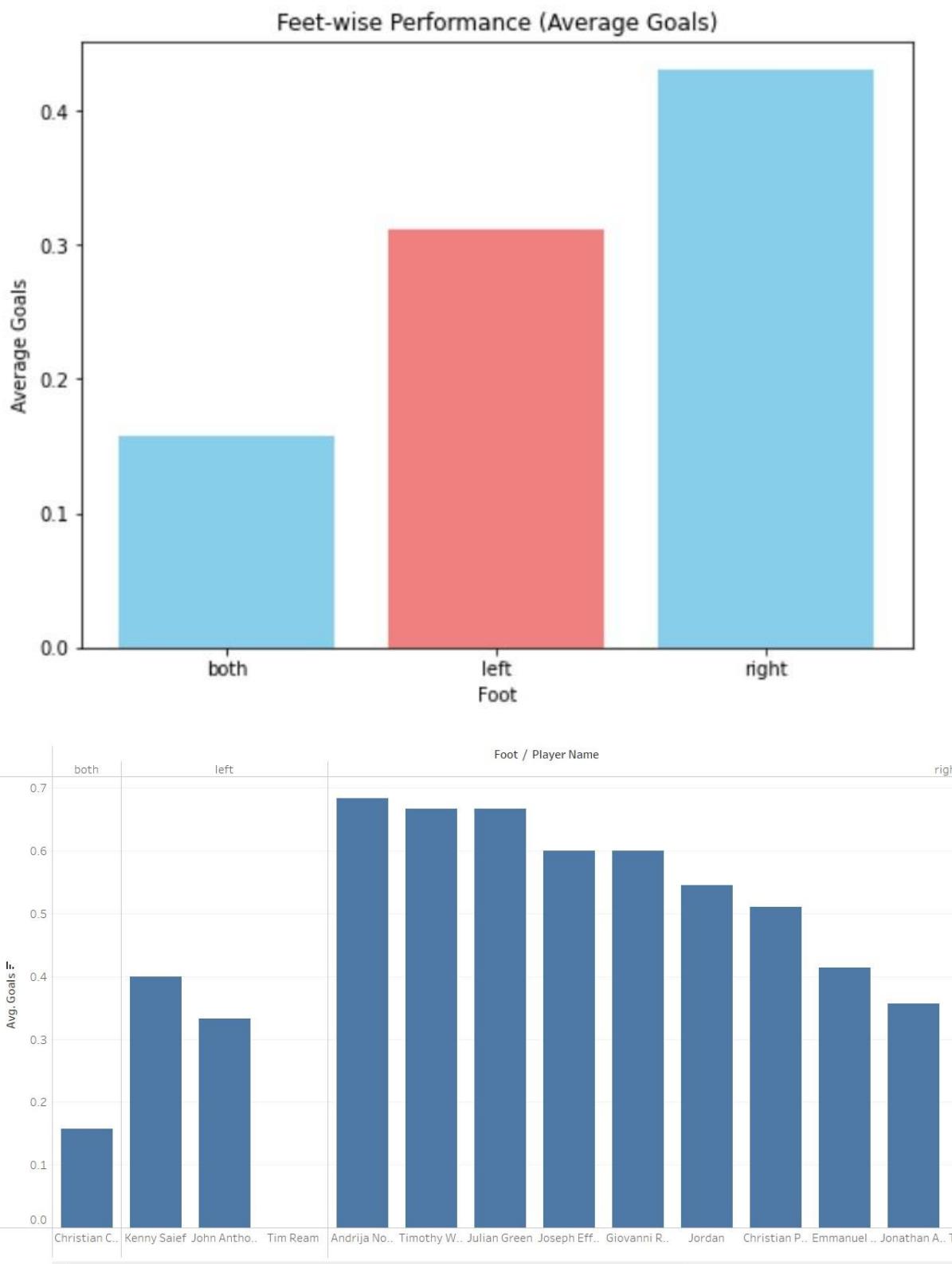
```
import pandas as pd import matplotlib.pyplot as plt
```

```
# Assume df is your DataFrame containing football match summary data
```

```
# Filter relevant columns feet_performance_df
= df[['foot', 'goals']]

# Group by foot and calculate the average goals for each foot
average_goals_by_foot = feet_performance_df.groupby('foot')['goals'].mean().reset_index()

# Plot the results
plt.figure(figsize=(8, 6))
plt.bar(average_goals_by_foot['foot'], average_goals_by_foot['goals'], color=['skyblue', 'lightcoral'])
plt.title('Feet-wise Performance (Average Goals)')
plt.xlabel('Foot') plt.ylabel('Average Goals')
plt.show()
```



#### Interpretation-

Maximum players shoot the goal with right feet followed by Left. Only some players make goals with both foots.

---

Q2 .

Are there trends in player demographics (country of birth) and their impact on team success? Using Rows.com –

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723 L1	0	0	0	0	70
3	1	2231426_497	2231426	49723 L1	0	0	0	0	70

AI analyst popup will open.

**AI Analyst**

Table1

Ask AI a question about your data...

### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Are there trends in player demographics (country of birth) and their impact on team success?**

Click on enter option.

Ask Are there trends in player demographics (country of birth) and their impact on team success?

In deep dives you can see one table with name Player Demographics and Average goals Scored is appeared. Click on insert Table option

## Deep dives

- Player Demographics and Average Assists
- Player Demographics and Average Goals Scored ...
- Player Demographics and Average ...
  - + Insert table
  - Copy formula

Player Demographics and Average goals Scored table will create below dataset. Select Table and insert chart.

### Player Demographics and Average Goals Scored <>

A	B	+
1	country_of_bir	AVG goals
2	England	0.6
3	United States	0.4238185255
4	Italy	0.4146341463
5	Germany	0.3360995851
6	Netherlands	0

□ | ⏪ ⏩ | + Insert ▲ ◎ Data Actions | T ▾

A1 | country\_

**Average performance**

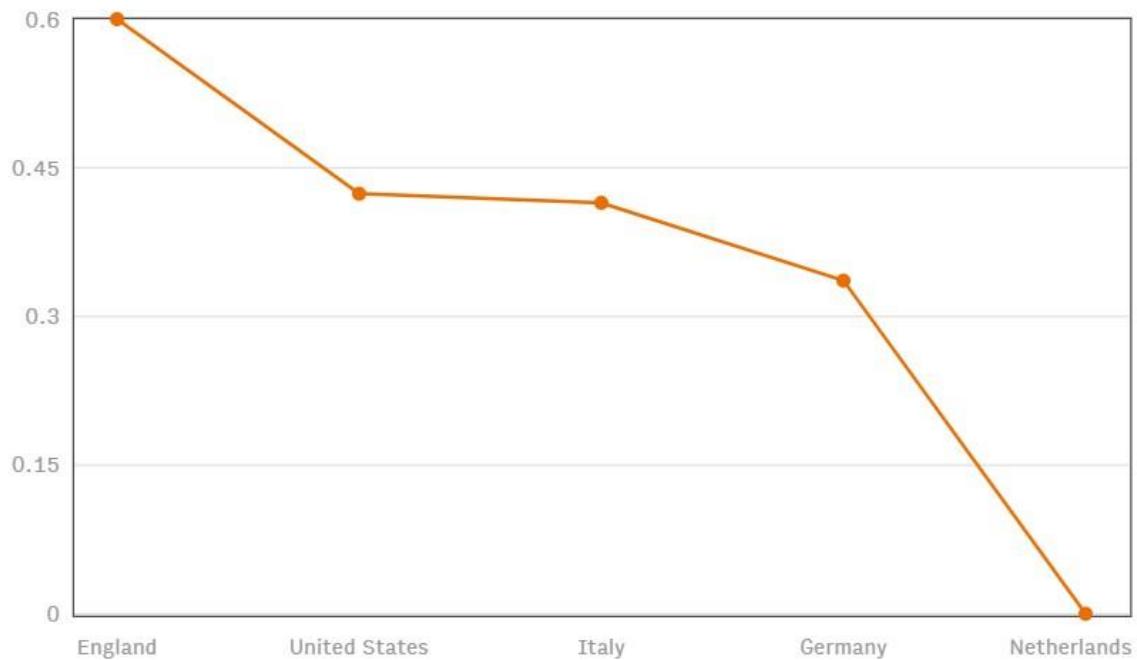
0.15  
0

- Table
- Data Table
- Pivot Table
- Chart
- Function
- Button

The output will look like this -

**Chart2**

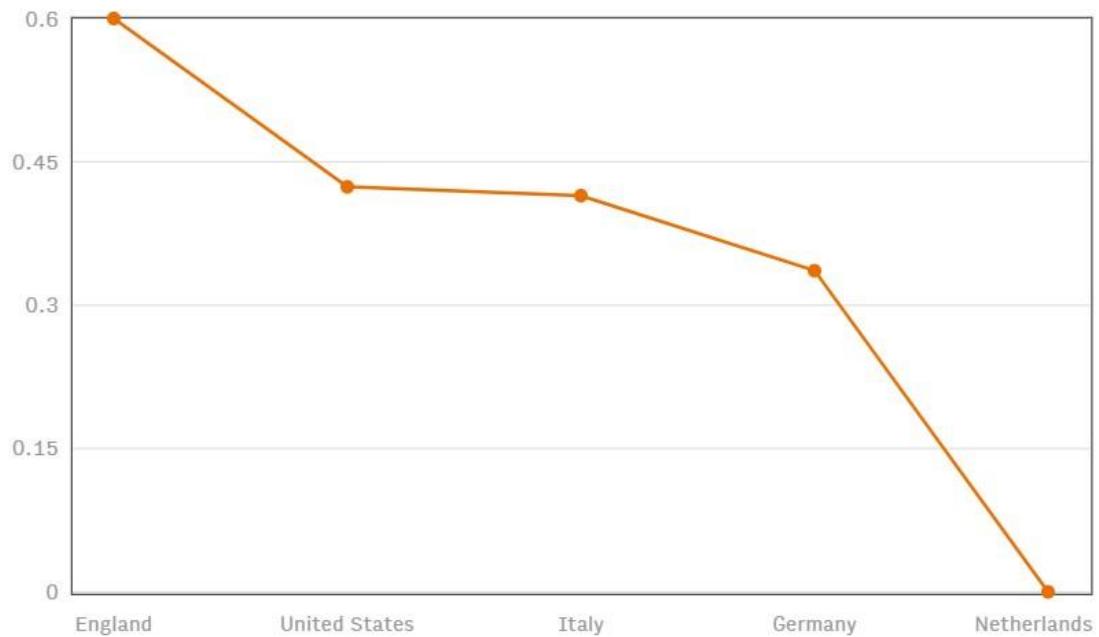
...



Rename the chart title as - Player democracies wise average goals

**Player democracies wise average goals**

...



Using SQL-

In MySQL workbench Run the following query -

-- player democracy wise average goals

```
SELECT country_of_birth, AVG(goals) from football_data_merged_cleaned GROUP BY country_of_birth ORDER BY AVG(goals) DESC;
```

	country_of_birth	AVG(goals)
▶	United States	0.4812
	Germany	0.3361

Using Python –

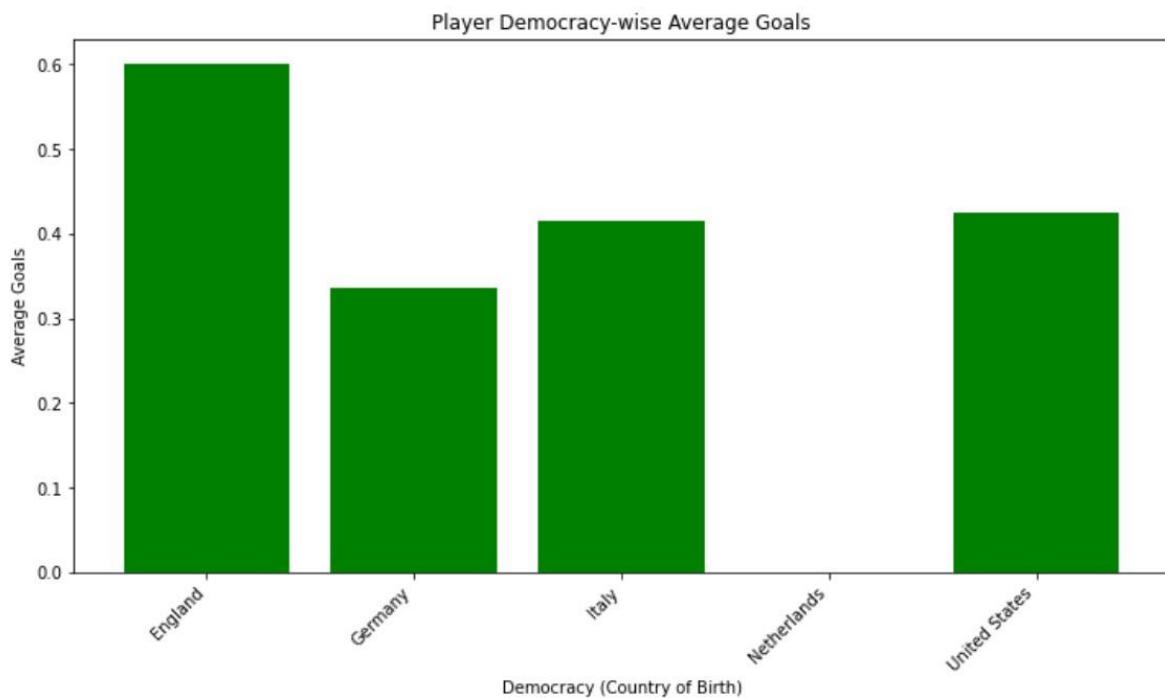
In jupyter notebook run the following code.

```
import pandas as pd import
matplotlib.pyplot as plt

# Assume df is your DataFrame containing football match summary data

# Group by country_of_birth and calculate average goals player_avg_goals =
df.groupby('country_of_birth')['goals'].mean().reset_index()

# Plot the results plt.figure(figsize=(12, 6))
plt.bar(player_avg_goals['country_of_birth'], player_avg_goals['goals'], color='green')
plt.title('Player Democracy-wise Average Goals') plt.xlabel('Democracy (Country of Birth)')
plt.ylabel('Average Goals') plt.xticks(rotation=45, ha='right') plt.show()
```



Interpretation -

Country England have maximum average number of goals followed by Italy and United State.

---

Q3. How do player attributes contribute to specific roles on the field?

Using Rows.com

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

	A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played	
2	0	2231426_497	2231426	49723	L1	0	0	0	0	70
3	1	2231426_497	2231426	49723	L1	0	0	0	0	70

AI analyst popup will open.

## AI Analyst

Table1



Ask AI a question about your data...



### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**How do player attributes contribute to specific roles on the field?**

Click on enter option.

How do player attributes contribute to specific roles on the field?



In deep dives Average Player attributes by Position and Sub-Position table will appear. Click on insert table option.

### Deep dives

Average Player Attributes for Left-Footed Players  
by Position

Average Player Attributes for Right-Footed Players  
by Position

Average Player Attributes by Position

Insert table

Copy formula

Below data you can see the table with name

### Average Player Attributes by Position and Sub-Position

A	B	C	D	E		
1	player_id	position_x	AVG height_in_	AVG market_value_in_eur	AVG highest_market_value_in_eur	+
2	49723	Defender	186	700000	5000000	
3	124732	Defender	194	2000000	15000000	
4	161204	Midfield	173	1000000	2000000	
5	209019	Attack	176	700000	3500000	
6	355369	Attack	191	7000000	10000000	
7	315779	Attack	177	25000000	60000000	

Select table and insert chart.

The screenshot shows the Power BI interface with a table named "Player demographic" selected. A context menu is open, with the "Chart" option highlighted. Other options visible in the menu include "Table", "Data Table", "Pivot Table", and "Function".

In Chart editor on right side, select position\_x in main axis, AvG market\_value\_in\_eur and AVG highest\_market\_value\_in\_eur in vertical left axis.

Main axis

position\_x

Vertical left axis

AVG market\_value\_in\_eur

AVG highest\_market\_value\_in\_eur

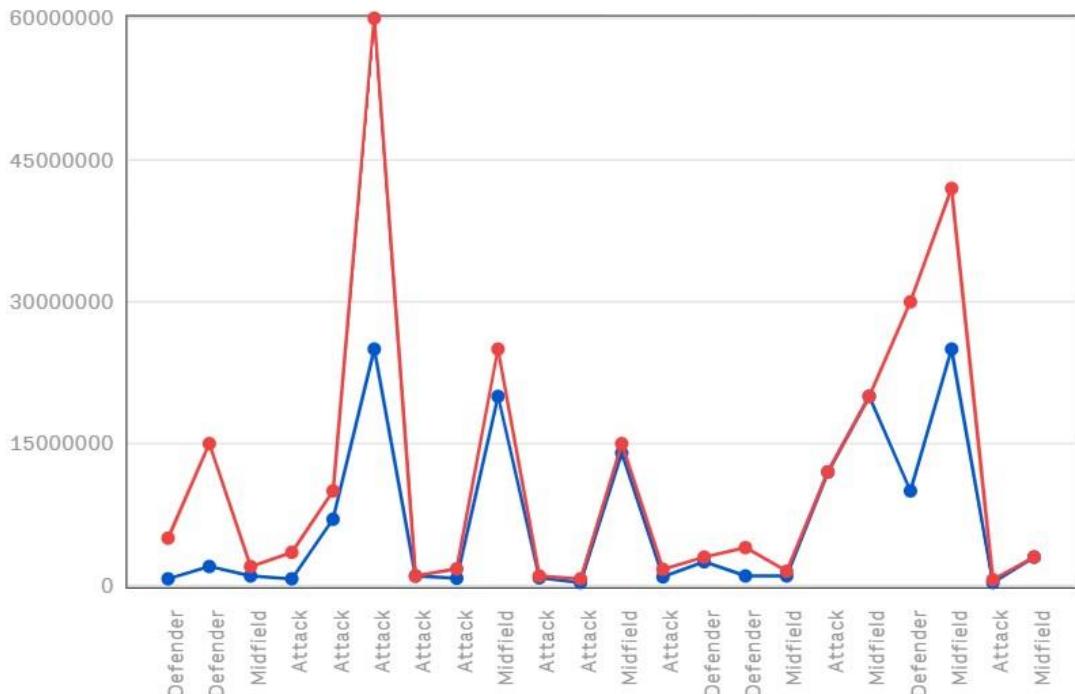
(+) Add series

Rename the line chart as - Position wise player attribute

## Position wise player attribute

...

● AVG market\_value\_in\_eur ● AVG highest\_market\_value\_in\_eur



Using SQL-

In MySQL workbench Run the following query -

-- Average player attribute by position

```
SELECT position_x, AVG(market_value_in_eur), AVG(highest_market_value_in_eur) from football_data_merged_cleaned
```

```
WHERE player_id IS NOT NULL AND position_x IS NOT NULL GROUP BY position_x;
```

	position_x	AVG(market_value_in_eur)	AVG(highest_market_value_in_eur)
▶	Defender	1484388.1856540085	11033755.274261603
	Midfield	18787234.04255319	23531914.893617023
	Attack	19841308.29800121	46683222.2895215

Using Python –

In Jupyter notebook run the following code -

```
import pandas as pd import
```

```
matplotlib.pyplot as plt
```

```

# Assume df is your DataFrame containing football match summary data

# Filter relevant columns filtered_df = df[['position_x',
'height_in_cm', 'market_value_in_eur']]

# Convert height_in_cm to meters for better readability filtered_df['height_in_m']
= filtered_df['height_in_cm'] / 100

# Convert market_value_in_eur to millions for better readability
filtered_df['market_value_in_million_eur'] = filtered_df['market_value_in_eur'] / 1e6

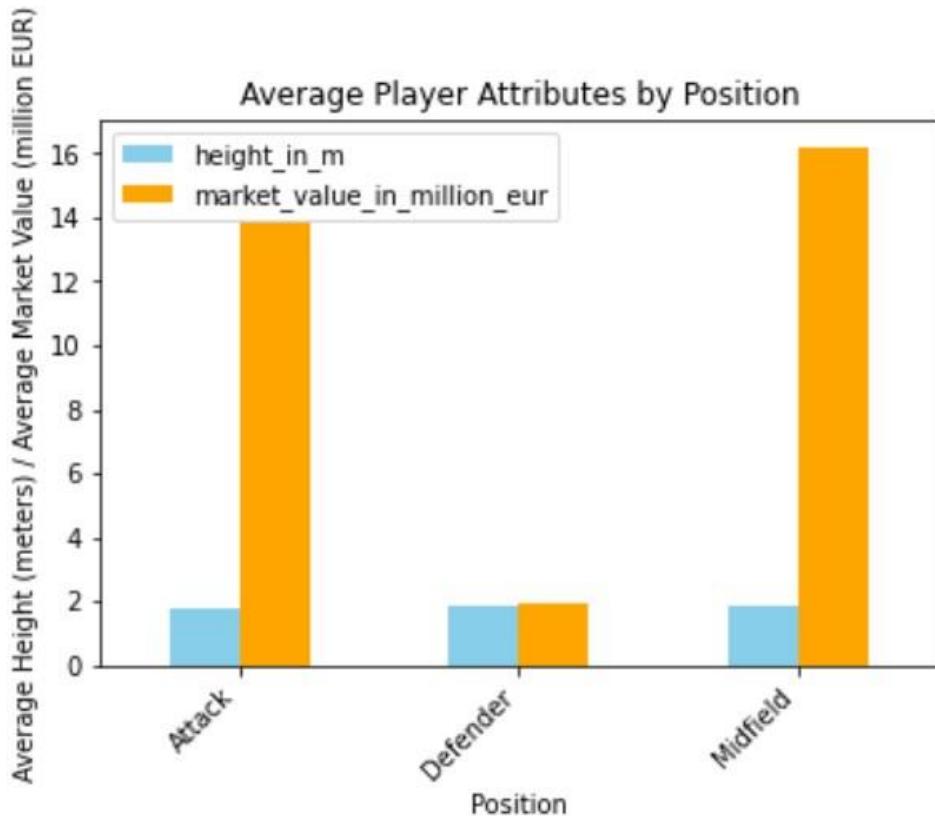
# Group by position and calculate the average height and market value
avg_attributes_by_position = filtered_df.groupby('position_x').agg({
    'height_in_m': 'mean',
    'market_value_in_million_eur': 'mean'
}).reset_index()

# Plot the results
plt.figure(figsize=(12, 6))

avg_attributes_by_position.plot(kind='bar', x='position_x', y=['height_in_m',
'market_value_in_million_eur'],
color=['skyblue', 'orange'], legend=True)

plt.title('Average Player Attributes by Position') plt.xlabel('Position')
plt.ylabel('Average Height (meters) / Average Market Value (million
EUR)') plt.xticks(rotation=45, ha='right') plt.show()

```



Interpretation –

For attack and Midfield position players average market value is more as compared to Defender.

---

Q. Identify groups of players based on their attributes and demographics.

Solution - k-means clustering df\_temp = df

# label\_encoder object knows how to understand word labels.

```
label_encoder = preprocessing.LabelEncoder()
```

```
# Encode labels comp_encode=
```

```
label_encoder.fit_transform(df_temp[['country_of_birth']])
```

```
print(comp_encode) sea_encode=
```

```
label_encoder.fit_transform(df_temp[['foot']]) print(comp_encode)
```

```
print(label_encoder.classes_)
```

```
# Assuming 'attributes' and 'demographics' are the columns you want to cluster on
```

```
X = df_temp[['height_in_cm', 'foot','country_of_birth']]
```

```
X
```

```

# Standardize the features by scaling them scalar
scaler = StandardScaler()

X['foot'] = X['foot'].astype(float)
X['height_in_cm'] = X['height_in_cm'].astype(float)
X['country_of_birth'] = X['country_of_birth'].astype(float)

X = X[['height_in_cm', 'foot','country_of_birth']]

cols = X.columns df_scale =
scaler.fit_transform(X) df_scale =
pd.DataFrame(df_scale, columns=cols)
df_scale.head()

X = df_scale[['height_in_cm', 'country_of_birth']]

X
Kmeans_model = KMeans(n_clusters=2)

Kmeans_model.fit(X)
cluster_centroid = Kmeans_model.cluster_centers_.print(cluster_centroid)
labels = Kmeans_model.labels_.print(labels) plt.scatter(X['height_in_cm'],
X['country_of_birth'], marker='.', c='orange') plt.scatter(-0.63793923, -
0.63793923, s=50, c='r', marker='s') plt.scatter(1.19906346, 1.19906346,
s=50, c='g', marker='s')
#plt.scatter(0.94397247, -1.55832725, s=50, c='r', marker='s')
#plt.scatter(3.69780107, 3.19156511, s=50, c='g', marker='s') plt.show()

```

## Rebuild the Model With the Optimum K-Value

```

X_scaled = X

# Determine the optimal number of clusters using the elbow method #
Note: You can change the range of clusters according to your dataset
wcss = [] for i in range(1, 11):
kmeans = KMeans(n_clusters=i, init='k-means++', random_state=42)
kmeans.fit(X_scaled) wcss.append(kmeans.inertia_)

```

```

# Plot the elbow method graph to determine the optimal number of clusters
plt.plot(range(1, 11), wcss) plt.title('Elbow Method') plt.xlabel('Number of
clusters') plt.ylabel('WCSS') plt.show()

# Based on the elbow method, choose the optimal number of clusters and perform K-means
clustering k = 2 # Enter the optimal number of clusters kmeans = KMeans(n_clusters=k,
init='k-means++', random_state=42) clusters = kmeans.fit_predict(X_scaled)

# Add the cluster labels to the original dataset data['Cluster']
= clusters

# Now you have the clustered data, and you can analyze each cluster accordingly print(data.head())

# You can also perform further analysis on each cluster, such as descriptive statistics
print(data.groupby('Cluster').mean())

# Visualize the clusters (assuming you have 2D data) plt.scatter(X_scaled['height_in_cm'],
X_scaled['country_of_birth'], c=clusters, cmap='viridis')
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s=300, c='red', marker='*',
label='Centroids') plt.title('Clusters of Players') plt.xlabel('Height') plt.ylabel('Country of birth')
plt.legend() plt.show()

```

### **Business Objective 10 : Contract Management:**

How does the contract status of players impact their on-field performance?

Are there patterns in player performance leading up to the expiration of their contracts?

What strategies can be employed to manage contract expiration risks and uncertainties?

¶

Q1 How does the contract status of players impact their on-field performance? Solution

By Using Rows.com –

Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	70
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.

**AI Analyst**

Table1

Ask AI a question about your data...

### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**How does the contract status of players impact their on-field performance?**

Click on enter option.

Ask How does the contract status of players impact their on-field performance?

In Quick insights you can see the result for contract status and average market value of player, minutes played, number of assists and average goals.

## Quick insights

The average market value of players with a contract is 11,558,537.21.

The average minutes played by players with a contract is 69.96.

The average number of assists by players with a contract is 0.14.

The average number of goals scored by players with a contract is 0.40.

Using SQL –

-- average market value

```
SELECT player_id, avg(goals), avg(minutes_played), avg(market_value_in_eur) from football_data_merged_cleaned
```

```
WHERE contract_expiration_date IS NOT NULL GROUP BY player_id ORDER BY contract_expiration_date DESC;
```

	player_id	avg(goals)	avg(minutes_played)	avg(market_value_in_eur)
▶	315779	0.5114	73.7955	25000000
	49723	0.3404	72.0000	700000
	332697	0.0000	59.7273	20000000
	124732	0.3333	71.7879	2000000
	161204	0.6667	74.3333	1000000
	209019	0.4000	66.2286	700000
	355369	0.5455	56.8182	7000000

Using Python – import

pandas as pd

```
# Assume df is your DataFrame containing football match summary data
```

```
# Filter relevant columns
```

```
player_stats = df[['player_id', 'name', 'market_value_in_eur', 'minutes_played']]
```

```
# Group by player and calculate average market value and minutes played average_stats
= player_stats.groupby(['player_id', 'name']).agg({
    'market_value_in_eur': 'mean',
    'minutes_played': 'mean'
}).reset_index()
```

# Display the results

```
print(average_stats)
```

	player_id	name	market_value_in_eur	minutes_played
0	49723	Timothy Chandler	700000.0	72.000000
1	103064	Terrence Boyd	750000.0	63.333333
2	124732	John Anthony Brooks	2000000.0	71.787879
3	145466	Tim Ream	1000000.0	48.600000
4	161204	Julian Green	1000000.0	74.333333
5	209019	Kenny Saief	700000.0	66.228571
6	282199	Erik Palmer-Brown	2500000.0	73.000000
7	307781	Lynden Gooch	1000000.0	66.333333
8	315762	Luca de la Torre	3000000.0	58.000000
9	315779	Christian Pulisic	25000000.0	73.795455
10	332697	Weston McKennie	20000000.0	70.896552
11	332705	Tyler Adams	20000000.0	68.769231
12	336160	Andrija Novakovich	900000.0	80.947368
13	355369	Jordan	7000000.0	56.818182
14	361104	SergiÃ±o Dest	10000000.0	74.666667
15	370846	Timothy Weah	14000000.0	52.444444
16	381187	Emmanuel Sabbi	800000.0	75.975610
17	393325	Josh Sargent	12000000.0	65.592593
18	504215	Giovanni Reyna	25000000.0	65.600000
19	537467	Joseph Efford	300000.0	65.400000
20	542776	Jonathan Amon	300000.0	65.000000
21	544931	Christian Cappis	1000000.0	62.947368

Interpretation –

1. The average market value of players with a contract is 11,558,537.21.
2. The average minutes played by players with a contract is 69.96.
3. The average number of assists by players with a contract is 0.14.
4. The average number of goals scored by players with a contract is 0.40.

Q2 . Are there patterns in player performance leading up to the expiration of their contracts?

Using Rows.com –

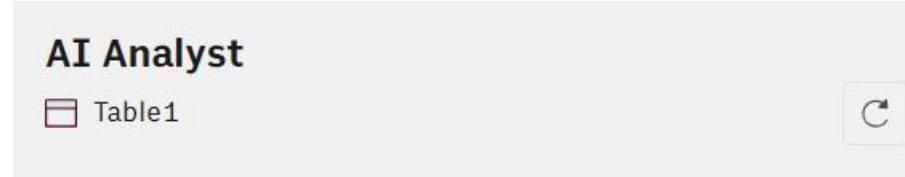
Step 1 :

Click on AI Analyst symbol to analyze dataset.

Table1

A	B	C	D	E	F	G	H	I	J
1	appearance_id	game_id	player_id	competition_id	yellow_cards	red_cards	goals	assists	minutes_played
2	0	2231426_497	2231426	49723	L1	0	0	0	0
3	1	2231426_497	2231426	49723	L1	0	0	0	70

AI analyst popup will open.



Ask AI a question about your data...

### Quick insights

The total number of appearance IDs is 3,924.

The total number of unique game IDs is 336.

The highest number of goals scored is 3.

The average minutes played per game is 69.96.

Type here prompt as –

**Are there patterns in player performance leading up to the expiration of their contracts?**

Click on enter option.

player performance leading up to the expiration of  
their contracts

You can see the table for Player performance VS Contract Status.

## Player Performance vs Contract Status

	A	B	C	D	E
1	player_id	goals	assists	minutes_playe	market_value_i
2	370846	1	0	90	14000000
3	370846	1	0	90	14000000
4	370846	1	0	90	14000000
5	370846	1	0	90	14000000
6	370846	1	0	90	14000000
7	370846	1	0	90	14000000
8	370846	1	0	90	14000000
9	370846	1	0	90	14000000

Using SQL-

In MySQL run following query.

-- player performance vs contract

```
SELECT player_id, sum(goals), sum(minutes_played), avg(market_value_in_eur) from
football_data_merged_cleaned
```

```
WHERE contract_expiration_date IS NOT NULL GROUP BY player_id ORDER BY
contract_expiration_date DESC;
```

	player_id	sum(goals)	sum(minutes_played)	avg(market_value_in_eur)
▶	315779	630	90916	25000000
	49723	96	20304	700000
	332697	0	5256	20000000
	124732	143	30797	2000000
	161204	4	446	1000000
	209019	56	9272	700000
	355369	144	15000	7000000
	307781	0	199	1000000

Q3 - What strategies can be employed to manage contract expiration risks and uncertainties?

Using Python-

Run the following code in Jupyter notebook.

```
import pandas as pd import
matplotlib.pyplot as plt import
seaborn as sns
```

```

# Assume df is your DataFrame containing football match summary data

# Select relevant columns for analysis
contract_data = df[['player_id', 'name', 'contract_expiration_date', 'market_value_in_eur',
'current_club_id']]

# Convert 'contract_expiration_date' to datetime format
contract_data['contract_expiration_date'] =
pd.to_datetime(contract_data['contract_expiration_date'])

# Calculate the remaining days until contract expiration
contract_data['days_until_expiration'] = (contract_data['contract_expiration_date'] -
pd.to_datetime('today')).dt.days

# Visualize distribution of remaining days until expiration
plt.figure(figsize=(10, 6))
sns.histplot(contract_data['days_until_expiration'], bins=20, kde=True)
plt.title('Distribution of Remaining Days until Contract Expiration')
plt.xlabel('Days until Contract Expiration') plt.ylabel('Count')
plt.show()

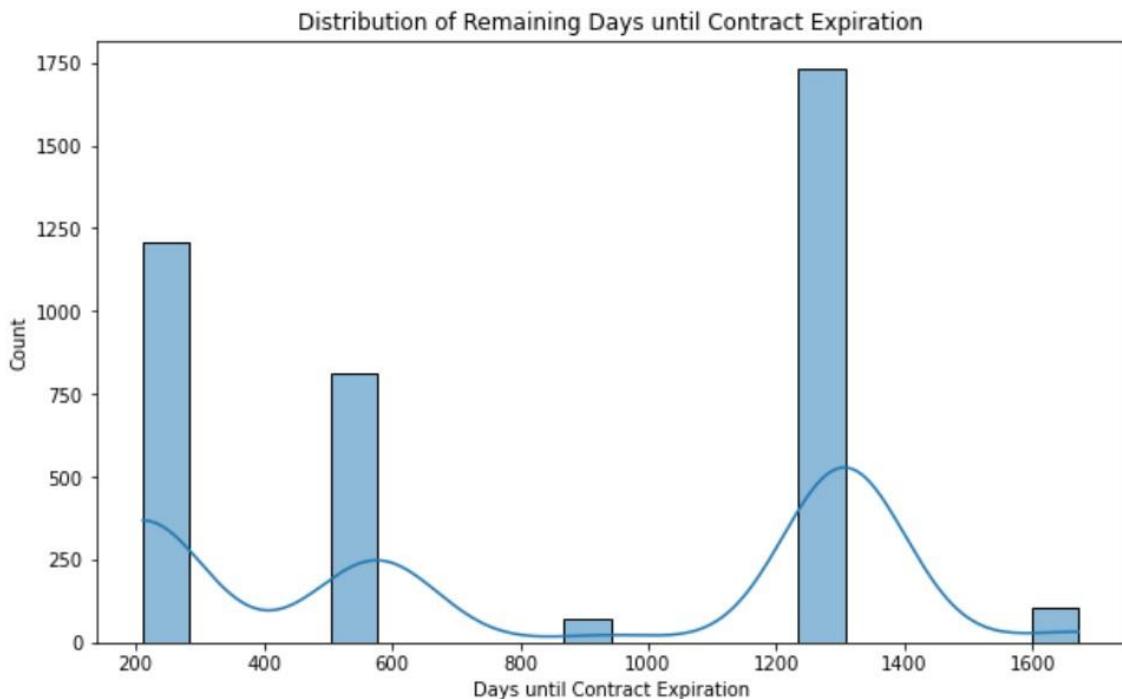
# Analyze players with contracts expiring soon threshold_days = 90 # You can adjust
this threshold based on your definition of "soon"
players_expiring_soon = contract_data[contract_data['days_until_expiration'] <=
threshold_days]

# Display players expiring soon print("Players with Contracts Expiring
Soon:") print(players_expiring_soon[['player_id', 'name',
'days_until_expiration']])

# Explore strategies to manage contract expiration risks
# Example: Consider offering contract extensions, negotiations, or scouting new players.

```

# Additional analysis and strategies can be added based on your specific dataset and goals.



Interpretation –

In next 6 months none of the players contract will expire. But maximum number of player's contract will end after 3.5 year. After six months approximately 1240 players will end with their contract.

---

### Summarization of all Interpretations

- Total number of unique game IDs is 336.
- Average number of goals scored is 0.40.
- Maximum number of away club goals is 7.
- Key attributes of players are average goals and assists, yellow and red cards.
- Players: Christian (Id: 315779), Emmanuel (ID: 381187), Jordon (ID: 355369) are the top goal scorers. Christian (Id: 315779) has the highest minutes played.

- Players: Christian (Id: 315779), Jonathan (ID: 542776), Jordon (ID: 355369) have the highest assists.
- Christian, Weston and Jordan have the highest market values. Emmanuel and Jonathan have low market values, even when they are top goal scorer and have highest assists respectively.
- Teams have highest average goals under the influence of Home club manager named 'Sean Dyche'.
- Maximum number of goals i.e. 126 have been scored by the team under the influence of Burnley Football Club (home\_club\_name) and Sean Dyche (home\_club\_manager\_name).
- Sean Dyche is a manager with high effectiveness followed by Lucien Favre.
- Manchester City Football Club (home\_club\_name) scored maximum average home club goals and SC Paderborn 07 (home\_club\_name) scored maximum average away club goals.
- There is a positive correlation of home club position with performance metrics: minutes played, goals and yellow cards while it has negative correlation with assists.
- The stadium ‘Santiago BernabAcu’ has maximum average attendance, i.e. 76894 and stadium ‘HelsingA r Stadion’ has minimum average attendance, i.e. 1119. Specific stadiums consistently attract higher attendance. Example, the stadiums such as ‘Santiago BernabAcu’, ‘Allianz Arena’, ‘Signal Iduna Park’ have very high average attendance as compared to other stadiums.
- The stadium ‘Signal Iduna Park’ has maximum total number of home club goals and maximum number of matches.
- Yellow cards has correlation with ‘goals’ = 0.03, with ‘home\_club\_goals’ = -0.069 and with ‘away\_club\_goals’ = -0.005. This shows yellow cards have not much impact on match outcomes (goals).
- Maximum yellow cards have been received by players in matches officiated by referees ‘Felix’, ‘Mikkel’, and ‘Guido’. Maximum goals scored in matches officiated by referees ‘Peter Kjaersgaard’ and ‘Jacracmy Stinat’. This means, few referees are associated with controversial and few referees are associated with high-scoring matches.
- Team ***NAC breda*** have maximum substitution time i.e. 89 minutes whereas team ***Arsenal Football Club*** have minimum substitution time i.e. 27 minutes.

- Team **Borussia Dortmund** have maximum player substitution frequency i.e. 494.s
- Correlation between goals and substitution is 0.02. Goals and cards also do not have much correlation.
- Attendance of people (match excitement) increases with increased number of goals in matches and increased number of competition types in an event.
- USC and DFL competitions have maximum average number of goals followed by DKP and NLP competitions.
- Domestic cup competition has maximum number of goals whereas other competitions have maximum average goals.
- Maximum players score the goal with right foot (such as Andrija, Timothy, Julian) followed by left foot (such as Kenny and John). Few players (such as Christian) score goals with both feet.
- Average market value is more for attack and midfield position players as compared to defender.
- The average market value of players with a contract is 11,558,537.21 (in euros).
- The average minutes played by players with a contract is 69.96.
- The average number of assists by players with a contract is 0.14.
- The average number of goals scored by players with a contract is 0.40.
- Contract expiration date with highest market values is in year 2027.
- None of the players' contract will expire in next 6 months. Maximum number of contracts will end after 3.5 years.
- Probability of a player scoring multiple goals is 2.91%
- Probability of players receiving a yellow card and scoring a goal is 6.5%
- Distribution of sample means for total minutes played by the players satisfy the properties of Central limit theorem.
- A player will receive a yellow card with 85% accuracy based on their previous performance metrics such as 'goals','assists','minutes\_played','yellow\_cards'.
- Errors are high when predict player's market value based on significant features such as height, minutes played, goals etc. using multiple linear regression.
- There is no significant difference in the average goals scored by the home clubs and the away clubs.

- There is no significant difference in average number of goals scored by players of left foot and right foot.

## Final Conclusion

In conclusion, the analysis of the provided summary reveals several key insights from the football dataset. The total number of unique game IDs is 336, with an average of 0.40 goals scored per game. Notably, Christian, Emmanuel, and Jordon emerge as the top goal scorers, with Christian also boasting the highest minutes played. Assists are prominently contributed by Christian, Jonathan, and Jordon, while players such as Christian, Weston, and Jordan hold the highest market values. Home club manager Sean Dyche demonstrates high effectiveness, correlating positively with various performance metrics.

The stadiums 'Santiago Bernabéu,' 'Allianz Arena,' and 'Signal Iduna Park' consistently attract higher attendance. Attendance shows a positive correlation with home club position, suggesting that stronger teams attract more fans. Referees play a role in match dynamics, with some associated with controversial matches and others with high-scoring encounters.

Substitution dynamics vary, with Team NAC Breda having the maximum substitution time, *Arsenal Football Club* having minimum substitution time, and team Borussia Dortmund exhibiting the highest player substitution frequency. Market values are higher for attack and midfield players, and contracts with the highest market values expire in 2027. The dataset also highlights that the majority of contracts will end after 3.5 years, with none expiring in the next six months. Overall, this comprehensive analysis provides valuable insights into player performance, team dynamics, and various factors influencing football matches and competitions.

Based on the analysis, it is found that certain player performance metrics, such as the likelihood of scoring multiple goals is 2.91% and receiving both a yellow card and scoring a goal is 6.5%. The Central Limit Theorem is validated through the distribution of sample means for total minutes played. Notably, a player's likelihood of receiving a yellow card can be predicted with 85% accuracy using their past performance data. However, predicting a player's market value using multiple linear regression is unreliable due to high errors. Furthermore, the data shows no significant difference in average goals scored between home and away clubs or between players using their left or right foot, indicating a balanced competitive environment. This information can guide clubs in strategic planning and player valuation, though improvements in valuation models are needed.