

1. La cantidad de juegos jugados en cada temporada por cada equipo, de cada liga (tome en cuenta que cada equipo puede jugar como visitante o como anfitrión).

SELECT

season,

teamID,

COUNT(gameID) AS juegos_jugados

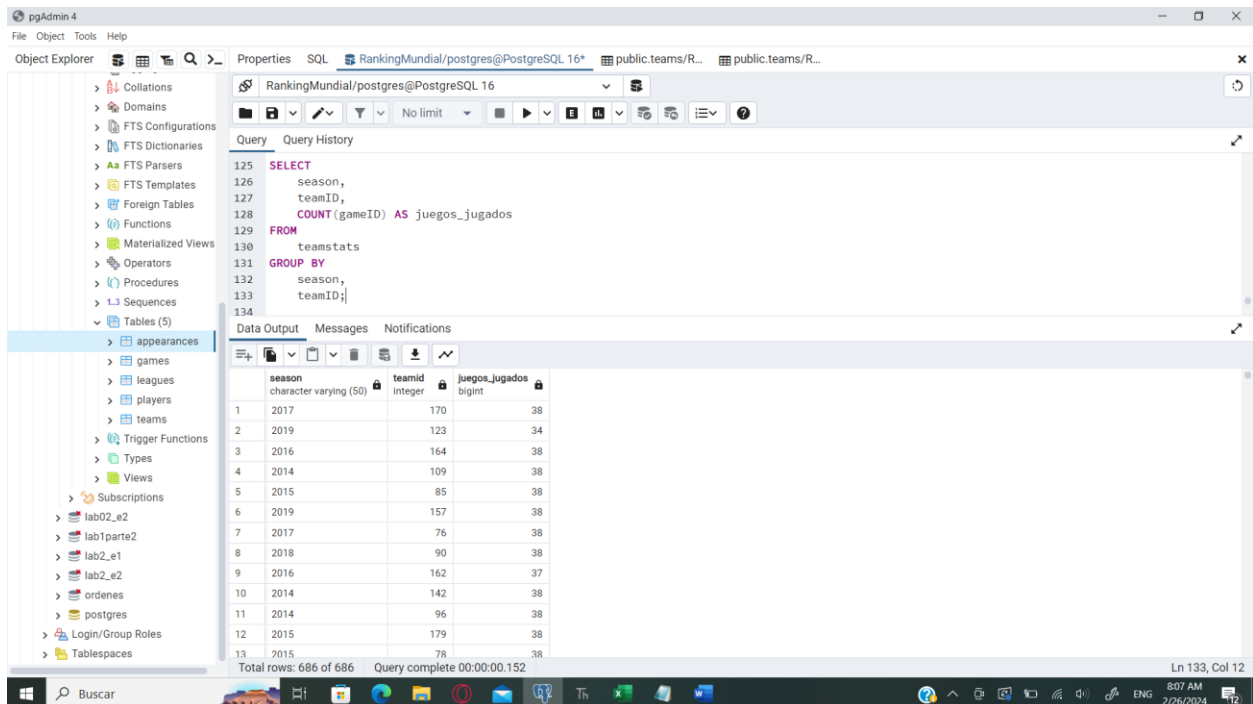
FROM

teamstats

GROUP BY

season,

teamID;



The screenshot shows the pgAdmin 4 interface. The left pane displays the database structure, including tables like 'appearances', 'games', 'leagues', 'players', 'teams', 'types', 'views', 'subscriptions', 'lab02_e2', 'lab1parte2', 'lab2_e1', 'lab2_e2', 'ordenes', 'postgres', 'Login/Group Roles', and 'Tablespaces'. The main pane shows a SQL query being executed in the 'RankingMundial/postgres@PostgreSQL 16*' database. The query is:

```
SELECT
season,
teamID,
COUNT(gameID) AS juegos_jugados
FROM
teamstats
GROUP BY
season,
teamID;
```

The 'Data Output' pane shows the results of the query, which are 686 rows. The first 13 rows are displayed:

season	teamid	juegos_jugados
2017	170	38
2019	123	34
2016	164	38
2014	109	38
2015	85	38
2019	157	38
2017	76	38
2018	90	38
2016	162	37
2014	142	38
2014	96	38
2015	179	38
2015	78	38

The status bar at the bottom indicates 'Total rows: 686 of 686' and 'Query complete 00:00:00.152'.

2. ¿Quién es el mejor equipo de todas las ligas y de todas las temporadas según las estadísticas de diferencia de goles?

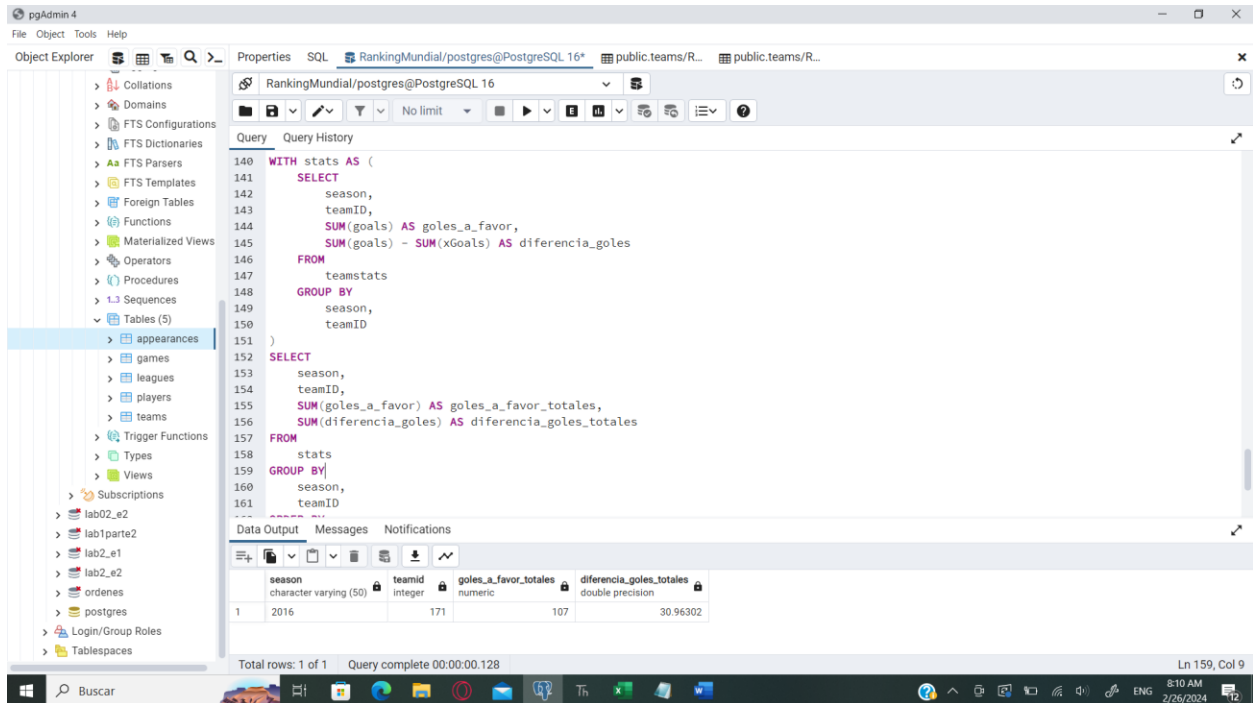
WITH stats AS (

SELECT

season,

teamID,

```
SUM(goals) AS goles_a_favor,
SUM(goals) - SUM(xGoals) AS diferencia_goles
FROM
    teamstats
GROUP BY
    season,
    teamID
)
SELECT
    season,
    teamID,
    SUM(goles_a_favor) AS goles_a_favor_totales,
    SUM(diferencia_goles) AS diferencia_goles_totales
FROM
    stats
GROUP BY
    season,
    teamID
ORDER BY
    diferencia_goles_totales DESC
LIMIT 1;
```



3. ¿Quiénes son los jugadores que han realizado mayor cantidad de goles a través de todas las temporadas? ¿Cuáles son los jugadores con mayor cantidad de pases izquierdos y pases derechos que han hecho goles? (Compare contra los resultados del inciso 2 y determine de manera textual si dichos jugadores pertenecen a los equipos del inciso anterior).

SELECT

shooterID,

COUNT(*) AS cantidad_goles

FROM

shots

WHERE

shotResult = 'Goal'

GROUP BY

shooterID

ORDER BY

cantidad_goles DESC

LIMIT 10;

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer showing the database structure. The main pane displays a SQL query and its results. The query is as follows:

```

SELECT
  shooterID,
  COUNT(*) AS cantidad_goles
FROM
  shots
WHERE
  shotResult = 'Goal'
GROUP BY
  shooterID
ORDER BY
  cantidad_goles DESC
LIMIT 10;

```

The results are shown in a table with the following data:

shooterid	cantidad_goles
1	2097
2	2371
3	227
4	2098
5	647
6	318
7	619
8	2270
9	1209
10	3294

4. Realice un comparativo de las probabilidades de todas las casas de apuesta por temporada, liga y equipo, eliminando aquellos equipos que no tienen estadísticas en ninguna casa de apuesta.

WITH pasos AS (

SELECT

shooterID,

COUNT(CASE WHEN lastAction = 'LeftFoot' THEN 1 END) AS pases_izquierdos,

COUNT(CASE WHEN lastAction = 'RightFoot' THEN 1 END) AS pases_derechos

FROM

shots

WHERE

shotResult = 'Goal'

GROUP BY

shooterID

)

SELECT

p.shooterID,

p.pases_izquierdos,

```
    p.pases_derechos,  
    s.goals  
FROM  
    pasos p  
INNER JOIN  
    (  
    SELECT  
        shooterID,  
        COUNT(*) AS goals  
    FROM  
        shots  
    WHERE  
        shotResult = 'Goal'  
    GROUP BY  
        shooterID  
    ) s ON p.shooterID = s.shooterID  
ORDER BY  
    (p.pases_izquierdos + p.pases_derechos) DESC  
LIMIT 10;
```

pgAdmin 4

Object Explorer

Properties SQL RankingMundial/postgres@PostgreSQL 16*

Query

```

140
141 WITH pasos AS (
142     SELECT
143         shooterID,
144         COUNT(CASE WHEN lastAction = 'LeftFoot' THEN 1 END) AS pases_izquierdos,
145         COUNT(CASE WHEN lastAction = 'RightFoot' THEN 1 END) AS pases_derechos
146     FROM
147         shots
148     WHERE
149         shotResult = 'Goal'
150     GROUP BY
151         shooterID
152 )

```

Data Output Messages Notifications

	shooterid integer	pases_izquierdos bigint	pases_derechos bigint	goals bigint
1	1489	0	0	35
2	4790	0	0	1
3	273	0	0	18
4	3936	0	0	2
5	5729	0	0	1
6	2466	0	0	15
7	5697	0	0	1
8	2196	0	0	9
9	1750	0	0	1
10	6114	0	0	2

Total rows: 10 of 10 Query complete 00:00:00.209 Ln 148, Col 10

5. ¿Cuál es el mejor equipo de todas las ligas y de todas las temporadas según las apuestas?

WITH probabilidades AS (

SELECT

gameID,

leagueID,

season,

homeTeamID,

awayTeamID,

CASE WHEN B365H != 0 THEN 1.0 / B365H ELSE NULL END AS prob_local,

CASE WHEN B365D != 0 THEN 1.0 / B365D ELSE NULL END AS prob_empate,

CASE WHEN B365A != 0 THEN 1.0 / B365A ELSE NULL END AS prob_visitante

FROM

games

)

SELECT

season,

leagueID,

```
CASE
    WHEN homeTeamID IS NOT NULL THEN homeTeamID
    ELSE awayTeamID
END AS teamID,
MAX(prob_local) AS prob_local_max,
MAX(prob_empate) AS prob_empate_max,
MAX(prob_visitante) AS prob_visitante_max
FROM
    probabilidades
GROUP BY
    season,
    leagueID,
    homeTeamID,
    awayTeamID
ORDER BY
    season,
    leagueID,
    teamID;
```

Query

```

WITH probabilidades AS (
SELECT
    gameID,
    leagueID,
    season,
    homeTeamID,
    awayTeamID,
    CASE WHEN B365H != 0 THEN 1.0 / B365H ELSE NULL END AS prob_local,
    CASE WHEN B365D != 0 THEN 1.0 / B365D ELSE NULL END AS prob_empate,
    CASE WHEN B365A != 0 THEN 1.0 / B365A ELSE NULL END AS prob_visitante
FROM
    games
)

```

Data Output

season	leagueid	teamid	prob_local_max	prob_empate_max	prob_visitante_max
1	2014	71	0.347222222222222	0.3225806451612903	0.347222222222222
2	2014	71	0.416666666666667	0.29411764705882354	0.3125
3	2014	71	0.5235602094240838	0.27027027027027023	0.23094688221709006
4	2014	71	0.16	0.25	0.6172839506172839
5	2014	71	0.21052631578947367	0.27777777777777778	0.5405405405405405
6	2014	71	0.4	0.29411764705882354	0.3333333333333333
7	2014	71	0.19047619047619047	0.26666666666666666	0.5714285714285714
8	2014	71	0.4166666666666667	0.30303030303030304	0.30303030303030304
9	2014	71	0.4	0.3076923076923077	0.35714285714285715
10	2014	71	0.425531914893617	0.30303030303030304	0.29411764705882354

Total rows: 1000 of 12680 Query complete 00:00:00.156 Ln 136, Col 51

5. ¿Cuál es el mejor equipo de todas las ligas y de todas las temporadas según las apuestas?

WITH probabilidades AS (

SELECT

gameID,

leagueID,

season,

homeTeamID,

awayTeamID,

CASE WHEN B365H != 0 THEN 1.0 / B365H ELSE NULL END AS prob_local,

CASE WHEN B365D != 0 THEN 1.0 / B365D ELSE NULL END AS prob_empate,

CASE WHEN B365A != 0 THEN 1.0 / B365A ELSE NULL END AS prob_visitante

FROM

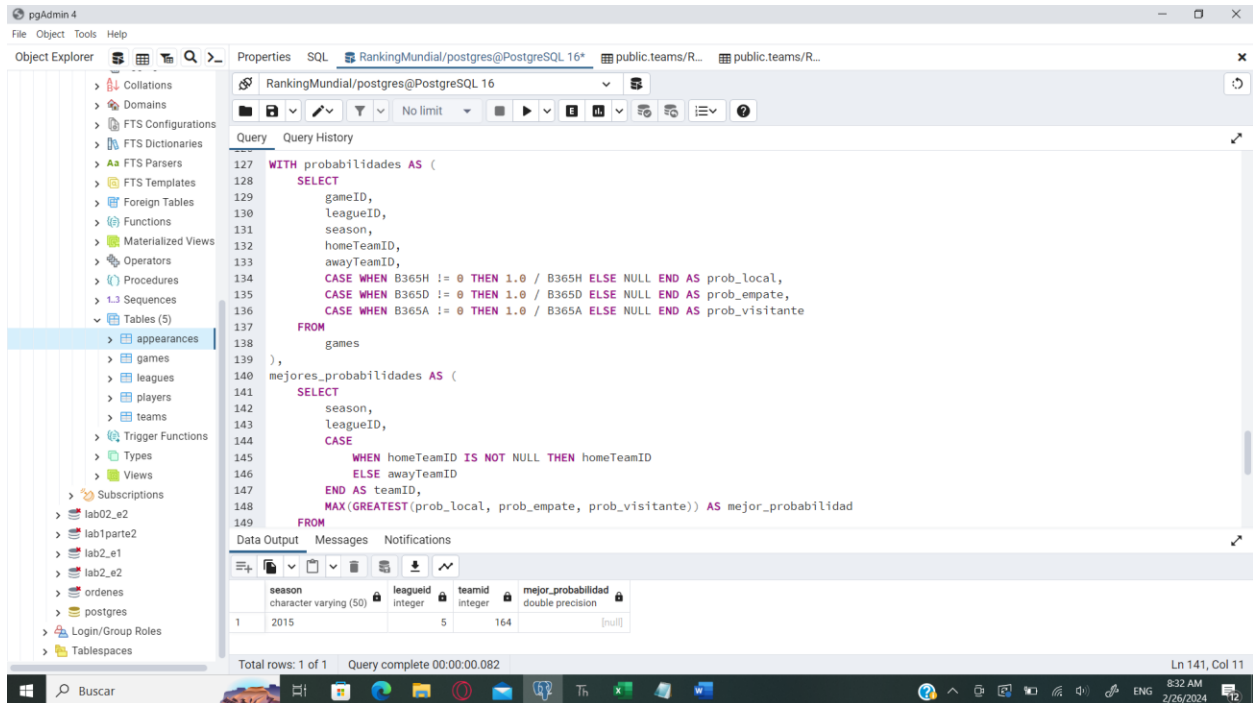
games

),

mejores_probabilidades AS (

SELECT


```
    season,
    leagueID,
CASE
    WHEN homeTeamID IS NOT NULL THEN homeTeamID
    ELSE awayTeamID
END AS teamID,
MAX(GREATEST(prob_local, prob_empate, prob_visitante)) AS mejor_probabilidad
FROM
    probabilidades
GROUP BY
    season,
    leagueID,
    homeTeamID,
    awayTeamID
)
SELECT
    season,
    leagueID,
    teamID,
    mejor_probabilidad
FROM
    mejores_probabilidades
ORDER BY
    mejor_probabilidad DESC
LIMIT 1;
```



6. ¿Quiénes son los jugadores de cada liga y cada temporada que tienen los mejores atributos – características de juego -pases, goles, etc.? ¿De acuerdo a este inciso, y comparándolo con el inciso 2 y 5 anteriores, alguno de los jugadores más valiosos se encuentra dentro del mejor equipo?

```

SELECT l.name AS league_name, g.season, p.name AS player_name,
       a.goals, a.assists, a.keyPasses
FROM appearances a
JOIN players p ON a.playerID = p.playerID
JOIN games g ON a.gameID = g.gameID
JOIN leagues l ON g.leagueID = l.leagueID
WHERE (a.goals > 0 OR a.assists > 0 OR a.keyPasses > 0)
ORDER BY league_name, g.season, (a.goals + a.assists + a.keyPasses) DESC;

```

pgAdmin 4

Object Explorer

RankingMundial/postgres@PostgreSQL 16*

public.teams/R... public.teams/R... public.leagues...

RankingMundial/postgres@PostgreSQL 16

Query Query History

```

123 ,
124
125
126
127
128
129
130 SELECT l.name AS league_name, g.season, p.name AS player_name,
131        a.goals, a.assists, a.keyPasses
132 FROM appearances a
133 JOIN players p ON a.playerID = p.playerID
134 JOIN games g ON a.gameID = g.gameID
135 JOIN leagues l ON g.leagueID = l.leagueID

```

Data Output Messages Notifications

	league_name	season	player_name	goals	assists	keypasses
1	Bundesliga	2014	Gonzalo Castro	2	1	10
2	Bundesliga	2014	Max Kruse	1	0	10
3	Bundesliga	2014	Kevin De Bruyne	0	2	6
4	Bundesliga	2014	Tranquillo Barnetta	0	3	5
5	Bundesliga	2014	Tranquillo Barnetta	0	1	7
6	Bundesliga	2014	Moritz Stoppelkamp	1	1	6
7	Bundesliga	2014	Mario Vrancic	0	0	8
8	Bundesliga	2014	Marco Reus	1	1	6
9	Bundesliga	2014	Arjen Robben	2	2	4
10	Bundesliga	2014	Henrikh Mkhitaryan	1	0	7
11	Bundesliga	2014	Hiroshi Kiyotake	1	1	6

Total rows: 1000 of 153840 Query complete 00:00:00.614 Ln 138, Col 1

7. Obtenga el rendimiento de los equipos en promedio, comparando goles metidos contra la expectativa de goles, determinando qué equipo era quien tenía más expectativa de goles contra quien fue en realidad el que acertó más goles (goals vs expected goals, xgoals) en general, pero también es necesario que lo muestre si dichos equipos jugaron como locales o como extranjeros.

WITH rendimiento_equipos AS (

SELECT

ts.teamID,

ts.season,

AVG(ts.goals) AS promedio_goles,

AVG(ts.xGoals) AS promedio_xGoals

FROM

teamstats ts

GROUP BY

ts.teamID,

ts.season

)

SELECT

re.teamID,

```

re.season,

re.promedio_goles,

re.promedio_xGoals,

t.name AS nombre_equipo

FROM

rendimiento_equipos re

INNER JOIN

teams t ON re.teamID = t.teamID;

```

The screenshot shows the pgAdmin 4 interface. The SQL query editor contains the following query:

```

WITH rendimiento_equipos AS (
    SELECT
        ts.teamID,
        ts.season,
        AVG(ts.goals) AS promedio_goles,
        AVG(ts.xGoals) AS promedio_xGoals
    FROM
        teamstats ts
    GROUP BY
        ts.teamID,
        ts.season
)
SELECT

```

The Data Output tab shows the following table:

teamid	season	promedio_goles	promedio_xgoals	nombre_equipo
1	72	1.4210526315789474	1.417529842105263	Everton
2	150	1.6578947368421053	1.806704552631579	Real Madrid
3	178	1.5000000000000000	1.2944184285714282	Lyon
4	129	2.4117647058823529	2.4532422941176475	Borussia Dortmund
5	168	1.0000000000000000	0.9933667500000001	Nantes
6	154	1.5000000000000000	1.4649854210526316	Villarreal
7	123	1.2352941176470588	1.1099452058823531	Werder Bremen
8	83	2.0263157894736842	1.67319097368421	Arsenal
9	166	1.5789473684210526	1.3754441315789476	Montpellier
10	84	1.1052631578947368	1.0515432105263158	Swansea

Total rows: 686 of 686 Query complete 00:00:00.069 Ln 133, Col 9

8. ¿Cuáles son las características/atributos de los equipos que han sido los líderes de sus ligas en las distintas temporadas? ¿Sus comportamientos son similares?

```

WITH LeagueLeaders AS (
    SELECT ts.season, ts.teamID, t.name AS team_name,
        AVG(ts.goals) AS avg_goals, AVG(ts.shots) AS avg_shots,
        AVG(ts.shotsOnTarget) AS avg_shots_on_target,
        AVG(ts.ppda) AS avg_ppda, AVG(ts.fouls) AS avg_fouls,
        AVG(ts.corners) AS avg_corners,
        AVG(ts.yellowCards) AS avg_yellow_cards,

```

```

        AVG(ts.redCards) AS avg_red_cards
FROM teamstats ts
JOIN teams t ON ts.teamID = t.teamID
WHERE ts.result = 'W' -- Considerando solo los partidos que el equipo ganó
GROUP BY ts.season, ts.teamID, t.name
)
SELECT ll.season, ll.team_name, ll.avg_goals, ll.avg_shots,
       ll.avg_shots_on_target, ll.avg_ppda, ll.avg_fouls,
       ll.avg_corners, ll.avg_yellow_cards, ll.avg_red_cards
FROM LeagueLeaders ll
JOIN (
    SELECT season, MAX(avg_goals) AS max_avg_goals,
           MAX(avg_shots) AS max_avg_shots,
           MAX(avg_shots_on_target) AS max_avg_shots_on_target,
           MAX(avg_ppda) AS max_avg_ppda,
           MAX(avg_fouls) AS max_avg_fouls,
           MAX(avg_corners) AS max_avg_corners,
           MAX(avg_yellow_cards) AS max_avg_yellow_cards,
           MAX(avg_red_cards) AS max_avg_red_cards
    FROM LeagueLeaders
    GROUP BY season
) AS MaxValues ON ll.season = MaxValues.season
AND (ll.avg_goals = MaxValues.max_avg_goals
     OR ll.avg_shots = MaxValues.max_avg_shots
     OR ll.avg_shots_on_target = MaxValues.max_avg_shots_on_target
     OR ll.avg_ppda = MaxValues.max_avg_ppda
     OR ll.avg_fouls = MaxValues.max_avg_fouls
     OR ll.avg_corners = MaxValues.max_avg_corners
     OR ll.avg_yellow_cards = MaxValues.max_avg_yellow_cards

```

OR II.avg_red_cards = MaxValues.max_avg_red_cards);

Query:

```

WITH LeagueLeaders AS (
SELECT ts.season, ts.teamID, t.name AS team_name,
      AVG(ts.goals) AS avg_goals, AVG(ts.shots) AS avg_shots,
      AVG(ts.shotsOnTarget) AS avg_shots_on_target,
      AVG(ts.ppda) AS avg_ppda, AVG(ts.fouls) AS avg_fouls,
      AVG(ts.corners) AS avg_corners,
      AVG(ts.yellowCards) AS avg_yellow_cards,
      AVG(ts.redCards) AS avg_red_cards
FROM teamstats ts
JOIN teams t ON ts.teamID = t.teamID
WHERE ts.result = 'W' -- Considerando solo los partidos que el equipo ganó
GROUP BY ts.season, ts.teamID, t.name

```

season	team_name	avg_goals	avg_shots	avg_shots_on_target	avg_ppda	avg_fouls	avg_corners	avg_nur
1	2017	Swansea	2.0000000000000000	9.7500000000000000	3.8750000000000000	23.5221375	8.3750000000000000	4.5000000000000000
2	2019	RasenBallsport Leipzig	3.5555555555555556	17.277777777777778	7.666666666666667	9.796866666666667	10.444444444444444	6.277777777777778
3	2017	Levante	2.2727272727272727	11.363636363636363	4.7272727272727273	12.497754545454542	14.090909090909091	5.3636363636363636
4	2014	Borussia Dortmund	2.6153846153846154	19.2307692307692308	7.7692307692307692	8.373053846153844	14.2307692307692308	5.9230769230769231
5	2017	Real Madrid	3.5454545454545455	17.454545454545455	8.090909090909091	9.760804545454548	11.7272727272727273	6.3636363636363636
6	2015	Atalanta	2.0000000000000000	13.000000000000000	5.0000000000000000	7.388445454545455	16.636363636363636	5.9090909090909091
7	2018	Toulouse	1.3750000000000000	10.625000000000000	4.1250000000000000	10.158825000000002	15.500000000000000	4.0000000000000000
8	2019	SPAL 2013	1.6000000000000000	13.800000000000000	4.4000000000000000	11.991380000000001	14.600000000000000	5.0000000000000000
9	2019	Bayern Munich	3.5000000000000000	18.500000000000000	7.8846153846153846	7.950626923076921	9.500000000000000	7.0000000000000000
10	2015	Granada	2.6000000000000000	11.500000000000000	5.0000000000000000	9.136730000000002	16.600000000000000	3.7000000000000000

9. ¿Según la casa de apuesta Beat365 (tome la mejor probabilidad de las 3 medidas), cuales deberían de ser los equipos que tenían la mayor probabilidad de ganar en cada una de las temporadas (seasons)?

SELECT g.season, t.name AS team_name, g.B365H AS probability_to_win

FROM games g

JOIN teams t ON g.homeTeamID = t.teamID

WHERE g.B365H = (SELECT MAX(B365H) FROM games WHERE season = g.season)

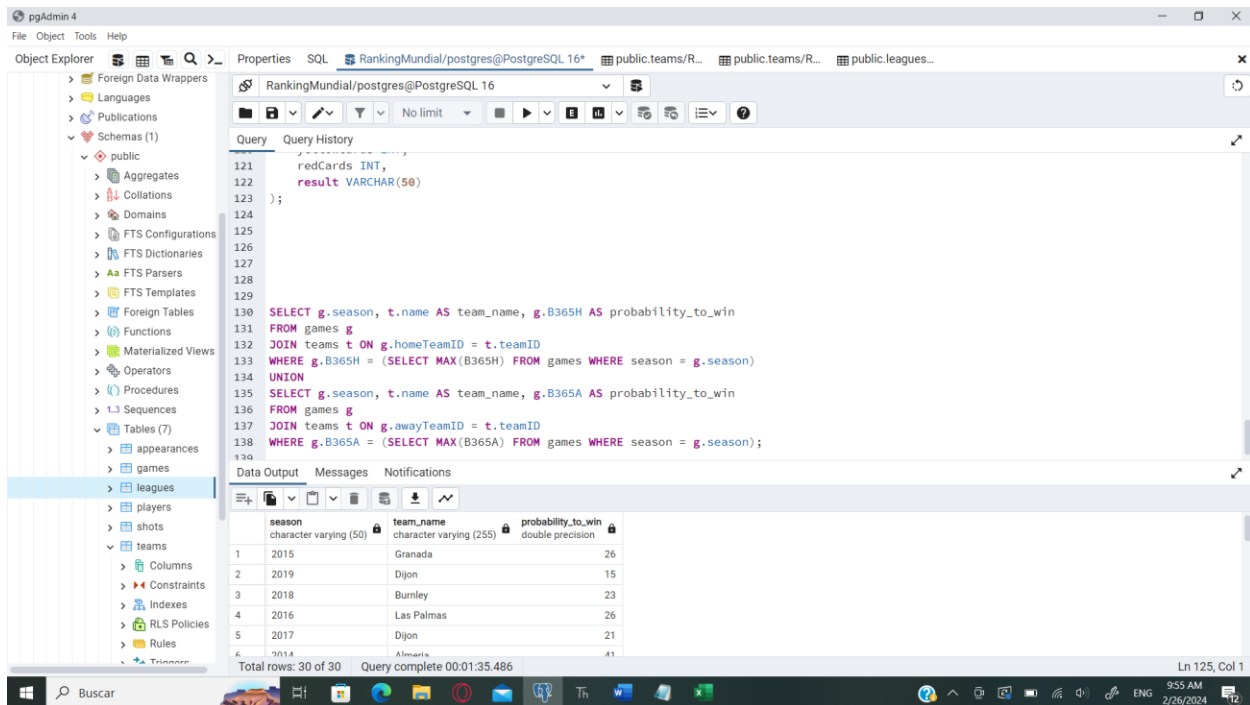
UNION

SELECT g.season, t.name AS team_name, g.B365A AS probability_to_win

FROM games g

JOIN teams t ON g.awayTeamID = t.teamID

WHERE g.B365A = (SELECT MAX(B365A) FROM games WHERE season = g.season);



10. Obtenga el top 10 de estadísticas de los equipos más limpios en jugar (mejor faltas, menos tarjetas amarillas, menos tarjetas rojas) y también el top 10 de los equipos más sucios.

```

SELECT ts.season, t.name AS team_name,
       AVG(ts.fouls) AS avg_fouls,
       AVG(ts.yellowCards) AS avg_yellow_cards,
       AVG(ts.redCards) AS avg_red_cards
FROM teamstats ts
JOIN teams t ON ts.teamID = t.teamID
GROUP BY ts.season, t.name
ORDER BY avg_fouls, avg_yellow_cards, avg_red_cards
LIMIT 10;

```

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer with the 'public' schema expanded. The main pane displays a SQL query in the 'Query' tab. Below the query editor is the 'Data Output' tab, which shows the results of the query in a table format. The table has 5 columns: 'season', 'team_name', 'avg_fouls', 'avg_yellow_cards', and 'avg_red_cards'. The results are sorted by 'avg_fouls' in descending order, limited to 10 rows.

season	team_name	avg_fouls	avg_yellow_cards	avg_red_cards
2018	Bayern Munich	8.1764705882352941	1.2941176470588235	0.058823529411764705882
2018	Borussia Dortmund	8.2058823529411765	1.1764705882352941	0.08823529411764705882
2018	Liverpool	8.2894736842105263	0.97368421052631578947	0.05263157894736842105
2015	Everton	8.2894736842105263	1.1578947368421053	0.13157894736842105263
2018	Manchester City	8.6315789473684211	1.1578947368421053	0.02631578947368421053
2019	Borussia Dortmund	8.6764705882352941	1.4411764705882353	0.02941176470588235294
2019	Liverpool	8.7105263157894737	1.0000000000000000	0.02631578947368421053
2018	West Ham	8.7105263157894737	1.5526315789473684	0.02631578947368421053
2018	Chelsea	8.9210526315789474	1.2894736842105263	0.0000000000000000
2017	Bournemouth	8.9210526315789474	1.4473684210526316	0.02631578947368421053

```

SELECT ts.season, t.name AS team_name,
       AVG(ts.fouls) AS avg_fouls,
       AVG(ts.yellowCards) AS avg_yellow_cards,
       AVG(ts.redCards) AS avg_red_cards
FROM teamstats ts
JOIN teams t ON ts.teamID = t.teamID
GROUP BY ts.season, t.name
ORDER BY avg_fouls DESC, avg_yellow_cards DESC, avg_red_cards DESC
LIMIT 10;

```


pgAdmin 4

File Object Tools Help

Object Explorer

- Foreign Data Wrappers
- Languages
- Publications
- Schemas (1)
 - public
 - Aggregates
 - Collations
 - Domains
 - FTS Configurations
 - FTS Dictionaries
 - FTS Parsers
 - FTS Templates
 - Foreign Tables
 - Functions
 - Materialized Views
 - Operators
 - Procedures
 - Sequences
 - Tables (7)
 - appearances
 - games
 - leagues
 - players
 - shots
 - teams
 - Columns
 - Constraints
 - Indexes
 - RLS Policies
 - Rules
 - Triggers

Properties SQL RankingMundial/postgres@PostgreSQL 16* public.teams/R... public.teams/R... public.leagues...

RankingMundial/postgres@PostgreSQL 16

Query Query History

```
127 SELECT ts.season, t.name AS team_name,
128        AVG(ts.fouls) AS avg_fouls,
129        AVG(ts.yellowCards) AS avg_yellow_cards,
130        AVG(ts.redCards) AS avg_red_cards
131 FROM teamstats ts
132 JOIN teams t ON ts.teamID = t.teamID
133 GROUP BY ts.season, t.name
134 ORDER BY avg_fouls DESC, avg_yellow_cards DESC, avg_red_cards DESC
135 LIMIT 10;
```

Data Output Messages Notifications

season	team_name	avg_fouls	avg_yellow_cards	avg_red_cards
character varying (50)	character varying (255)	numeric	numeric	numeric
1 2019	Getafe	18.5526315789473684	3.5789473684210526	0.10526315789473684211
2 2014	Hamburger SV	18.1764705882352941	2.4411764705882353	0.14705882352941176471
3 2014	Bayer Leverkusen	17.9117647058823529	2.2352941176470588	0.11764705882352941176
4 2014	Lazio	17.8421052631578947	2.7368421052631579	0.23684210526315789474
5 2017	Getafe	17.7368421052631579	3.4736842105263158	0.13157894736842105263
6 2014	Hoffenheim	17.6470588235294118	1.9411764705882353	0.05882352941176470588
7 2014	Parma Calcio 1913	17.3421052631578947	2.5789473684210526	0.21052631578947368421
8 2014	Hertha Berlin	17.3235294117647059	2.1176470588235294	0.11764705882352941176
9 2014	Werder Bremen	17.2941176470588235	1.9705882352941176	0.05882352941176470588
10 2015	Torino	17.2631578947368421	2.5000000000000000	0.07894736842105263158

Total rows: 10 of 10 Query complete 00:00:00.077 Ln 124, Col 1

9:57 AM 2/26/2024