/\*

About the Data

The first CSV file is for ball-by-ball data and it has information of all the 193468 balls bowled between the years 2008 and 2020. It has 17 columns and below is the details of those 17

columns.

The second file contains match-wise data and has data of 816 IPL matches. This table has 17 columns and below is a short description of the columns in this table.

\*/

1. Create a table named ‘matches’ with appropriate data types for columns

Create Table matches ( "id" INTEGER PRIMARY KEY NOT NULL,

"city" varchar(20) NOT NUll,

"date" varchar(20),

"player\_of\_match" varchar(50),

"venue" text,

"neutral\_venue" INTEGER,

"team1" varchar(150),

"team2" varchar(150),

"toss\_winner" varchar(150),

"toss\_decision" varchar(15),

"winner" varchar(150),

"result" varchar(15),

"result\_margin" integer,

"eliminator" varchar(20),

"method" varchar(20),

"umpire1" varchar(50),

"umpire2" varchar(50)

);

2. Create a table named ‘deliveries’ with appropriate data types for columns

Create Table deliveries ( "id" INTEGER,

"inning" INTEGER,

"over" INTEGER,

"ball" INTEGER,

"batsman" varchar(50),

"non\_striker" varchar(50),

"bowler" varchar(50),

"batsman\_runs" INTEGER,

"extra\_runs" INTEGER,

"total\_runs" INTEGER,

"is\_wicket" INTEGER,

"dismissal\_kind" varchar(25),

"player\_dismissed" varchar(50),

"fielder" varchar(50),

"extras\_type" varchar(20),

"batting\_team" varchar(50),

"bowling\_team" varchar(50)

);

3. Import data from CSV file "IPL\_matches.csv" attached in resources to ‘matches’

COPY matches FROM

'E:\cdrive\start\_tech\_inteern\sql\IPL\_matches.csv' DELIMITER ',' CSV HEADER;

4. Import data from CSV file "IPL\_Ball.csv" attached in resources to ‘deliveries’

COPY deliveries FROM

'E:\cdrive\start\_tech\_inteern\sql\IPL\_Ball.csv' DELIMITER ',' CSV HEADER;

4b. An overview of the datasets

SELECT \*

FROM matches

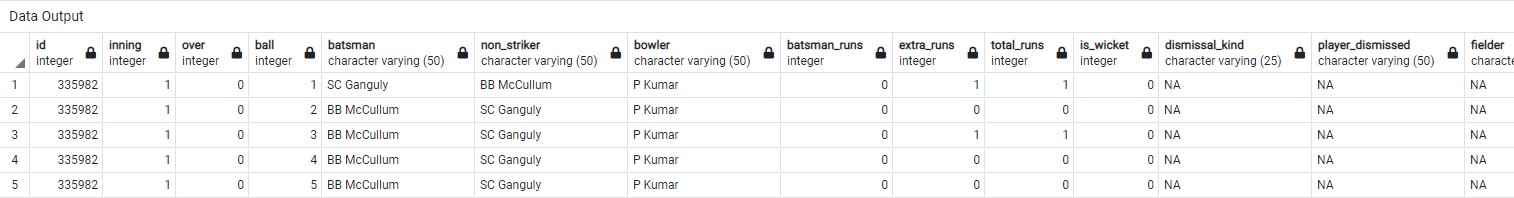
LIMIT 5;



SELECT \*

FROM deliveries

LIMIT 5;

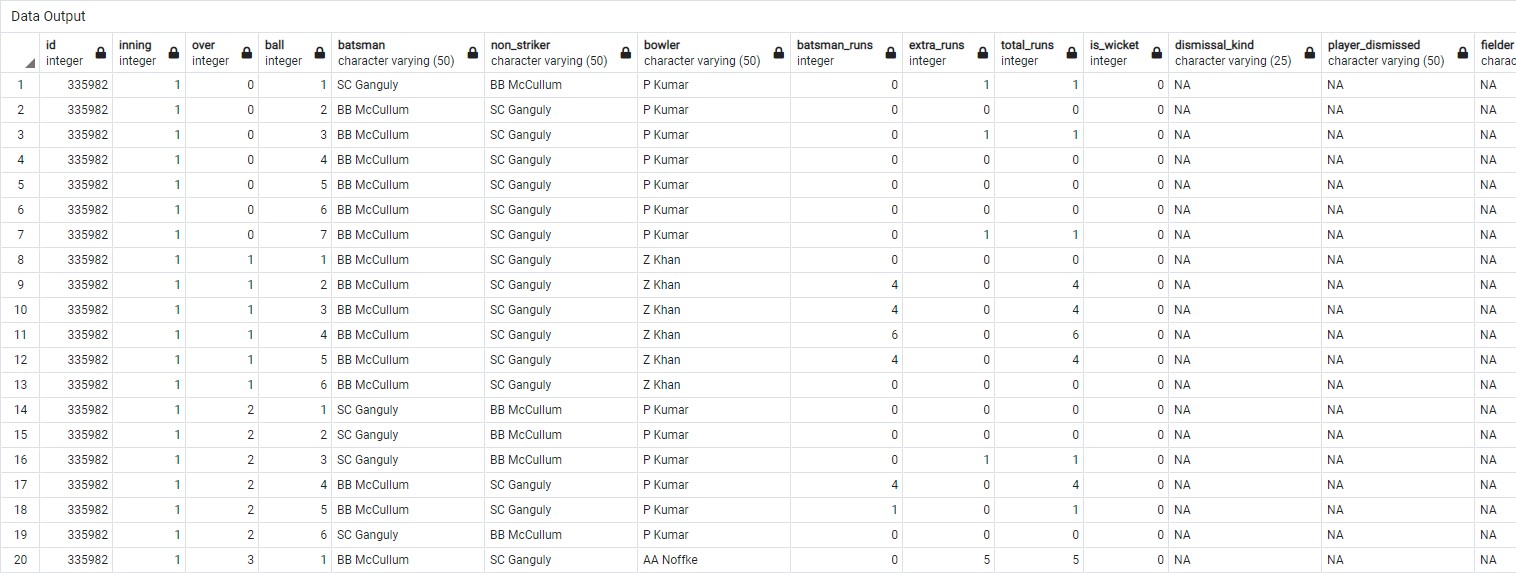


5. Select the top 20 rows of the deliveries table.

SELECT \*

FROM deliveries

LIMIT 20;

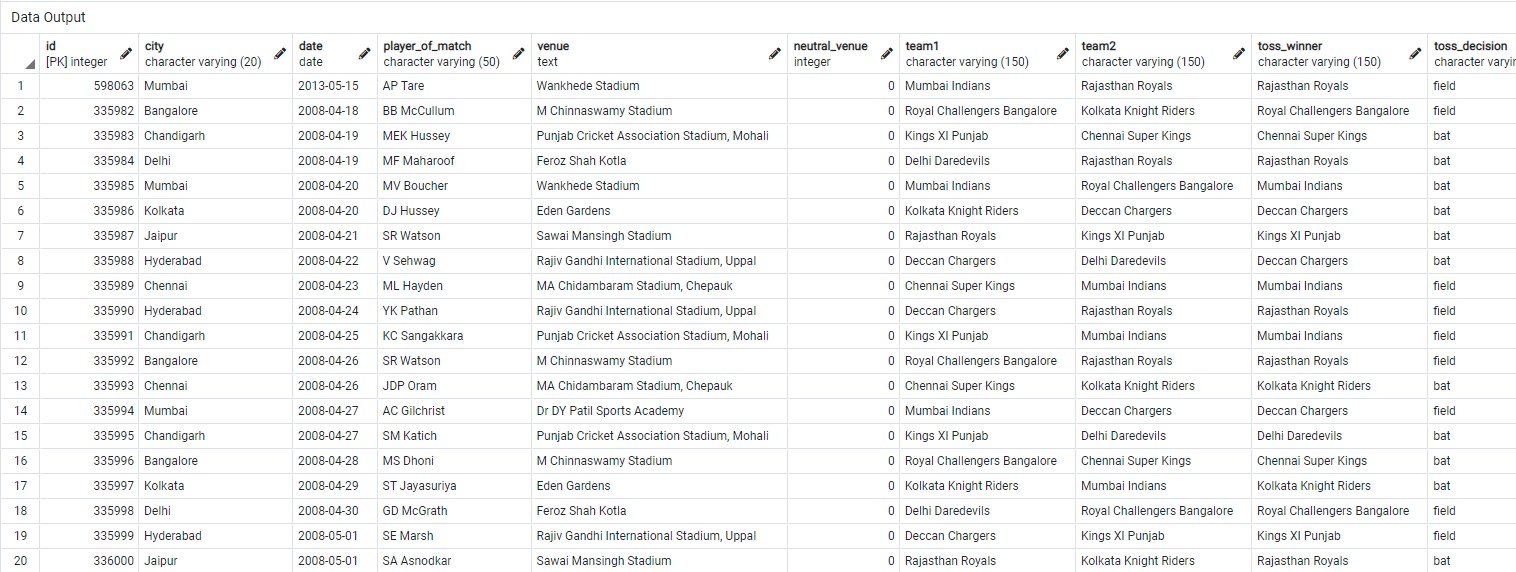


6. Select the top 20 rows of the matches table.

SELECT \*

FROM matches

LIMIT 20;



7. Fetch data of all the matches played on 2nd May 2013.

/\* My date is in Varchar datatype, so I will change it to date format and convert it to date datatype\*/

UPDATE matches

SET date = TO\_DATE(date, 'DD-MM-YYYY');

Alter table matches

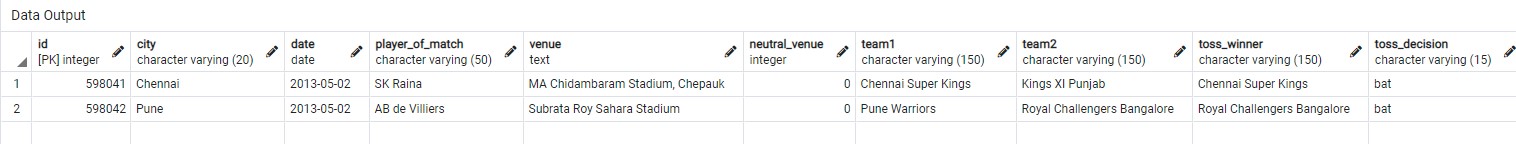
Alter column date SET DATA TYPE date

USING date::date;

SELECT \*

FROM matches

WHERE date = '2013-05-02';



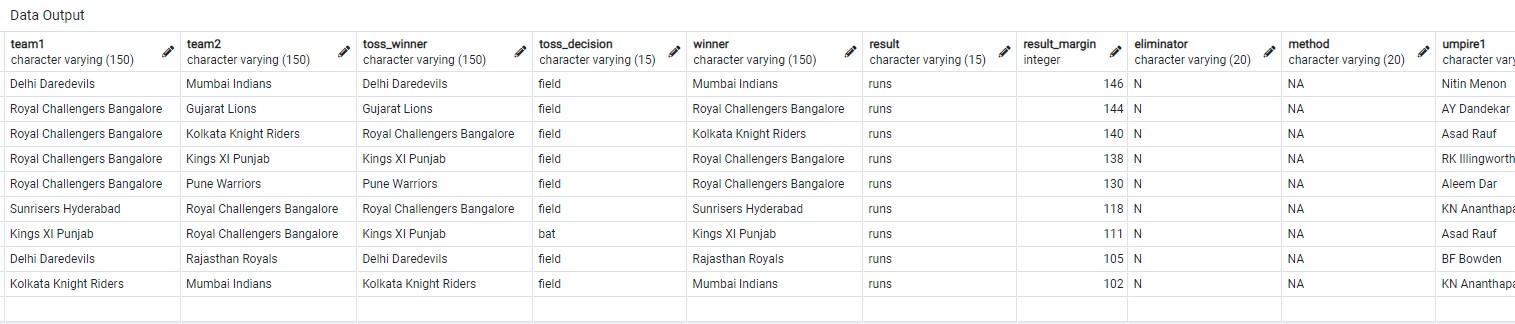
8. Fetch data of all the matches where the margin of victory is more than 100 runs.

SELECT \*

FROM matches

WHERE result\_margin > 100

ORDER BY result\_margin DESC;



9. Fetch data of all the matches where the final scores of both teams tied and order it in descending order of the date.

SELECT \*

FROM matches

WHERE result = 'tie'

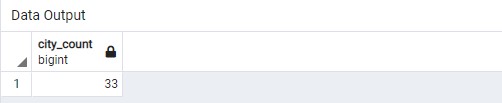
ORDER BY date DESC;



10. Get the count of cities that have hosted an IPL match.

SELECT COUNT(DISTINCT city) city\_count

FROM matches



/\* If you want to analyze data further you can write your own queries.\*/

11. Get the number of times each city have hosted an IPL match, in descending order.

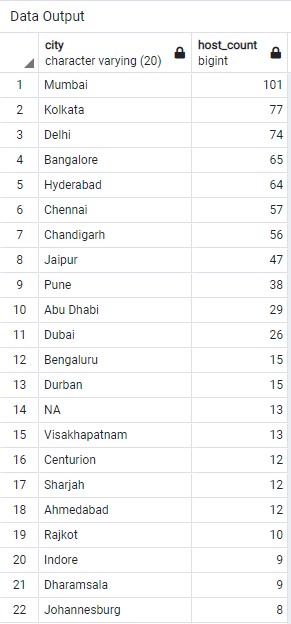
SELECT city,

COUNT(\*) host\_count

FROM matches

GROUP BY city

ORDER BY 2 DESC;



12. Number of games played each year

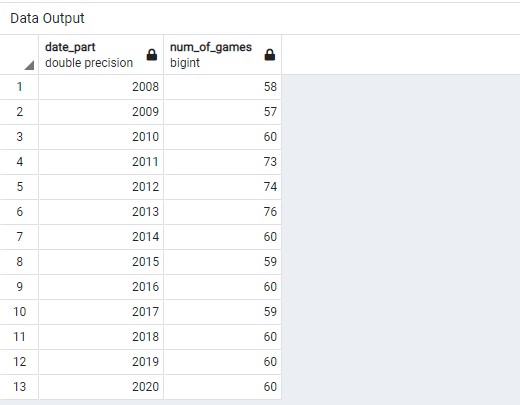
SELECT EXTRACT('year' from date),

COUNT(\*) num\_of\_games

FROM matches

GROUP BY 1

ORDER BY 1



13. Get the number of times a player has been named player of match, in descending order.

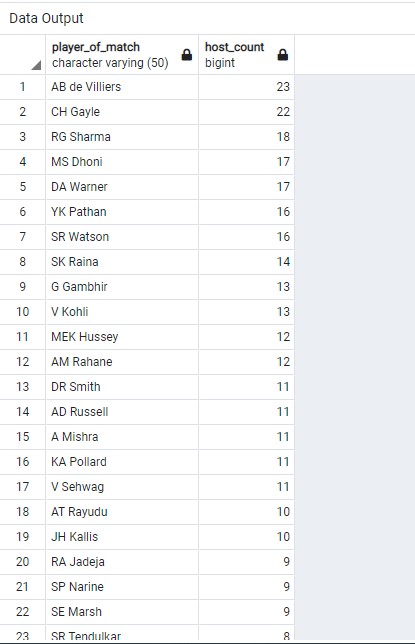
SELECT player\_of\_match,

COUNT(\*) host\_count

FROM matches

GROUP BY 1

ORDER BY 2 DESC;



14. Return a query showing home and away wins.

WITH home\_away as (

Select city,

venue,

team1,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

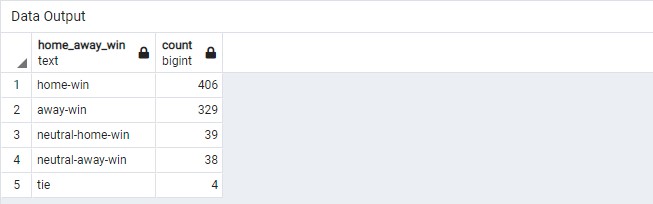
FROM matches)

Select home\_away\_win, COUNT(\*)

FROM home\_away

GROUP BY 1

ORDER BY 2 DESC;



15. Return a query showing home and away wins count by city.

WITH home\_away as (

Select city,

venue,

team1,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

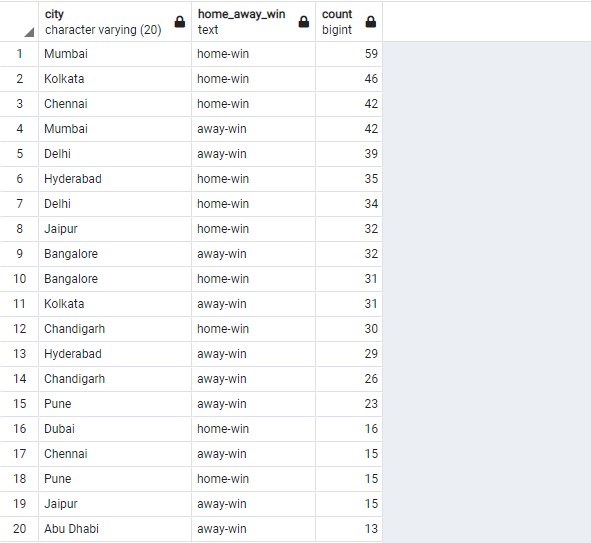
FROM matches)

Select city, home\_away\_win, COUNT(\*)

FROM home\_away

GROUP BY 1,2

ORDER BY 3 DESC;



16. Which team has recorded highest home win.

WITH home\_away as (

Select city,

venue,

team1,

team2,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

FROM matches)

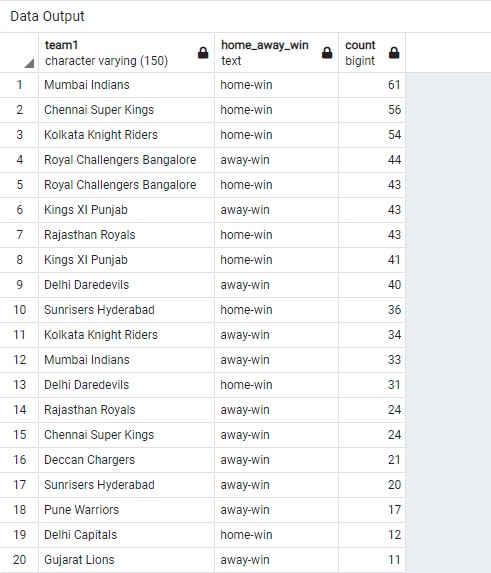
Select team1, home\_away\_win, COUNT(\*)

FROM home\_away

WHERE home\_away\_win='home-win'

GROUP BY 1,2

ORDER BY 3 DESC;



17. Which team has recorded highest away win.

WITH home\_away as (

Select city,

venue,

team1,

team2,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

FROM matches)

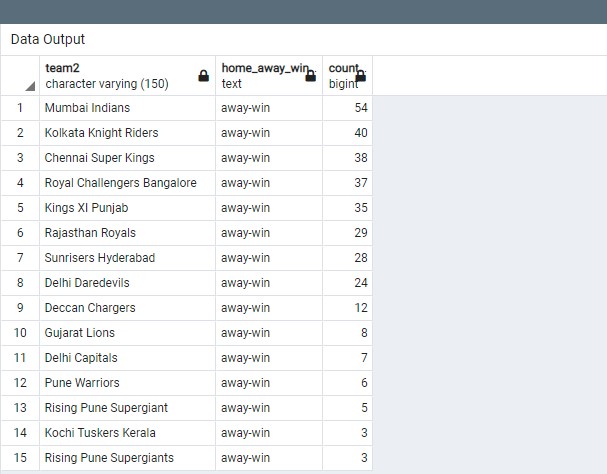
Select team2, home\_away\_win, COUNT(\*)

FROM home\_away

WHERE home\_away\_win='away-win'

GROUP BY 1,2

ORDER BY 3 DESC;



18. Which stadium has recorded highest away win.

WITH home\_away as (

Select city,

venue,

team1,

team2,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

FROM matches)

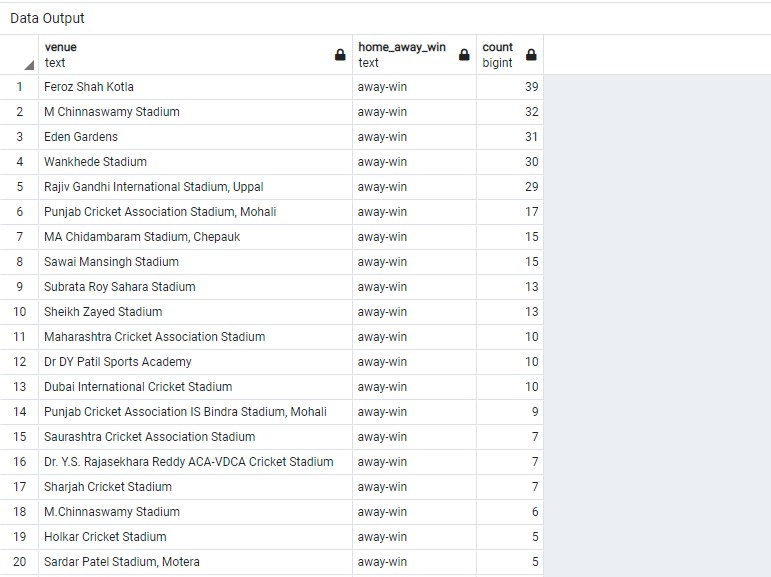
Select venue, home\_away\_win, COUNT(\*)

FROM home\_away

WHERE home\_away\_win='away-win'

GROUP BY 1,2

ORDER BY 3 DESC;



19. Which stadium has recorded highest home win.

WITH home\_away as (

Select city,

venue,

team1,

team2,

neutral\_venue,

CASE WHEN (neutral\_venue=0) AND (team1= winner) THEN 'home-win'

WHEN (neutral\_venue=0) AND (team2 = winner) THEN 'away-win'

WHEN (neutral\_venue=1) AND (team1 = winner) THEN 'neutral-home-win'

WHEN (neutral\_venue=1) AND (team2 = winner) THEN 'neutral-away-win'

ELSE 'tie'

END AS "home\_away\_win"

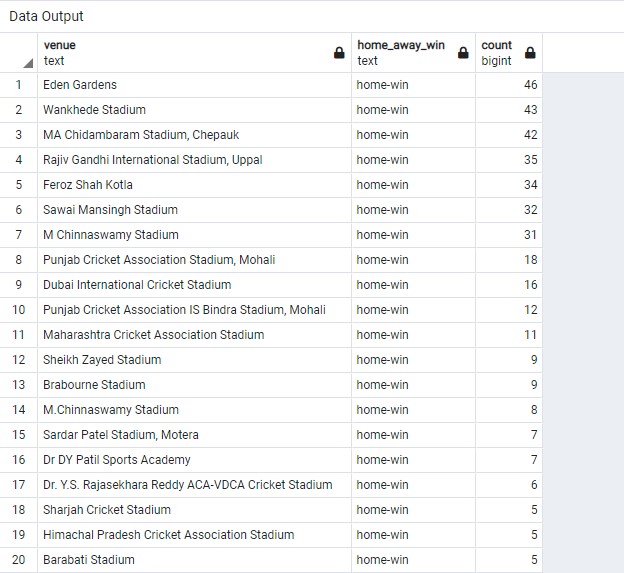
FROM matches)

Select venue, home\_away\_win, COUNT(\*)

FROM home\_away

WHERE home\_away\_win='home-win'

GROUP BY 1,2



ORDER BY 3 DESC;

20. Number of Teams

SELECT team1

FROM matches

Union

Select team2

FROM matches;



21. Home team number of games, number of wins, winning ratio

SElECT ma.team1,

COUNT(\*) num\_of\_games,

t1.num\_of\_wins,

(t1.num\_of\_wins/COUNT(\*)::float) win\_ratio

FROM matches ma

JOIN ( SELECT team1,

COUNT(\*) num\_of\_wins

FROM matches ma

WHERE (winner = team1) AND (neutral\_venue=0)

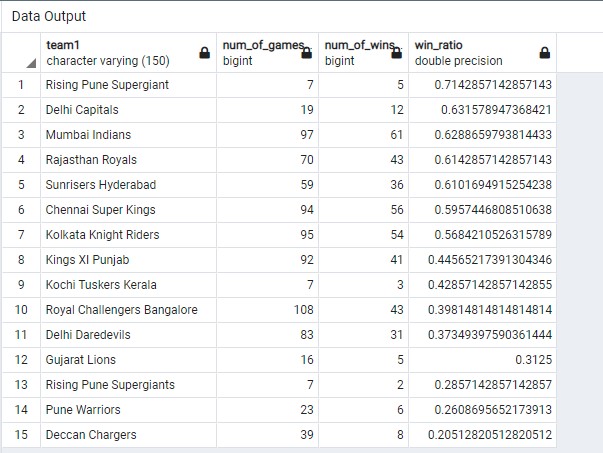
GROUP BY 1

ORDER BY 2 DESC) t1

ON ma.team1=t1.team1

GROUP BY 1,3

ORDER BY 4 DESC;



22. Away team number of games, number of wins, winning ratio

SElECT t2.team2,

COUNT(\*) num\_of\_games,

t2.num\_of\_wins,

(t2.num\_of\_wins/COUNT(\*)::float) win\_ratio

FROM matches ma

JOIN ( SELECT team2,

COUNT(\*) num\_of\_wins

FROM matches ma

WHERE (winner = team2) AND (neutral\_venue=0)

GROUP BY 1

ORDER BY 2 DESC) t2

ON ma.team2=t2.team2

GROUP BY 1,3

ORDER BY 4 DESC;

