

South African Soccer Dataset – SQL

1. View the first 100 rows of the dataset to understand its structure.

The screenshot shows the SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'PHINDILE'. The 'Tables' node under 'Soccer_Team_Analysis' contains 'dbo.soccer_team_analysis'. The 'Results' tab in the center displays the output of the following query:

```
SELECT *
FROM soccer_team_analysis;
-- 1. View the first 100 rows of the dataset to understand its structure.
SELECT TOP 100 *
FROM soccer_team_analysis;
```

The results show 10 rows of player data:

player_name	team	date_of_birth	age	marital_status	number_of_kids	nationality	country_of_birth	position	preferred_foot
"Spanda Dlamini"	"Stellenbosch FC"	1995-05-28 00:00:00.000000	30	"Widowed"	0	"South African"	"South African"	"Defender"	"Right"
"Thabo Ndlonu"	"Cape Town City"	2004-06-01 00:00:00.000000	21	"Single"	1	"Zimbabwean"	"Zimbabwean"	"Forward"	"Right"
"Vusi Mofe"	"Stellenbosch FC"	2006-09-20 00:00:00.000000	19	"Single"	0	"Nigerian"	"Nigerian"	"Forward"	"Left"
"Thembu Mahlangu"	"Bloemfontein Celtic"	2004-02-08 00:00:00.000000	21	"Divorced"	0	"Zambian"	"Zambian"	"Goalkeeper"	"Left"
"Nokuthula Sthole"	"Polokwane City"	2003-03-18 00:00:00.000000	22	"Divorced"	3	"Nigerian"	"Nigerian"	"Goalkeeper"	"Both"
"Thembi Sthole"	"Kaizer Chiefs"	1989-02-14 00:00:00.000000	36	"Married"	2	"Nigerian"	"Nigerian"	"Goalkeeper"	"Left"
"Siyanda Mahlangu"	"Chippa United"	1989-12-09 00:00:00.000000	36	"Single"	0	"Zambian"	"Zambian"	"Defender"	"Both"
"Lerato Mashaba"	"Polokwane City"	1998-01-31 00:00:00.000000	27	"Widowed"	0	"Ghananian"	"Ghananian"	"Forward"	"Right"
"Nomza Mahlangu"	"Polokwane City"	1991-07-16 00:00:00.000000	34	"Divorced"	0	"Malawian"	"Malawian"	"Forward"	"Left"
"Tumelo Khumalo"	"Kaizer Chiefs"	1996-08-28 00:00:00.000000	29	"Married"	4	"Malawian"	"Malawian"	"Forward"	"Both"

At the bottom, a message indicates "Query executed successfully."

2. Count the total number of players in the dataset.

The screenshot shows the SQL Server Management Studio interface. The Object Explorer on the left shows the database structure for 'PHINDILE'. The 'Tables' node under 'Soccer_Team_Analysis' contains 'dbo.soccer_team_analysis'. The 'Results' tab in the center displays the output of the following query:

```
SELECT *
FROM soccer_team_analysis;
-- 1. View the first 100 rows of the dataset to understand its structure.
SELECT TOP 100 *
FROM soccer_team_analysis;
-- 2. Count the total number of players in the dataset.
SELECT COUNT(*) AS total_players
FROM soccer_team_analysis;
```

The results show 1 row of data:

total_players
300

At the bottom, a message indicates "Query executed successfully."

3. List all unique teams in the league.

```

SELECT *
FROM soccer_team_analysis;

-- 1. View the first 100 rows of the dataset to understand its structure.
SELECT TOP 100 *
FROM soccer_team_analysis;
-- 2. Count the total number of players in the dataset.
SELECT COUNT(*) AS total_players
FROM soccer_team_analysis;
-- 3. List all unique teams in the league.
SELECT DISTINCT team
FROM soccer_team_analysis
ORDER BY team;

```

team	Count
"AmaZulu FC"	27
"Bloemfontein Celtic"	23
"Cape Town City"	22
"Chippa United"	21
"Golden Arrows"	20
"Kaizer Chiefs"	20
"Mamelodi Sundowns"	19
"Moroka Swallows"	19
"Orlando Pirates"	18
"Polokwane City"	18
"Richards Bay FC"	16

Query executed successfully.

4. Count how many players are in each team.

```

-- 2. Count the total number of players in the dataset.
SELECT COUNT(*) AS total_players
FROM soccer_team_analysis;
-- 3. List all unique teams in the league.
SELECT DISTINCT team
FROM soccer_team_analysis
ORDER BY team;
-- 4. Count how many players are in each team.
SELECT team, COUNT(*) AS player_count
FROM soccer_team_analysis
GROUP BY team
ORDER BY player_count DESC;

```

team	player_count
"Polokwane City"	27
"Golden Arrows"	23
"AmaZulu FC"	22
"Chippa United"	21
"Stellenbosch FC"	20
"SuperSport United"	20
"TS Galaxy"	20
"Kaizer Chiefs"	20
"Mamelodi Sundowns"	19
"Moroka Swallows"	19
"Royal AM"	18

Query executed successfully.

5. Identify the top 10 players with the most goals.

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure for "PHINDILE (SQL Server 16.0.1000.6 - Phindle(Pretty))".
- SQL Query Editor:** Contains the following T-SQL code:


```

13 ORDER BY team;
-- 4. Count how many players are in each team.
14 SELECT team, COUNT(*) AS player_count
15 FROM soccer_team_analysis
16 GROUP BY team
17 ORDER BY player_count DESC;
-- 5. Identify the top 10 players with the most goals.
18 SELECT TOP 10 player_name, goals
19 FROM soccer_team_analysis
20 ORDER BY goals DESC;
21
22
23
24
25
      
```
- Results Grid:** Displays the top 10 players with the most goals, ordered by goals in descending order.

player_name	goals
"Vusi Molefe"	99
"Thabo Ndlonu"	98
"Gugu Hlongwane"	98
"Thabo Sithole"	98
"Manda Mabena"	98
"Mpho Mahlangu"	97
"Botumelo Nkosi"	96
"Mpho Radebe"	92
"Khanyi Nkosi"	92
"Sipho Phiri"	91

- Status Bar:** Shows "Query executed successfully." and other session details.

6. Find the average salary for players in each team.

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure for "PHINDILE (SQL Server 16.0.1000.6 - Phindle(Pretty))".
- SQL Query Editor:** Contains the following T-SQL code:


```

22 ORDER BY goals DESC;
-- 6. Find the average salary for players in each team.
23 SELECT team, AVG(average_salary_zar) AS avg_salary
24 FROM soccer_team_analysis
25 GROUP BY team
26 ORDER BY avg_salary DESC;
27
28
29
      
```
- Results Grid:** Displays the average salary for players in each team, ordered by average salary in descending order.

team	avg_salary
"Royal AM"	209078.331111111
"Cape Town City"	208407.432
"Golden Arrows"	195057.643913043
"SuperSport United"	195482.9785
"Mamelodi Sundowns"	194404.484736842
"Richards Bay FC"	193006.199230769
"Kaizer Chiefs"	188954.4615
"Moroka Swallows"	186786.827894737
"TS Galaxy"	184719.574
"Stellenbosch FC"	182326.7355
"Chippa United"	180947.634761905

- Status Bar:** Shows "Query executed successfully." and other session details.

7. Retrieve the top 10 players with the highest market value.

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure for "PHINDILE (SQL Server 16.0.1000.6 - Phindle(Pretty))".
- SQL Query1...Pretty (56)*:** The query window contains the following T-SQL code:


```

24 SELECT team, AVG(average_salary_zar) AS avg_salary
25 FROM soccer_team_analysis
26 GROUP BY team
27 ORDER BY avg_salary DESC;
-- 7. Retrieve the top 10 players with the highest market value.
28 SELECT TOP 10 player_name, market_value_zar
29 FROM soccer_team_analysis
30 ORDER BY market_value_zar DESC;
31
32
33
      
```
- Results:** The results table shows 10 rows of data:

	player_name	market_value_zar
1	"Ayanda Mabaso"	24979190.35
2	"Lebogang Mabena"	24887821.58
3	"Gugu Mathaba"	2486294.34
4	"Kagiso Mokoena"	24550192.81
5	"Botumelo Radebe"	24540830.44
6	"Thabo Ndlonvu"	24380712.61
7	"Sipho Mashaba"	24342131.59
8	"Lerato Ngobeni"	24231618.07
9	"Thabo Tshabalala"	24229791.64
10	"Lerato Tshabalala"	24226785.17
- Status Bar:** Shows "Query executed successfully." and other session details.

8. Calculate the average passing accuracy for each position.

The screenshot shows the SSMS interface with the following details:

- Object Explorer:** Shows the database structure for "PHINDILE (SQL Server 16.0.1000.6 - Phindle(Pretty))".
- SQL Query1...Pretty (56)*:** The query window contains the following T-SQL code:


```

24 SELECT team, AVG(average_salary_zar) AS avg_salary
25 FROM soccer_team_analysis
26 GROUP BY team
27 ORDER BY avg_salary DESC;
-- 7. Retrieve the top 10 players with the highest market value.
28 SELECT TOP 10 player_name, market_value_zar
29 FROM soccer_team_analysis
30 ORDER BY market_value_zar DESC;
-- 8. Calculate the average passing accuracy for each position.
31 SELECT position, AVG(passing_accuracy) AS avg_passing_accuracy
32 FROM soccer_team_analysis
33 GROUP BY position
34 ORDER BY avg_passing_accuracy DESC;
35
36
37
      
```
- Results:** The results table shows 4 rows of data:

	position	avg_passing_accuracy
1	"Goalkeeper"	83.9397058823529
2	"Forward"	83.210843373494
3	"Midfielder"	82.8177215189873
4	"Defender"	82.6642857142857
- Status Bar:** Shows "Query executed successfully." and other session details.

9. Compare shot accuracy with goals to find correlations.

The screenshot shows the SSMS interface with the Object Explorer on the left and a query results window on the right. The query results window displays a single row of data from the `soccer_team_analysis` table.

```
-- 9. Compare shot accuracy with goals to find correlations.
SELECT
    AVG(CAST(shot_accuracy AS FLOAT)) AS avg_shot_accuracy,
    AVG(CAST(goals AS FLOAT)) AS avg_goals,
    COUNT(*) AS total_players
FROM soccer_team_analysis;
```

	avg_shot_accuracy	avg_goals	total_players
1	65.159333333334	33.1466666666667	300

Query executed successfully.

10. Compute total goals and assists for each team

The screenshot shows the SSMS interface with the Object Explorer on the left and a query results window on the right. The query results window displays a list of 11 teams with their total goals and assists.

```
-- 9. Compare shot accuracy with goals to find correlations.
SELECT
    AVG(CAST(shot_accuracy AS FLOAT)) AS avg_shot_accuracy,
    AVG(CAST(goals AS FLOAT)) AS avg_goals,
    COUNT(*) AS total_players
FROM soccer_team_analysis;
-- 10. Compute total goals and assists for each team.
SELECT
    team,
    SUM(CAST(goals AS FLOAT)) AS total_goals,
    SUM(CAST(assists AS FLOAT)) AS total_assists
FROM soccer_team_analysis
GROUP BY team
ORDER BY total_goals DESC;
```

team	total_goals	total_assists
1 "Polokwane City"	1300	783
2 "Sekhukhune United"	842	608
3 "Mokoka Swallows"	729	452
4 "Stellenbosch FC"	725	514
5 "Cape Town City"	684	205
6 "Golden Arrows"	676	519
7 "Bleekfontein Celtic"	613	566
8 "KaiZer Chiefs"	597	481
9 "SuperSport United"	580	514
10 "Mamelodi Sundowns"	570	475
11 "AmaZulu FC"	546	568

Query executed successfully.

11. Count players by their marital status.

Soccer_Team_Analysis

```

46     SUM(CAST(goals AS FLOAT)) AS total_goals,
47     SUM(CAST(assists AS FLOAT)) AS total_assists
48   FROM soccer_team_analysis
49   GROUP BY team
50   ORDER BY total_goals DESC;
-- 11. Count players by their marital status.
51   SELECT marital_status, COUNT(*) AS count_players
52   FROM soccer_team_analysis
53   GROUP BY marital_status
54   ORDER BY count_players DESC;
55
56
57
58

```

No issues found

marital_status	count_players
"Single"	79
"Widowed"	78
"Divorced"	78
"Married"	65

Query executed successfully.

12. Count players by nationality.

Soccer_Team_Analysis

```

47     SUM(CAST(assists AS FLOAT)) AS total_assists
48   FROM soccer_team_analysis
49   GROUP BY team
50   ORDER BY total_goals DESC;
-- 11. Count players by their marital status.
51   SELECT marital_status, COUNT(*) AS count_players
52   FROM soccer_team_analysis
53   GROUP BY marital_status
54   ORDER BY count_players DESC;
-- 12. Count players by nationality.
55   SELECT nationality, COUNT(*) AS count_players
56   FROM soccer_team_analysis
57   GROUP BY nationality
58   ORDER BY count_players DESC;
59
60
61

```

No issues found

nationality	count_players
"Ghananian"	47
"South African"	46
"Zimbabwean"	45
"Zambian"	44
"Malawian"	42
"Nigerian"	39
"Mozambican"	37

Query executed successfully.

13. Find average market value grouped by nationality

```

-- 12. Count players by nationality.
SELECT nationality, COUNT(*) AS count_players
FROM soccer_team_analysis
GROUP BY nationality
ORDER BY count_players DESC;
-- 13. Find average market value grouped by nationality.
SELECT nationality, AVG(market_value_zar) AS avg_market_value_zar
FROM soccer_team_analysis
GROUP BY nationality
ORDER BY avg_market_value_zar DESC;

```

14. Determine how many player contracts end in each year.

```

-- 13. Find average market value grouped by nationality.
SELECT nationality, AVG(market_value_zar) AS avg_market_value_zar
FROM soccer_team_analysis
GROUP BY nationality
ORDER BY avg_market_value_zar DESC;
-- 14. Determine how many player contracts end in each year.
SELECT YEAR(contract_end_year) AS contract_end_year,
       COUNT(*) AS players_count
FROM soccer_team_analysis
GROUP BY YEAR(contract_end_year)
ORDER BY contract_end_year;

```

Query executed successfully.

15. Identify players whose contracts end next year.

```

65 ORDER BY avg.market_value_zar DESC;
66 -- 14. Determine how many player contracts end in each year.
67 SELECT YEAR(contract_end_year) AS contract_end_year,
68     COUNT(*) AS players_count
69 FROM soccer_team_analysis
70 GROUP BY YEAR(contract_end_year)
71 ORDER BY contract_end_year;
72 -- 15. Identify players whose contracts end next year.
73 SELECT player_name, team, contract_end_year
74 FROM soccer_team_analysis
75 WHERE YEAR(contract_end_year) = 2026;
76
77
78
79

```

No issues found

Results Messages

player_name	team	contract_end_year

Query executed successfully.

16. Summarize the number of players by injury status.

```

67 SELECT YEAR(contract_end_year) AS contract_end_year,
68     COUNT(*) AS players_count
69 FROM soccer_team_analysis
70 GROUP BY YEAR(contract_end_year)
71 ORDER BY contract_end_year;
72 -- 15. Identify players whose contracts end next year.
73 SELECT player_name, team, contract_end_year
74 FROM soccer_team_analysis
75 WHERE YEAR(contract_end_year) = 2026;
76 --16. Summarize the number of players by injury status.
77 SELECT injury_status, COUNT(*) AS count_players
78 FROM soccer_team_analysis
79 GROUP BY injury_status
80 ORDER BY count_players DESC;
81

```

No issues found

Results Messages

injury_status	count_players
"Recovering"	104
"Healthy"	99
"Injured"	97

Query executed successfully.

17. Calculate goals per match ratio for each player.

The screenshot shows the SSMS interface with the Object Explorer on the left and a query results window on the right. The query results window displays a table of player statistics.

```

--16. Summarize the number of players by injury status.
SELECT injury_status, COUNT(*) AS count_players
FROM soccer_team_analysis
GROUP BY injury_status
ORDER BY count_players DESC;

-- 17. Calculate goals per match ratio for each player.
SELECT
    player_name,
    goals,
    matches_played,
    CAST(goals AS FLOAT) / NULLIF(matches_played, 0) AS goals_per_match
FROM soccer_team_analysis
ORDER BY goals_per_match DESC;

```

	player_name	goals	matches_played	goals_per_match
1	"Thembi Zulu"	72	3	24
2	"Thabo Mthembu"	88	5	17.6
3	"Gugu Tahabala"	57	9	6.33333333333333
4	"Lindive Mabaso"	67	13	5.15384615384615
5	"Mpho Zulu"	5	1	5
6	"Sipho Tshabalala"	110	26	4.23076923076923
7	"Bontu Nkosi"	96	25	3.84
8	"Kerbo Tshabalala"	106	30	3.53333333333333
9	"Lindive Zulu"	34	11	3.09090909090909
10	"Sipho Phiri"	91	33	2.75757575757576
11	"Sipho Dlamini"	43	22	1.95454545454545

Query executed successfully.

18. Count how many players are managed by each agent.

The screenshot shows the SSMS interface with the Object Explorer on the left and a query results window on the right. The query results window displays a table of agent statistics.

```

--18. Count how many players are managed by each agent.
SELECT agent, COUNT(*) AS player_count
FROM soccer_team_analysis
GROUP BY agent
ORDER BY player_count DESC;

```

agent	player_count
"PlayerFirst"	63
"Pro Sport"	62
"nan"	62
"SoccerLink Africa"	62
"SA Elite Agents"	51

Query executed successfully.

19. Calculate average height and weight by player position.

Soccer_Team_Analysis

```

88 ORDER BY goals_per_match DESC;
89 -- 18. Count how many players are managed by each agent.
90 SELECT agent, COUNT(*) AS player_count
91 FROM soccer_team_analysis
92 GROUP BY agent
93 ORDER BY player_count DESC;
94 -- 19. Calculate average height and weight by player position.
95 SELECT
96     position,
97     AVG(height_cm) AS avg_height_cm,
98     AVG(weight_kg) AS avg_weight_kg
99 FROM soccer_team_analysis
100 GROUP BY position
101 ORDER BY avg_height_cm DESC;
102

```

No issues found

position	avg_height_cm	avg_weight_kg
Defender	182	78
Forward	179	78
Gaolekeeper	179	78
Midfielder	179	76

Ln: 95, Ch: 1 (175 chars, 7 lines) SPC CRLF Windows 1252

Query executed successfully.

20. Identify players with the highest combined goals and assists.

Soccer_Team_Analysis

```

96 position,
97     AVG(height_cm) AS avg_height_cm,
98     AVG(weight_kg) AS avg_weight_kg
99 FROM soccer_team_analysis
100 GROUP BY position
101 ORDER BY avg.height_cm DESC;
102 -- 20. Identify players with the highest combined goals and assists.
103 SELECT
104     player_name,
105     goals,
106     assists,
107     (goals + assists) AS total_contribution
108 FROM soccer_team_analysis
109 ORDER BY total_contribution DESC;
110

```

No issues found

player_name	goals	assists	total_contribution
Vusi Molefe	99	77	9977
Gugu Hlongwane	98	38	9838
Manda Mabena	98	37	9837
Thabo Ndlovu	98	28	9828
Thabo Sithole	98	2	982
Khanyi Ndlovu	9	75	975
Mpho Mahlangu	97	4	974
Tumelo Dlamini	9	7	97
Manda Sithole	9	7	97
Bontlelo Nkosi	96	54	9654
Nomas Mahlangu	9	6	96

Ln: 102, Ch: 1 (228 chars, 8 lines) SPC CRLF Windows 1252

Query executed successfully.