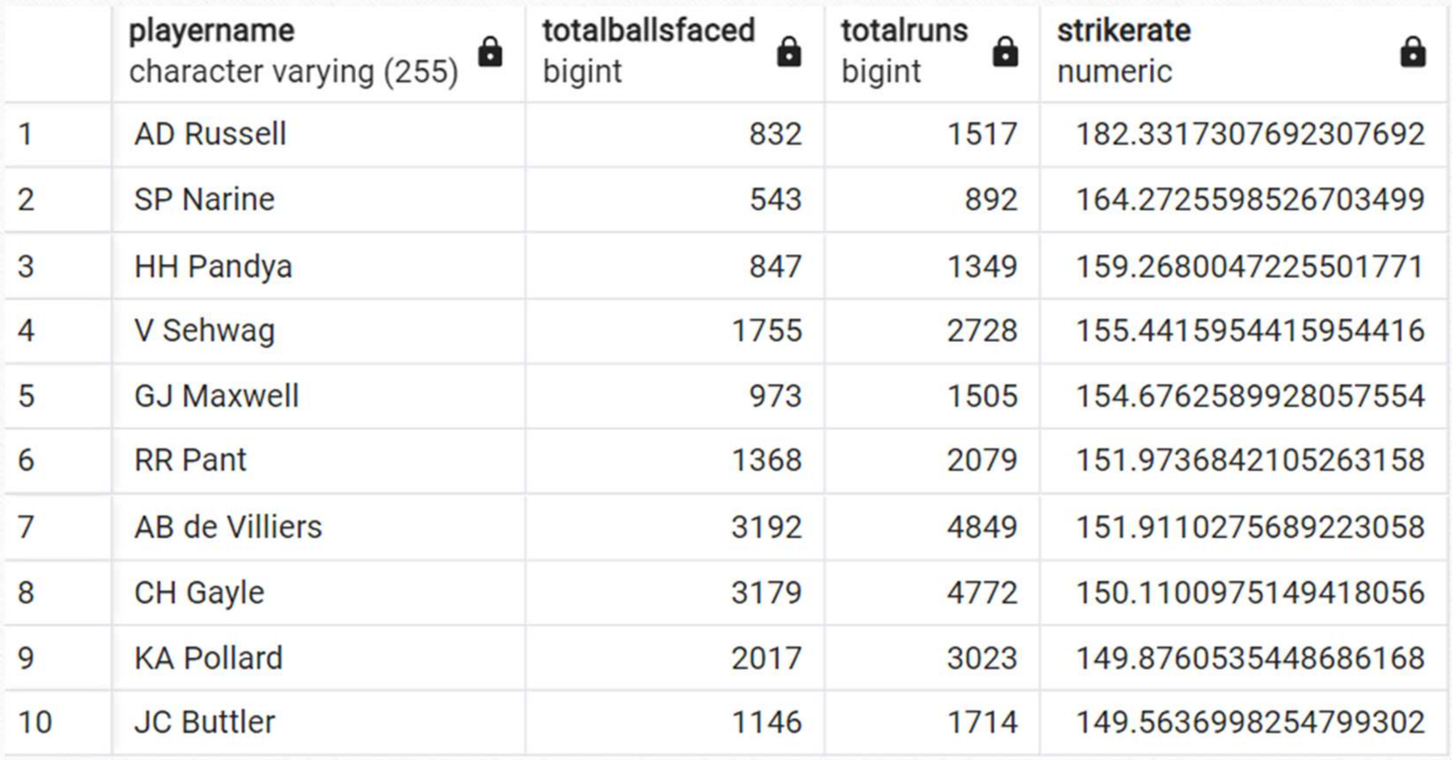
SQL Project : IPL Auction

Bharath Babu

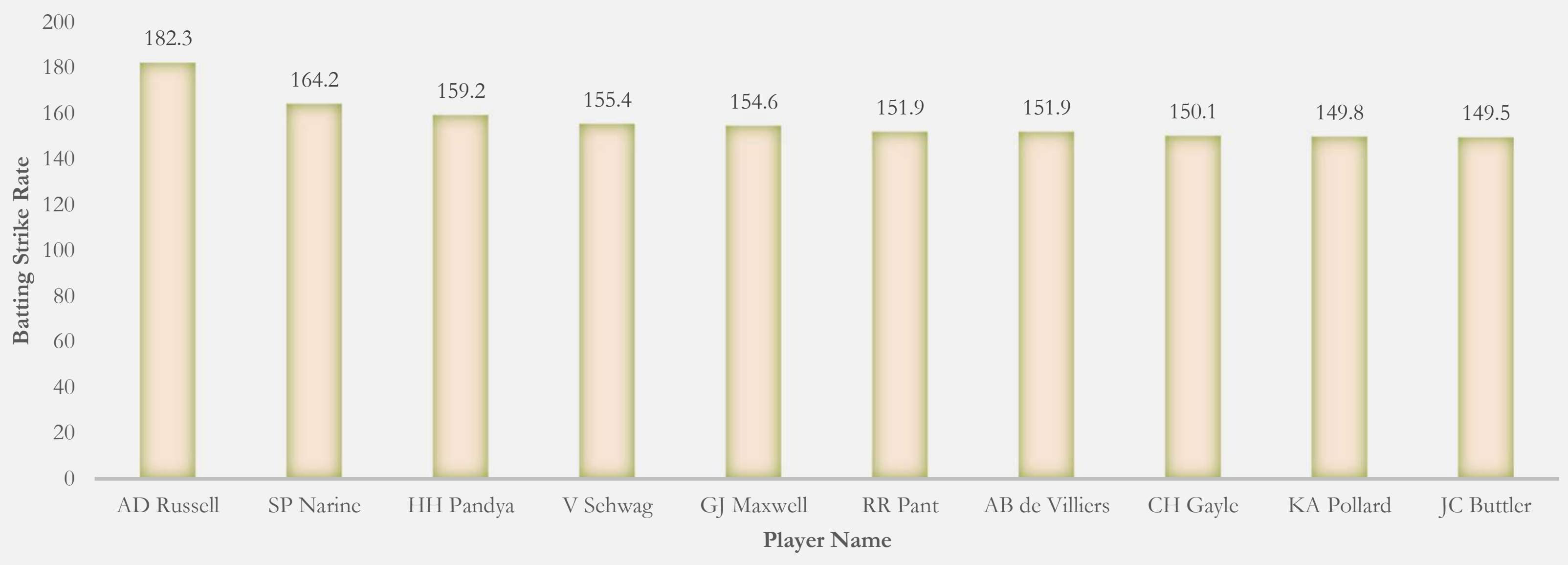
Top 10 batters with high Strike Rate

```
select batsman as playername,
       count(*) as totalballsfaced,
       sum(batsman_runs) as totalruns,
       (sum(batsman_runs) * 100.0) / count(*) as strikerate
from deliveries
where extras_type != 'wides'
group by batsman
having count(*) >= 500
order by strikerate desc
limit 10;
```



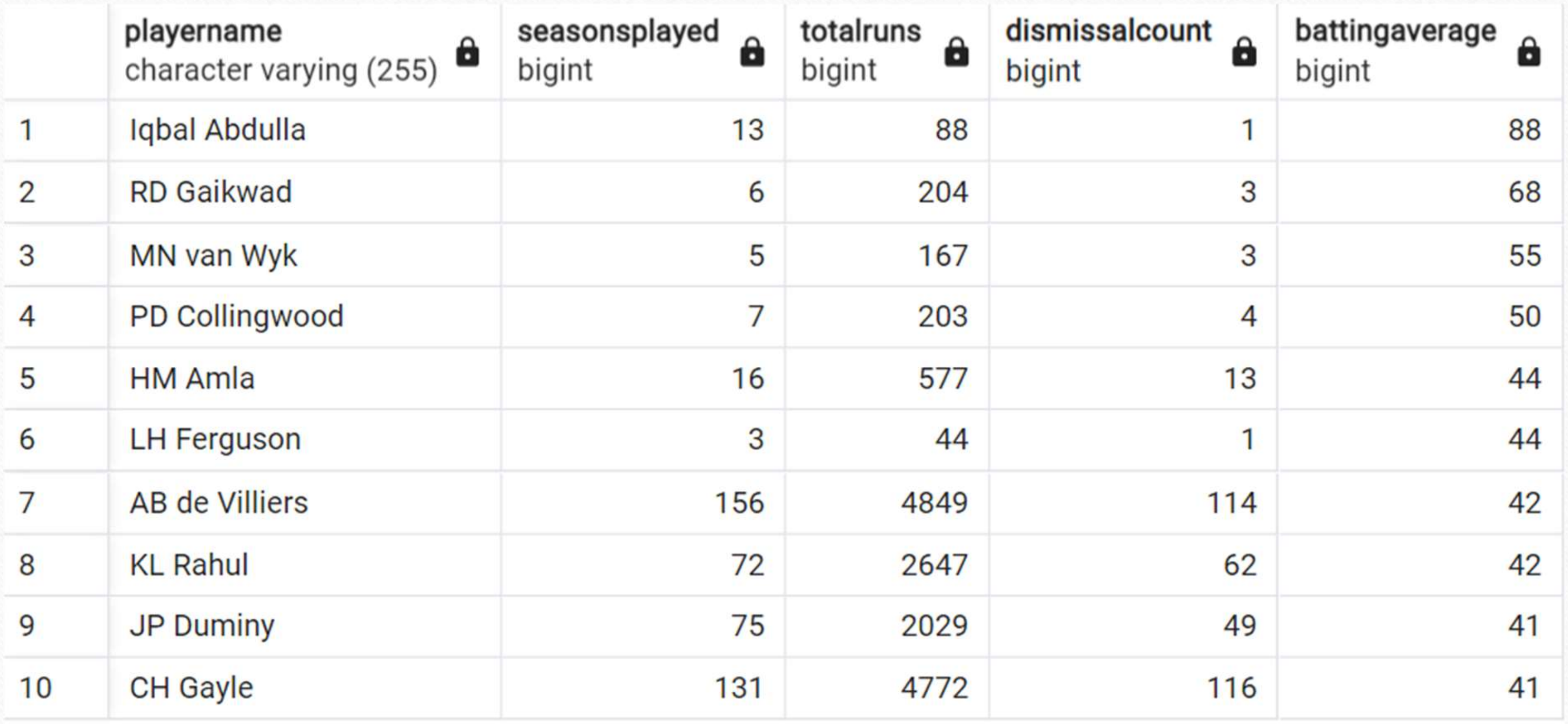
	playername character varying (255)	totalballsfaced bigint	totalruns bigint	strikerate numeric
1	AD Russell	832	1517	182.3317307692307692
2	SP Narine	543	892	164.2725598526703499
3	HH Pandya	847	1349	159.2680047225501771
4	V Sehwag	1755	2728	155.4415954415954416
5	GJ Maxwell	973	1505	154.6762589928057554
6	RR Pant	1368	2079	151.9736842105263158
7	AB de Villiers	3192	4849	151.9110275689223058
8	CH Gayle	3179	4772	150.1100975149418056
9	KA Pollard	2017	3023	149.8760535448686168
10	JC Buttler	1146	1714	149.5636998254799302

TOP 10 BATTERS WITH HIGH STRIKE RATE

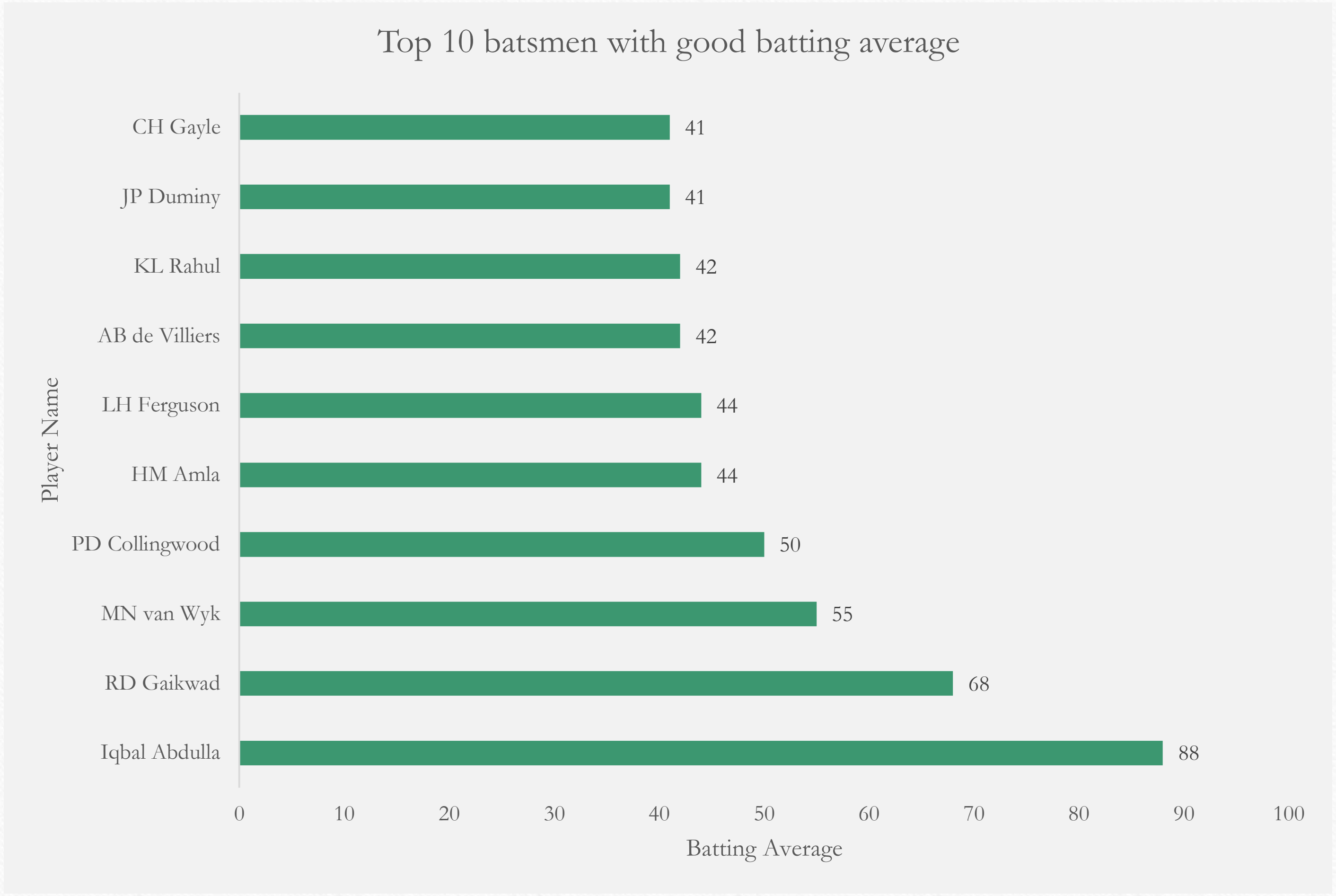


Top 10 Batsmen with Good Average

```
select batsman as playername,
       count(distinct id) as seasonsplayed,
       sum(batsman_runs) as totalruns,
       sum(case when is_wicket = 1 then 1 else 0 end) as dismissalcount,
       sum(batsman_runs) / sum(case when is_wicket = 1 then 1 else 0 end) as battingaverage
from deliveries
group by batsman
having sum(case when is_wicket = 1 then 1 else 0 end) > 0 and count(distinct id) > 2
order by battingaverage desc
limit 10;
```

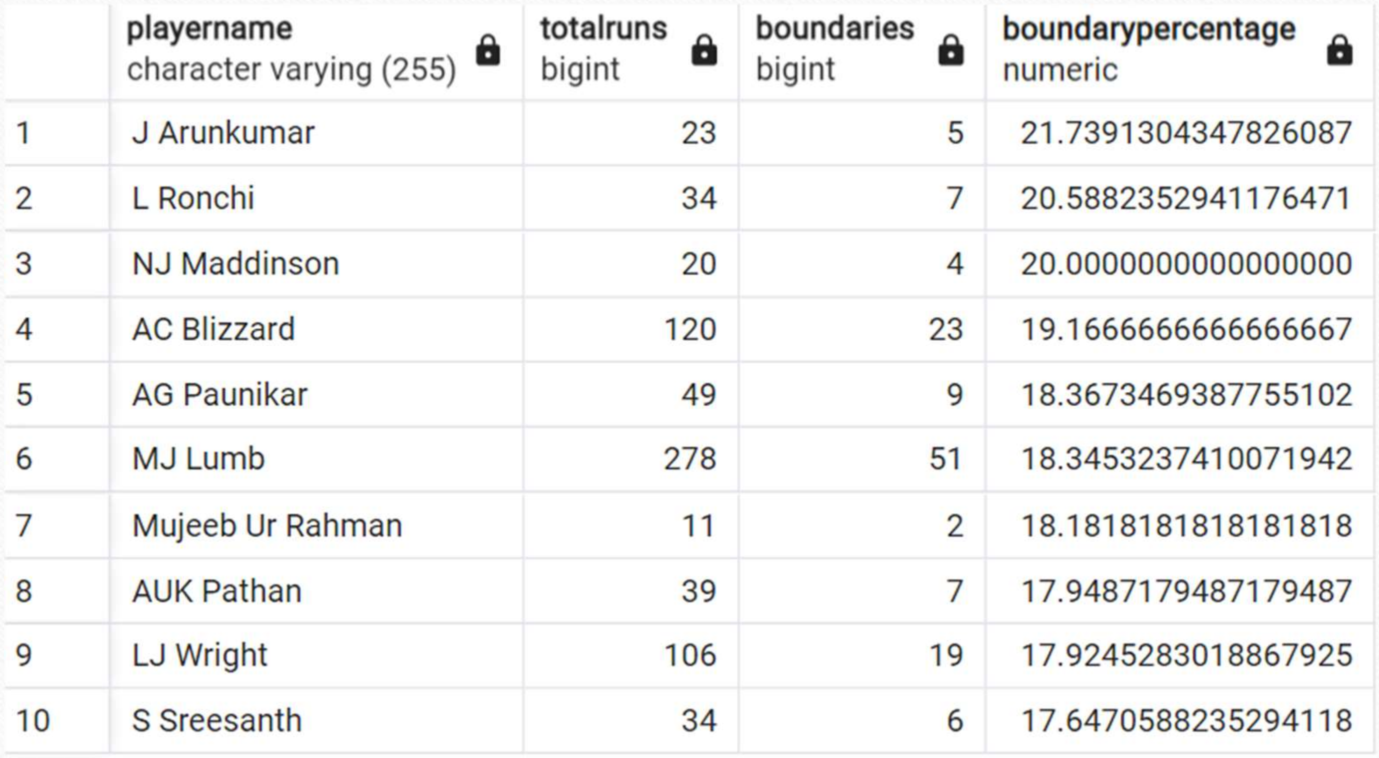


	playername character varying (255)	seasonsplayed bigint	totalruns bigint	dismissalcount bigint	battingaverage bigint
1	Iqbal Abdulla	13	88	1	88
2	RD Gaikwad	6	204	3	68
3	MN van Wyk	5	167	3	55
4	PD Collingwood	7	203	4	50
5	HM Amla	16	577	13	44
6	LH Ferguson	3	44	1	44
7	AB de Villiers	156	4849	114	42
8	KL Rahul	72	2647	62	42
9	JP Duminy	75	2029	49	41
10	CH Gayle	131	4772	116	41

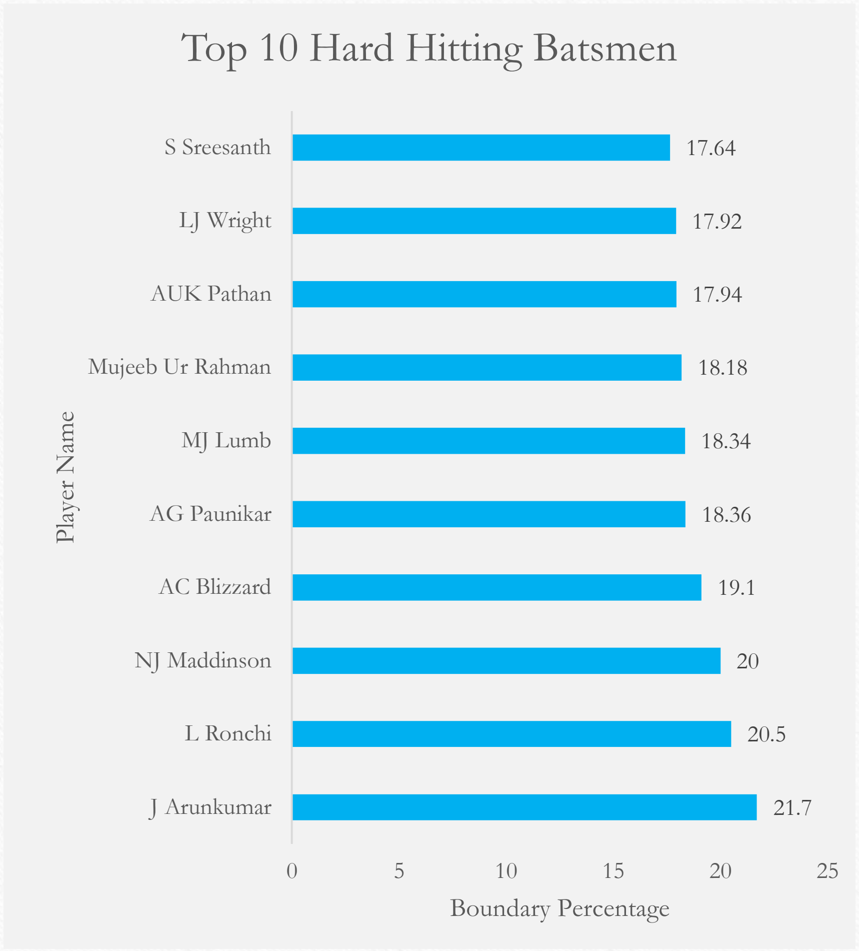

Top 10 batsmen with good batting average
Player Name
CH Gayle
41
JP Duminy
41
KL Rahul
42
AB de Villiers
42
LH Ferguson
44
HM Amla
44
PD Collingwood
50
MN van Wyk
55
RD Gaikwad
68
Iqbal Abdulla
88
0
10
20
30
40
50
60
70
80
90
100
Batting Average

Top 10 Hard Hitting Batsmen

```
select batsman as playername,
	sum(batsman_runs) as totalruns,
	sum(case when batsman_runs = 4 or batsman_runs = 6 then 1 else 0 end) as boundaries,
	(sum(case when batsman_runs = 4 or batsman_runs = 6 then 1 else 0 end) * 100.0) / sum(batsman_runs) as boundarypercentage
from deliveries
group by batsman
having count(distinct id) > 2
order by boundarypercentage desc
limit 10;
```

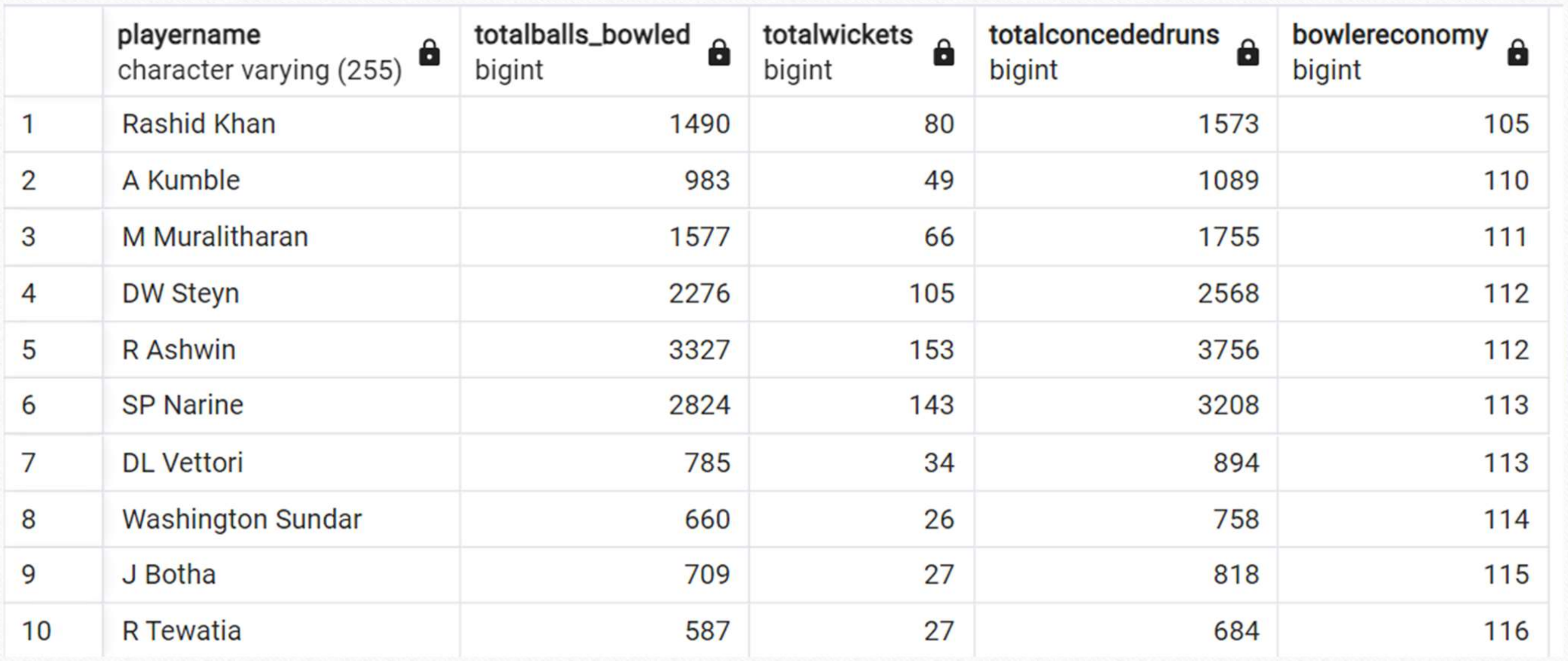


	playername character varying (255)	totalruns bigint	boundaries bigint	boundarypercentage numeric
1	J Arunkumar	23	5	21.7391304347826087
2	L Ronchi	34	7	20.5882352941176471
3	NJ Maddinson	20	4	20.0000000000000000
4	AC Blizzard	120	23	19.1666666666666667
5	AG Paunikar	49	9	18.3673469387755102
6	MJ Lumb	278	51	18.3453237410071942
7	Mujeeb Ur Rahman	11	2	18.1818181818181818
8	AUK Pathan	39	7	17.9487179487179487
9	LJ Wright	106	19	17.9245283018867925
10	S Sreesanth	34	6	17.6470588235294118


Top 10 Hard Hitting Batsmen
Player Name
S Sreesanth
17.64
LJ Wright
17.92
AUK Pathan
17.94
Mujeeb Ur Rahman
18.18
MJ Lumb
18.34
AG Paunikar
18.36
AC Blizzard
19.1
NJ Maddinson
20
L Ronchi
20.5
J Arunkumar
21.7
0
5
10
15
20
25
Boundary Percentage

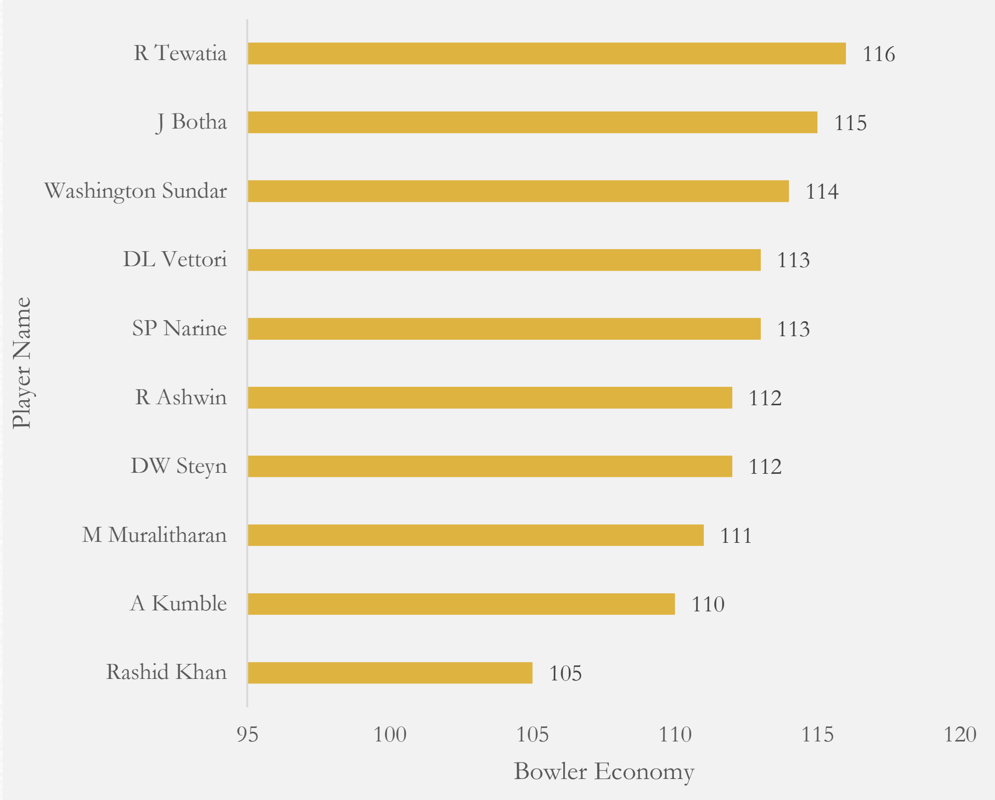
Top 10 Bowlers with Economy

```
select bowler as playername,
        count(*) as totalballs_bowled,
        sum(is_wicket) as totalwickets,
        sum(total_runs) as totalconcededruns,
        (sum(total_runs) * 100) / count(*) as bowlereconomy
from deliveries
group by bowler
having count(*) >= 500
order by bowlereconomy asc
limit 10;
```



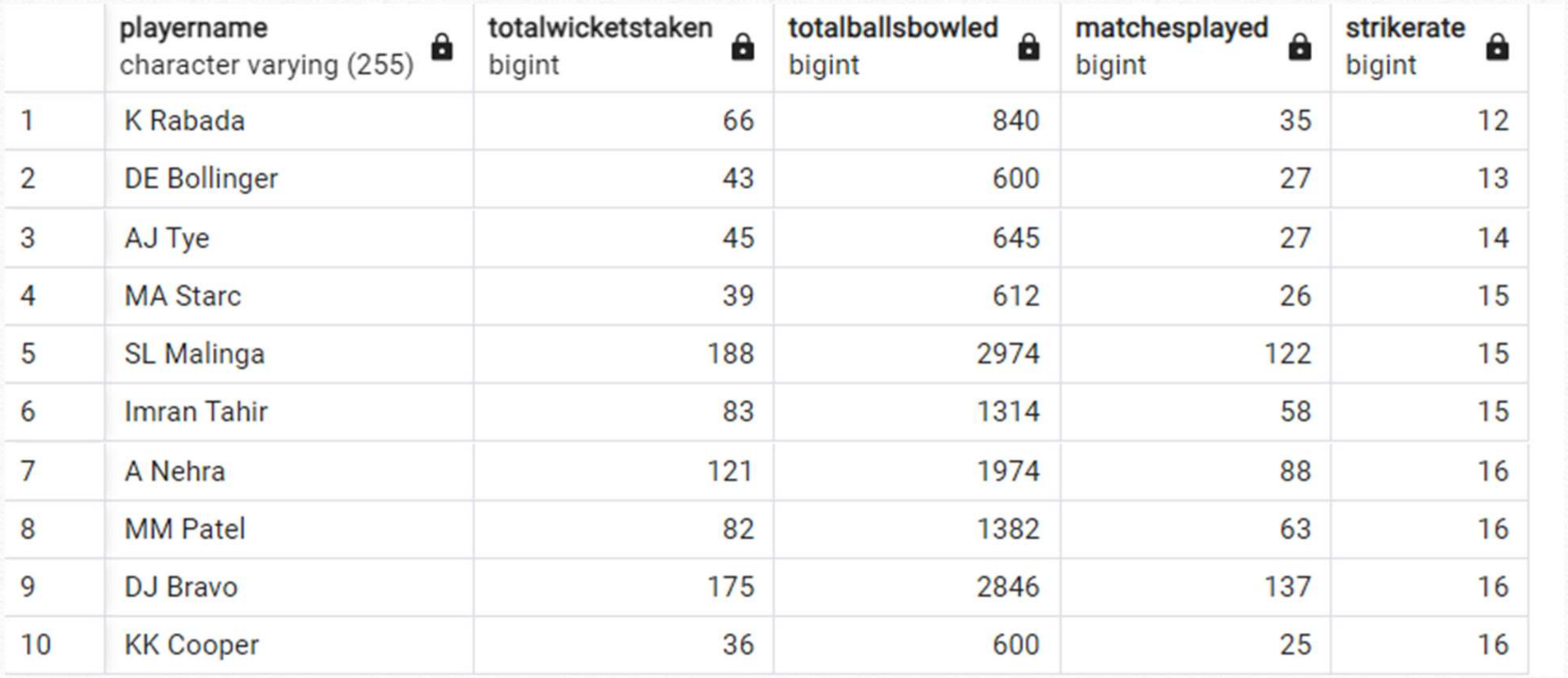
	playername character varying (255)	totalballs_bowled bigint	totalwickets bigint	totalconcededruns bigint	bowlereconomy bigint
1	Rashid Khan	1490	80	1573	105
2	A Kumble	983	49	1089	110
3	M Muralitharan	1577	66	1755	111
4	DW Steyn	2276	105	2568	112
5	R Ashwin	3327	153	3756	112
6	SP Narine	2824	143	3208	113
7	DL Vettori	785	34	894	113
8	Washington Sundar	660	26	758	114
9	J Botha	709	27	818	115
10	R Tewatia	587	27	684	116

Top 10 Bowlers with Economy

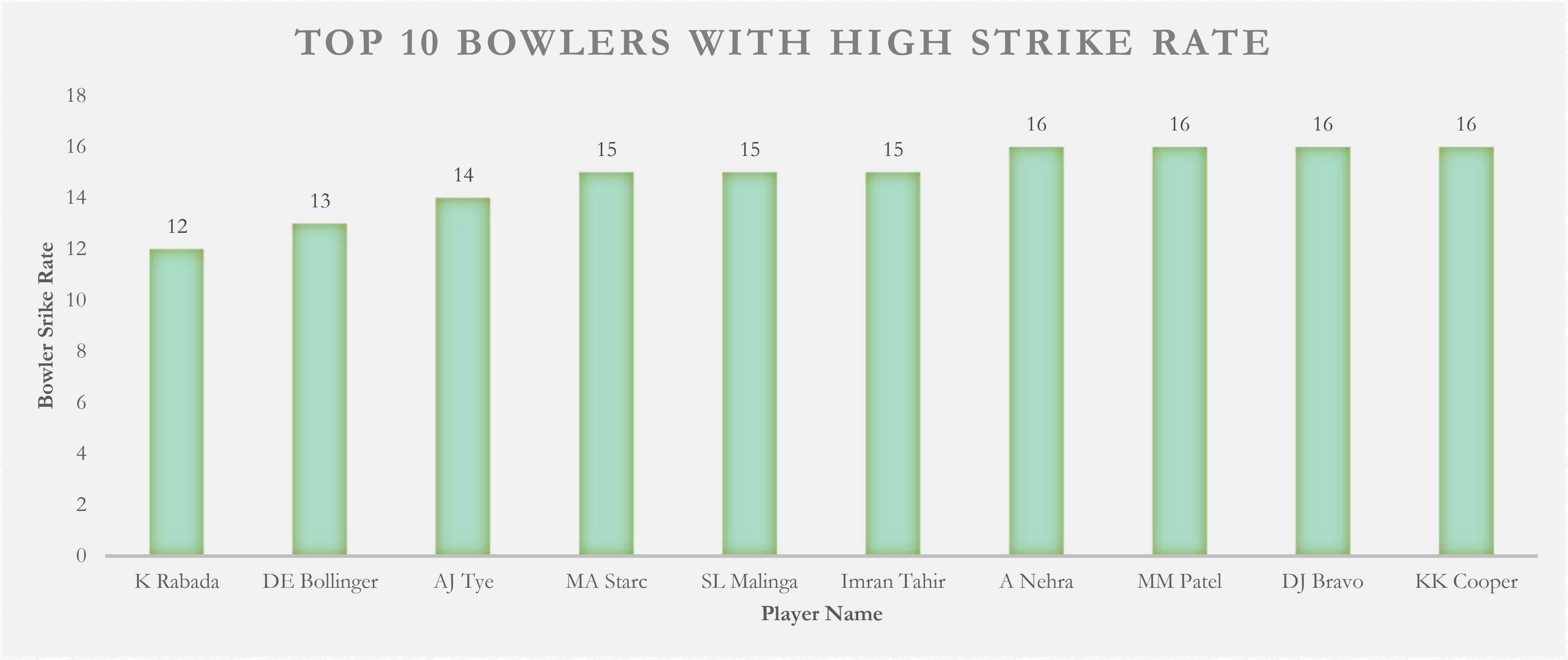


Top 10 Bowlers with High Strike Rate

```
select bowler as playername,
    sum(is_wicket) as totalwicketstaken,
    count(*) as totalballsbowled,
    count(distinct id) as matchesplayed,
    count(*) / sum(is_wicket) as strikerate
from deliveries
group by bowler
having count(*) >= 500
order by strikerate asc
limit 10;
```

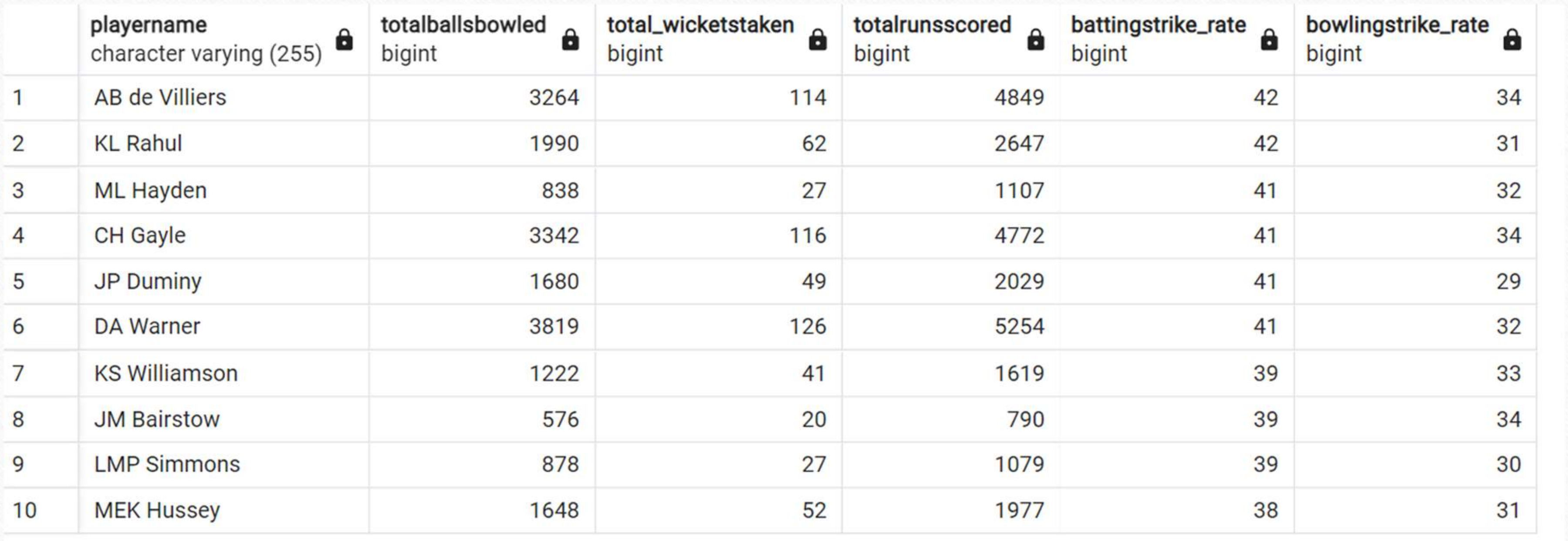


	playername character varying (255)	totalwicketstaken bigint	totalballsbowled bigint	matchesplayed bigint	strikerate bigint
1	K Rabada	66	840	35	12
2	DE Bollinger	43	600	27	13
3	AJ Tye	45	645	27	14
4	MA Starc	39	612	26	15
5	SL Malinga	188	2974	122	15
6	Imran Tahir	83	1314	58	15
7	A Nehra	121	1974	88	16
8	MM Patel	82	1382	63	16
9	DJ Bravo	175	2846	137	16
10	KK Cooper	36	600	25	16

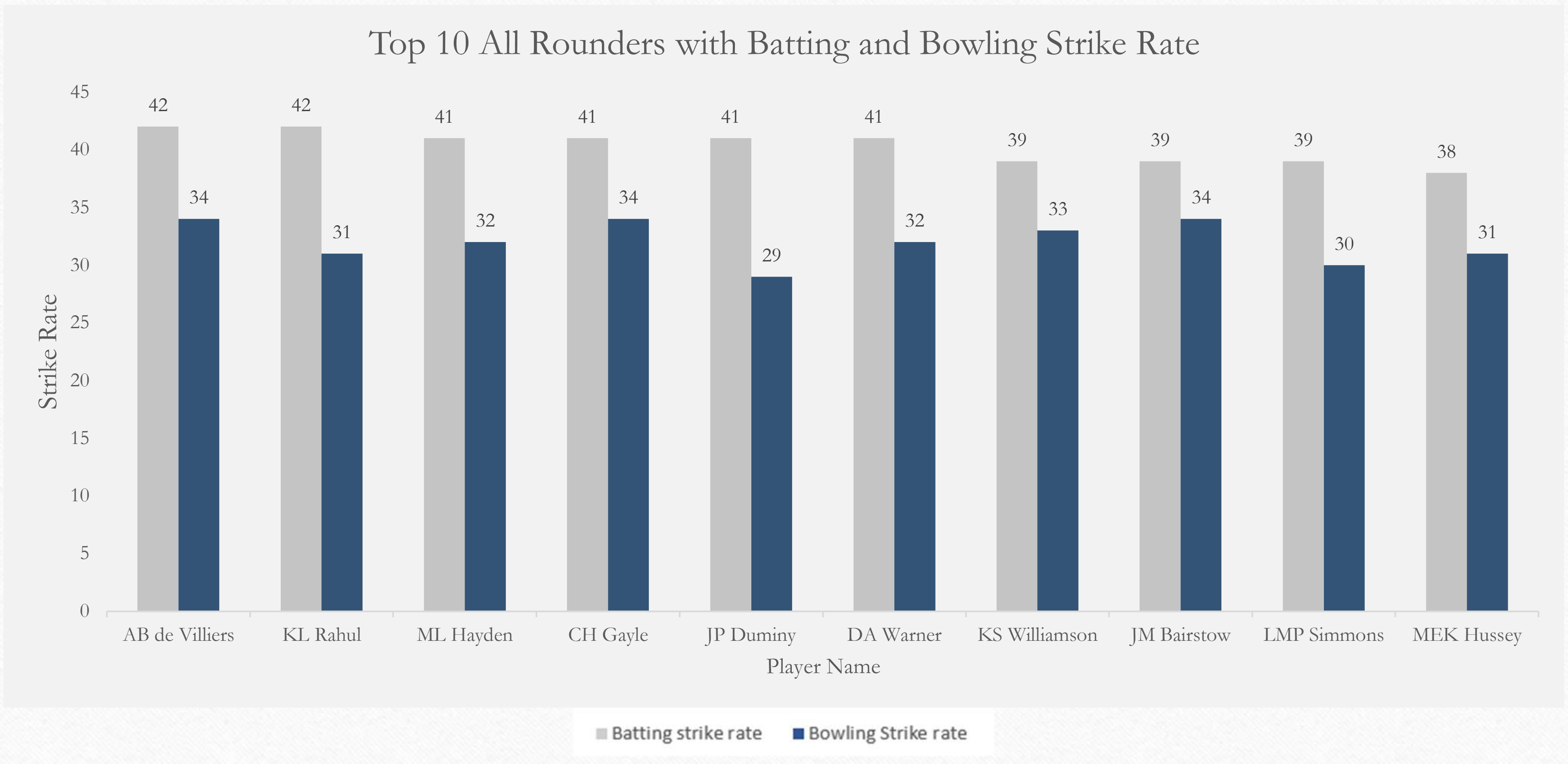

TOP 10 BOWLERS WITH HIGH STRIKE RATE
Bowler Srike Rate
Player Name
0
2
4
6
8
10
12
14
16
18
K Rabada 12
DE Bollinger 13
AJ Tye 14
MA Starc 15
SL Malinga 15
Imran Tahir 15
A Nehra 16
MM Patel 16
DJ Bravo 16
KK Cooper 16

Top 10 All Rounders

```
select batsman as playername,
    count(*) as totalballsbowled,
    sum(is_wicket) as total_wicketstaken,
    sum(batsman_runs) as totalrunsscored,
    sum(batsman_runs) / sum(is_wicket) as battingstrike_rate,
    sum(is_wicket) * 1000 / count(*) as bowlingstrike_rate
from deliveries
group by batsman
having count(*) >= 500
order by sum(batsman_runs) / sum(is_wicket) desc,
     sum(is_wicket) / count(*) asc
limit 10;
```



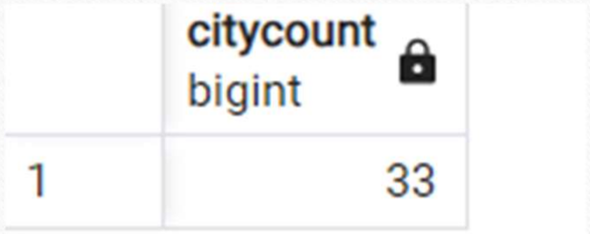
	playername character varying (255)	totalballsbowled bigint	total_wicketstaken bigint	totalrunsscored bigint	battingstrike_rate bigint	bowlingstrike_rate bigint
1	AB de Villiers	3264	114	4849	42	34
2	KL Rahul	1990	62	2647	42	31
3	ML Hayden	838	27	1107	41	32
4	CH Gayle	3342	116	4772	41	34
5	JP Duminy	1680	49	2029	41	29
6	DA Warner	3819	126	5254	41	32
7	KS Williamson	1222	41	1619	39	33
8	JM Bairstow	576	20	790	39	34
9	LMP Simmons	878	27	1079	39	30
10	MEK Hussey	1648	52	1977	38	31


Top 10 All Rounders with Batting and Bowling Strike Rate
Strike Rate
Player Name
45
40
35
30
25
20
15
10
5
0
AB de Villiers 42 34
KL Rahul 42 31
ML Hayden 41 32
CH Gayle 41 34
JP Duminy 41 29
DA Warner 41 32
KS Williamson 39 33
JM Bairstow 39 34
LMP Simmons 39 30
MEK Hussey 38 31
Batting strike rate
Bowling Strike rate

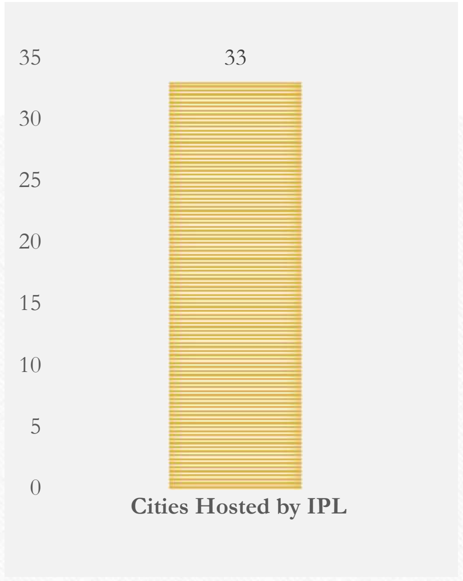
1. Get the count of cities that have hosted an IPL match

count of cities

```
select count(distinct city) as citycount
from matches;
```

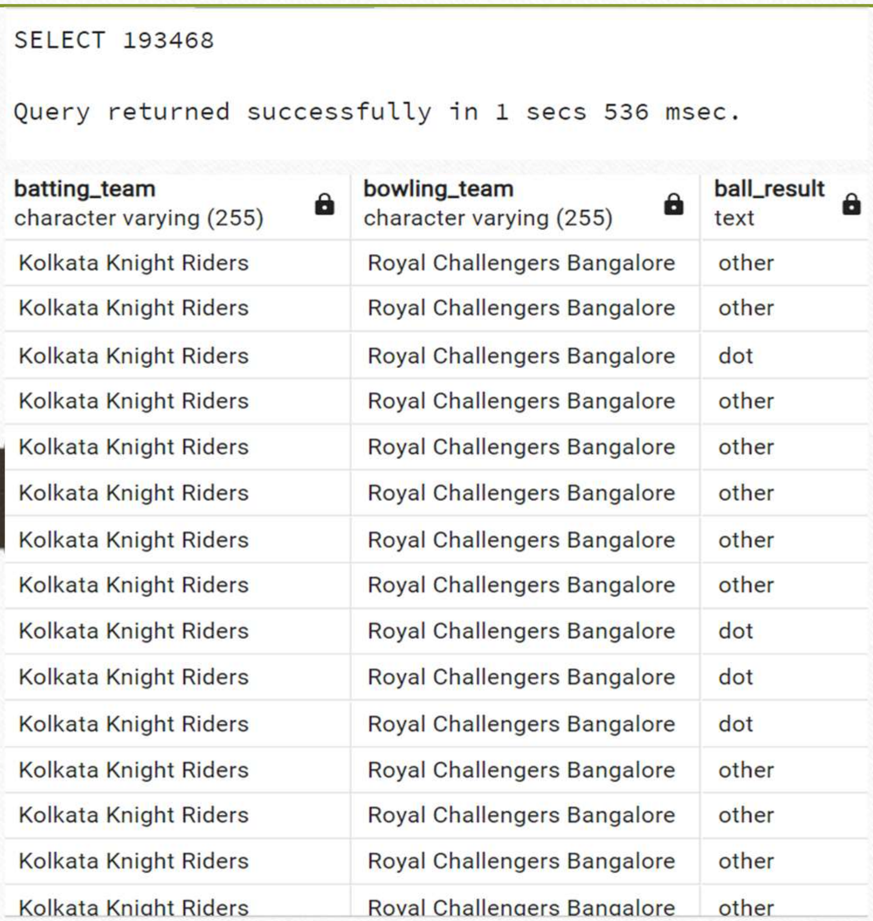


	citycount bigint
1	33



2 Create table deliveries_v02 with all the columns of the table ‘deliveries’ and an additional column ball_result containing values boundary, dot or other depending on the total_run (boundary for >= 4, dot for 0 and other for any other number)

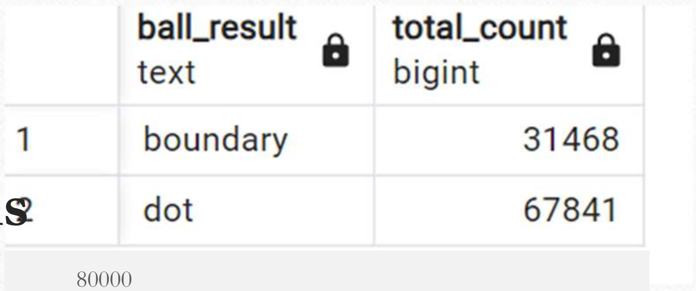
```
create table deliveries_v02 as
select *, case when total_runs >= 4 then 'boundary'
               when total_runs = 0 then 'dot'
               else 'other'
          end as ball_result
from deliveries;
```



```
SELECT 193468

Query returned successfully in 1 secs 536 msec.
```

batting_team character varying (255)	bowling_team character varying (255)	ball_result text
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	dot
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	dot
Kolkata Knight Riders	Royal Challengers Bangalore	dot
Kolkata Knight Riders	Royal Challengers Bangalore	dot
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other
Kolkata Knight Riders	Royal Challengers Bangalore	other

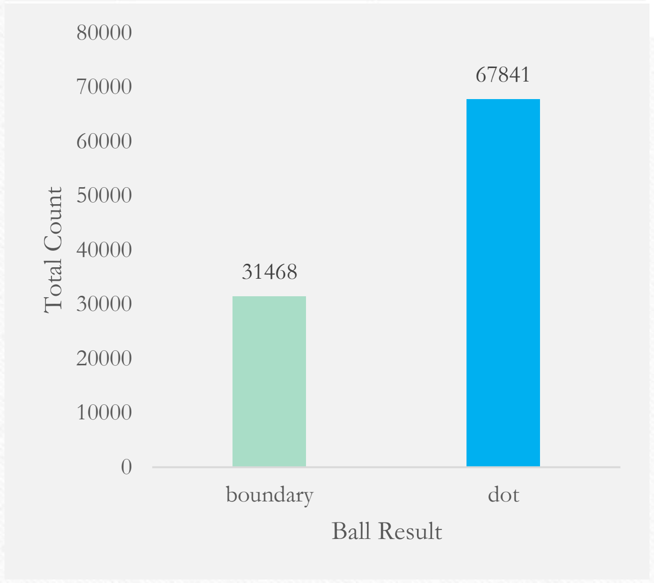
3. Write a query to fetch the total number of boundaries and dot balls from the deliveries_v02 table.



	ball_result text	total_count bigint
1	boundary	31468
2	dot	67841

Total number of boundaries and dot balls

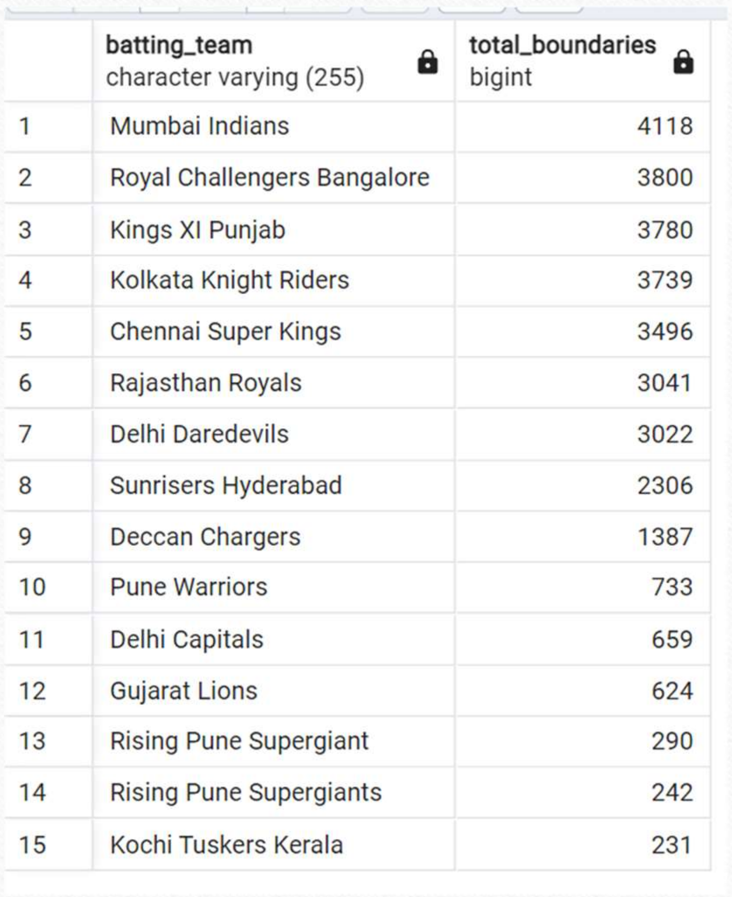
```
select ball_result, count(*) as total_count
from deliveries_v02
where ball_result in ('boundary', 'dot')
group by ball_result;
```



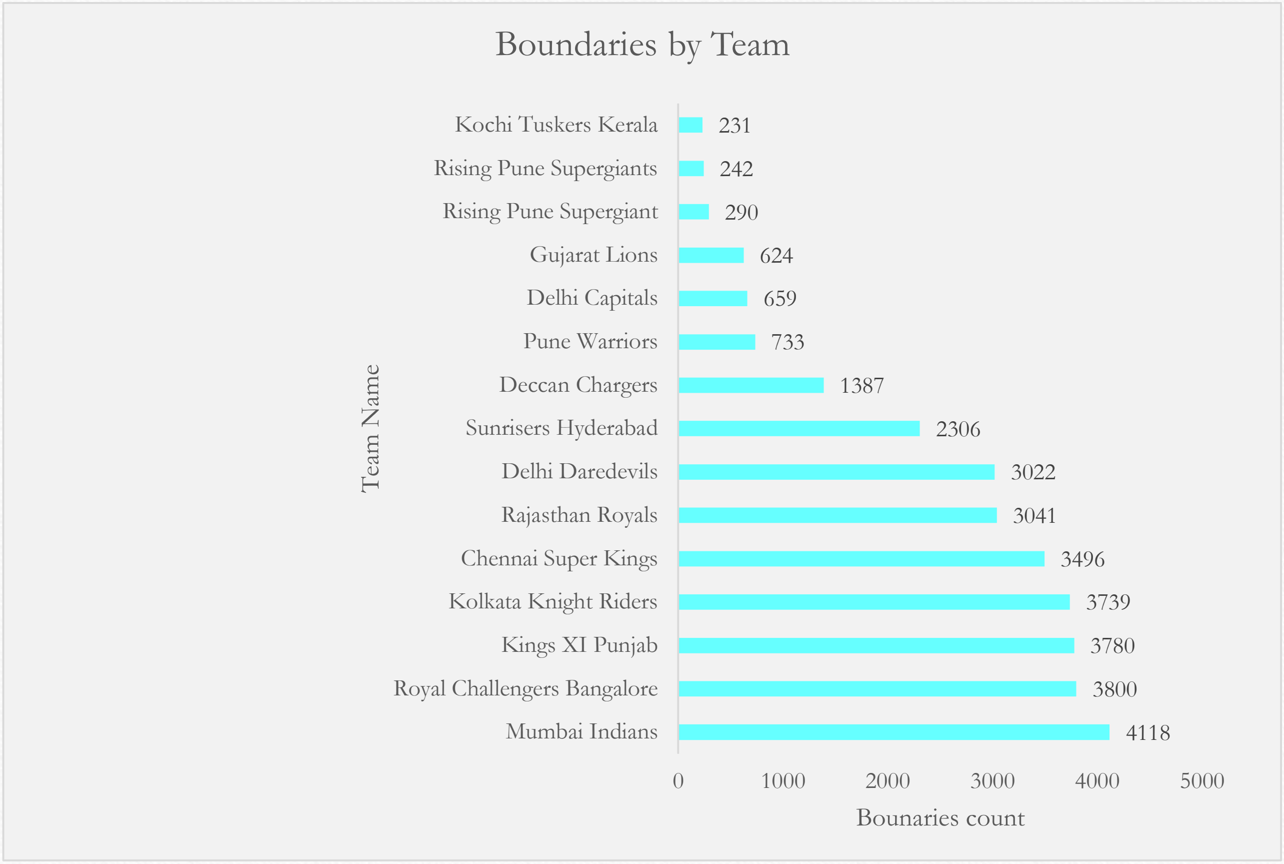
4. Write a query to fetch the total number of boundaries scored by each team from the deliveries_v02 table and order it in descending order of the number of boundaries scored.

Boundaries scored by each team

```
select batting_team, count(*) as total_boundaries
from deliveries_v02
where ball_result = 'boundary'
group by batting_team
order by total_boundaries desc;
```



	batting_team character varying (255)	total_boundaries bigint
1	Mumbai Indians	4118
2	Royal Challengers Bangalore	3800
3	Kings XI Punjab	3780
4	Kolkata Knight Riders	3739
5	Chennai Super Kings	3496
6	Rajasthan Royals	3041
7	Delhi Daredevils	3022
8	Sunrisers Hyderabad	2306
9	Deccan Chargers	1387
10	Pune Warriors	733
11	Delhi Capitals	659
12	Gujarat Lions	624
13	Rising Pune Supergiant	290
14	Rising Pune Supergiants	242
15	Kochi Tuskers Kerala	231

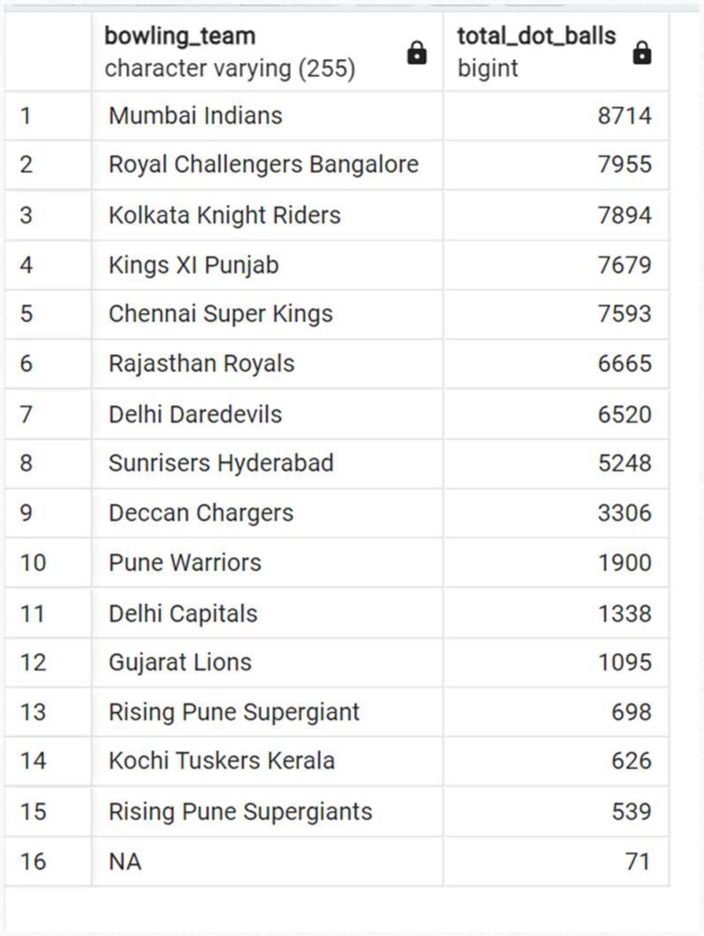


Boundaries by Team
Team Name
Kochi Tuskers Kerala 231
Rising Pune Supergiants 242
Rising Pune Supergiant 290
Gujarat Lions 624
Delhi Capitals 659
Pune Warriors 733
Deccan Chargers 1387
Sunrisers Hyderabad 2306
Delhi Daredevils 3022
Rajasthan Royals 3041
Chennai Super Kings 3496
Kolkata Knight Riders 3739
Kings XI Punjab 3780
Royal Challengers Bangalore 3800
Mumbai Indians 4118
0
1000
2000
3000
4000
5000
Bounaries count

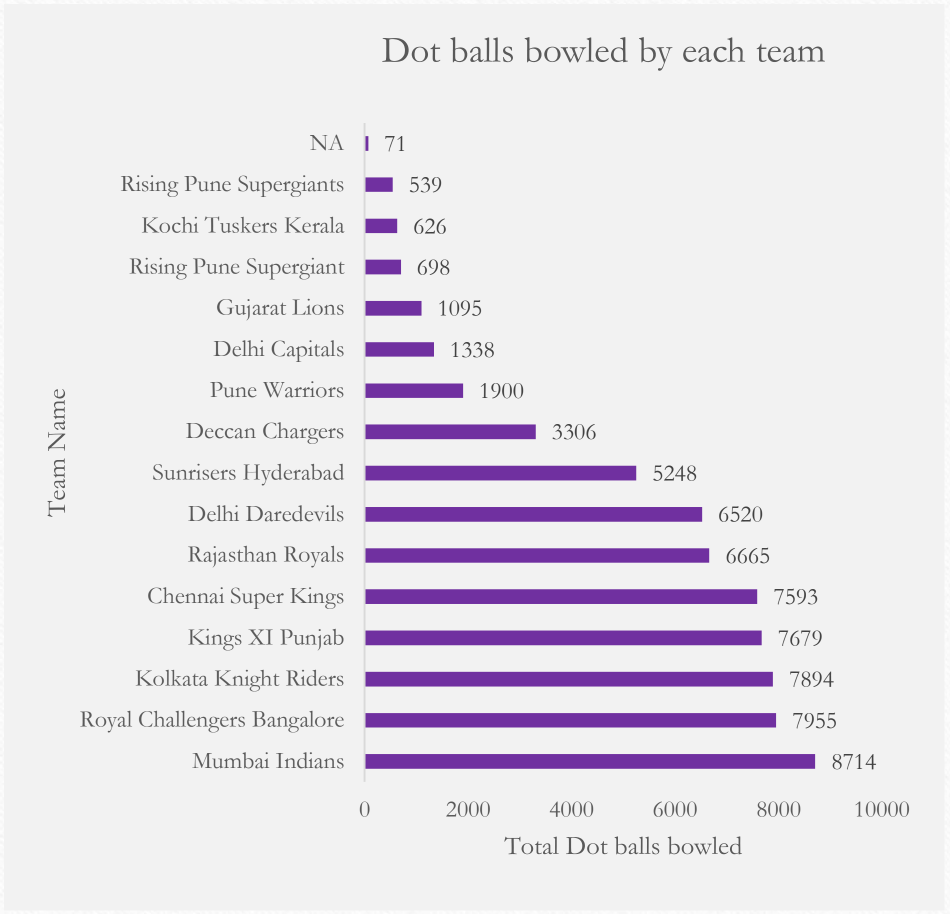
5. Write a query to fetch the total number of dot balls bowled by each team and order it in descending order of the total number of dot balls bowled.

Total number of dot balls bowled by each team

```
 select bowling_team, count(*) as total_dot_balls
from deliveries_v02
where ball_result = 'dot'
group by bowling_team
order by total_dot_balls desc;
```



	bowling_team character varying (255)	total_dot_balls bigint
1	Mumbai Indians	8714
2	Royal Challengers Bangalore	7955
3	Kolkata Knight Riders	7894
4	Kings XI Punjab	7679
5	Chennai Super Kings	7593
6	Rajasthan Royals	6665
7	Delhi Daredevils	6520
8	Sunrisers Hyderabad	5248
9	Deccan Chargers	3306
10	Pune Warriors	1900
11	Delhi Capitals	1338
12	Gujarat Lions	1095
13	Rising Pune Supergiant	698
14	Kochi Tuskers Kerala	626
15	Rising Pune Supergiants	539
16	NA	71

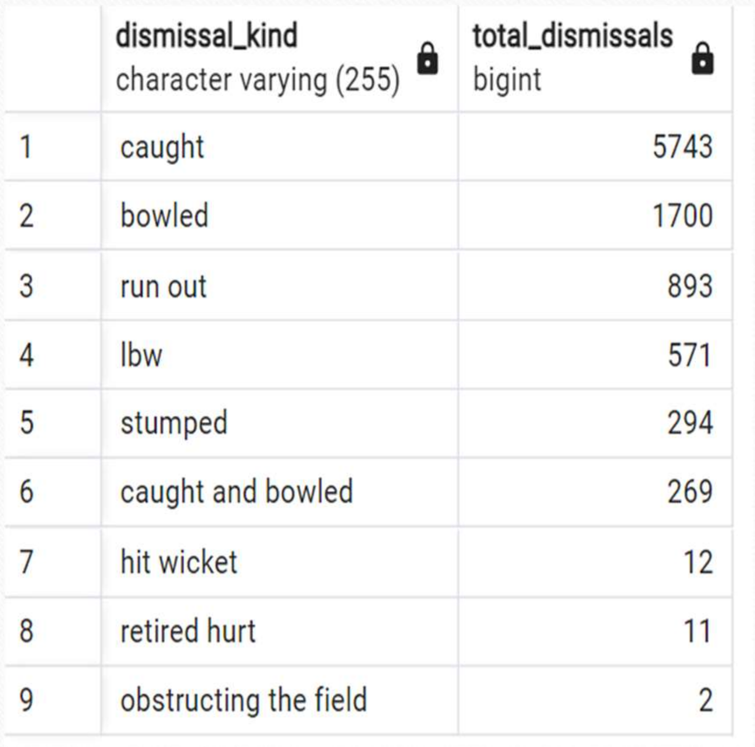


Dot balls bowled by each team
Team Name
NA 71
Rising Pune Supergiants 539
Kochi Tuskers Kerala 626
Rising Pune Supergiant 698
Gujarat Lions 1095
Delhi Capitals 1338
Pune Warriors 1900
Deccan Chargers 3306
Sunrisers Hyderabad 5248
Delhi Daredevils 6520
Rajasthan Royals 6665
Chennai Super Kings 7593
Kings XI Punjab 7679
Kolkata Knight Riders 7894
Royal Challengers Bangalore 7955
Mumbai Indians 8714
0
2000
4000
6000
8000
10000
Total Dot balls bowled

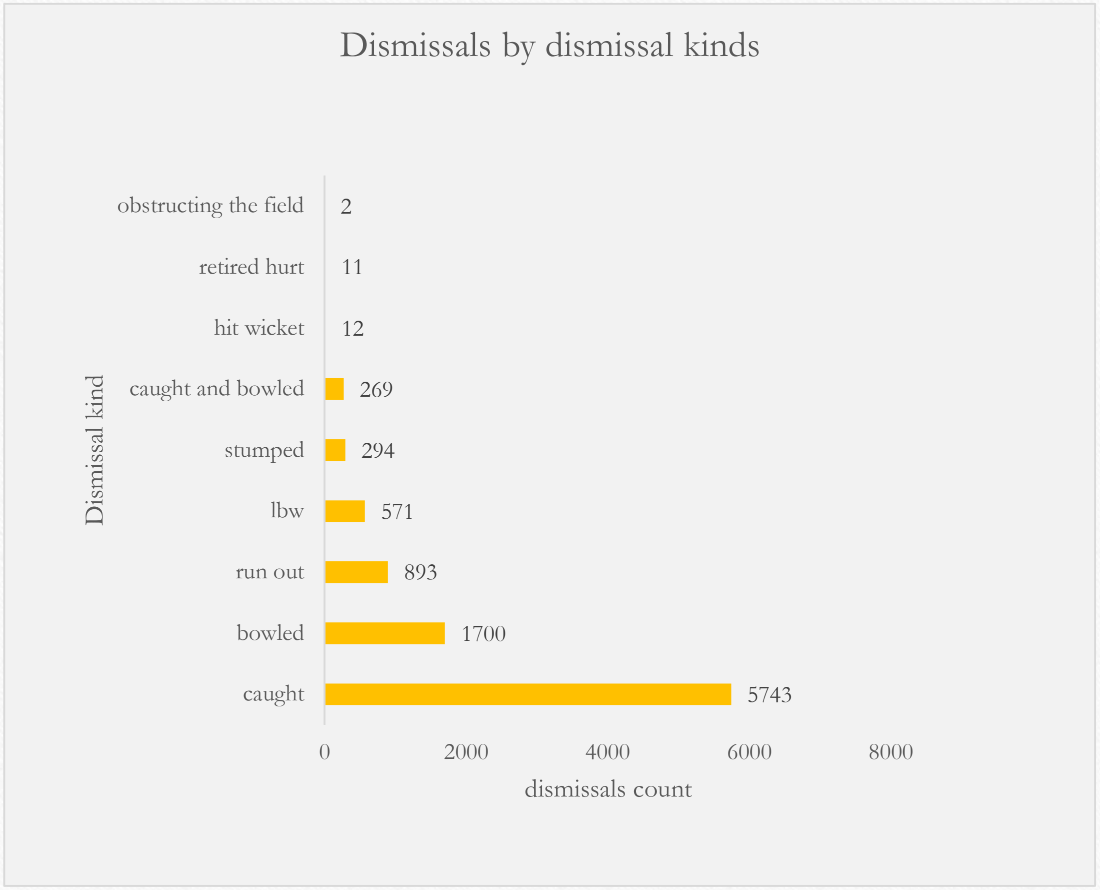
6. Write a query to fetch the total number of dismissals by dismissal kinds where dismissal kind is not NA

Total number of dismissals by dismissal kinds

```
 select dismissal_kind, count(*) as total_dismissals
from deliveries_v02
where dismissal_kind != 'NA'
group by dismissal_kind
order by total_dismissals desc;
```



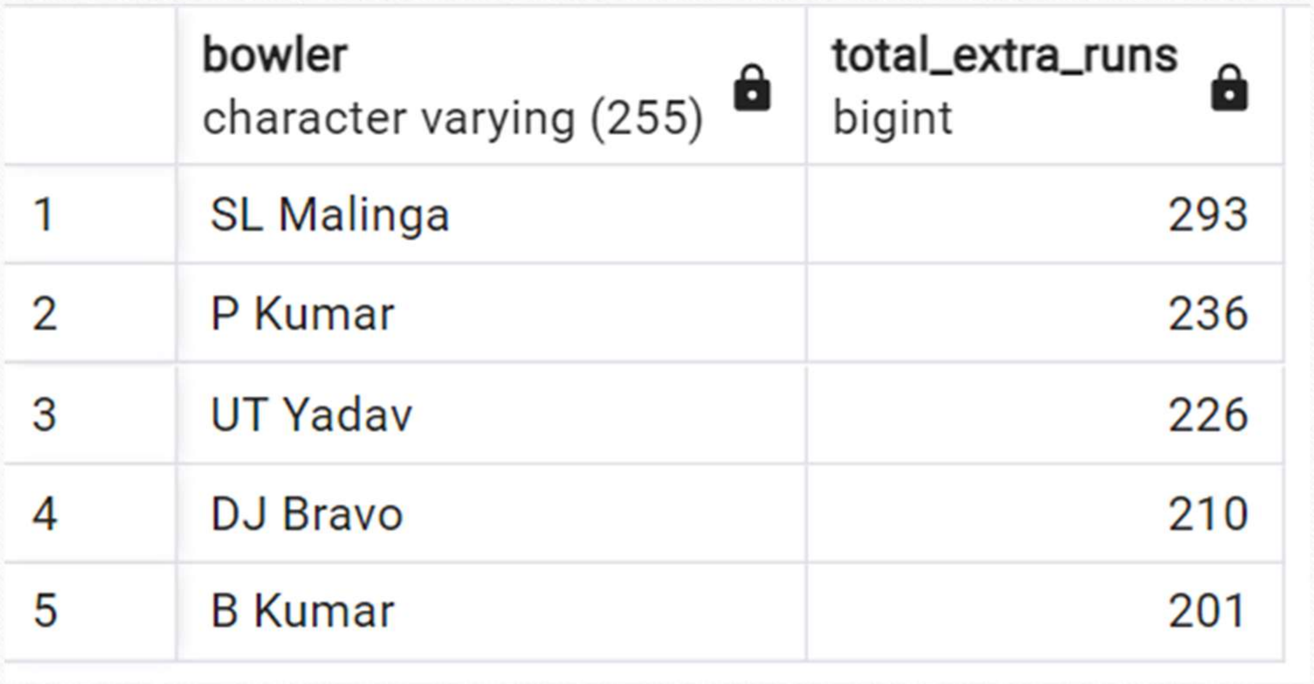
	dismissal_kind character varying (255)	total_dismissals bigint
1	caught	5743
2	bowled	1700
3	run out	893
4	lbw	571
5	stumped	294
6	caught and bowled	269
7	hit wicket	12
8	retired hurt	11
9	obstructing the field	2


Dismissals by dismissal kinds
Dismissal kind
obstructing the field
2
retired hurt
11
hit wicket
12
caught and bowled
269
stumped
294
lbw
571
run out
893
bowled
1700
caught
5743
0
2000
4000
6000
8000
dismissals count

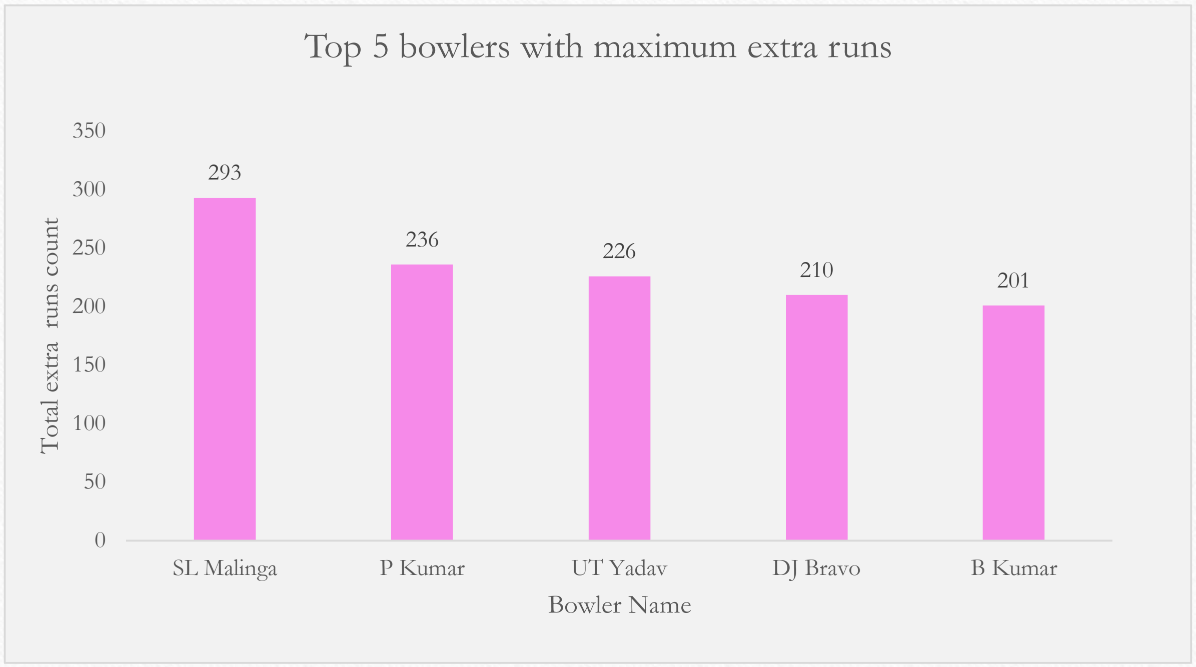
7. Write a query to get the top 5 bowlers who conceded maximum extra runs from the deliveries table

Top 5 bowlers who conceded maximum extra runs

```
select bowler, sum(extra_runs) as total_extra_runs
from deliveries_v02
group by bowler
order by total_extra_runs desc
limit 5;
```



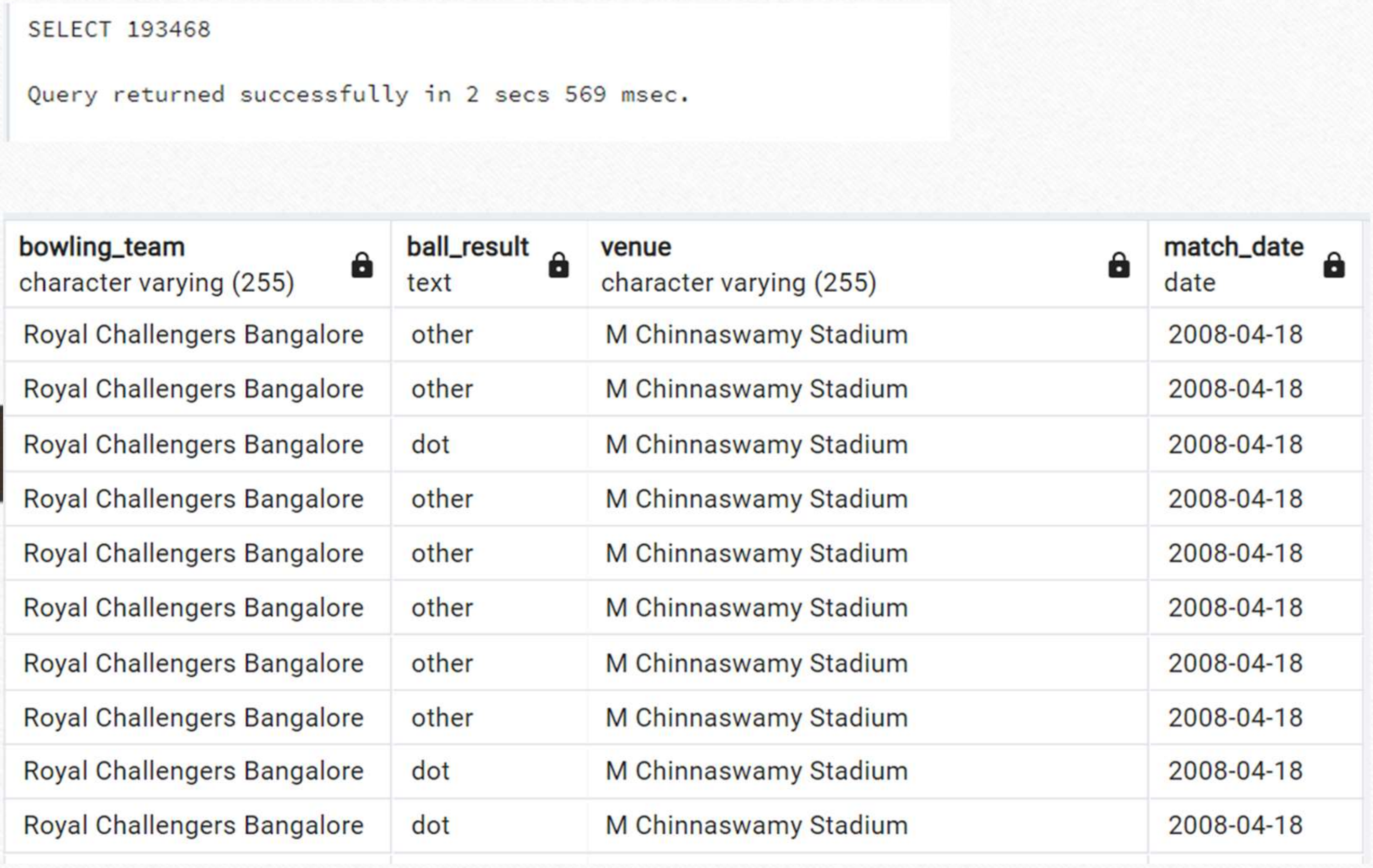
	bowler character varying (255)	total_extra_runs bigint
1	SL Malinga	293
2	P Kumar	236
3	UT Yadav	226
4	DJ Bravo	210
5	B Kumar	201


Top 5 bowlers with maximum extra runs
Total extra runs count
350
300
250
200
150
100
50
0
293
236
226
210
201
SL Malinga
P Kumar
UT Yadav
DJ Bravo
B Kumar
Bowler Name

8. Write a query to create a table named deliveries_v03 with all the columns of deliveries_v02 table and two additional column (named venue and match_date) of venue and date from table matches

create a table named deliveries_v03

```
 create table deliveries_v03 as
select d2.*, m.venue, m.date as match_date
from deliveries_v02 d2
join matches m on d2.id = m.id;
```



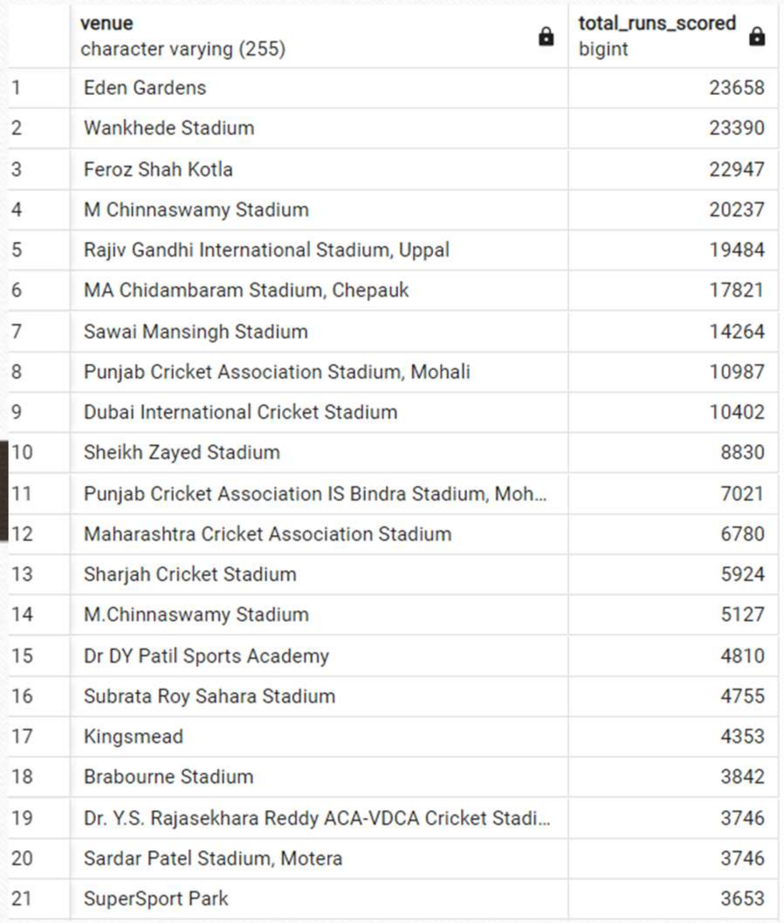
```
SELECT 193468

Query returned successfully in 2 secs 569 msec.
```

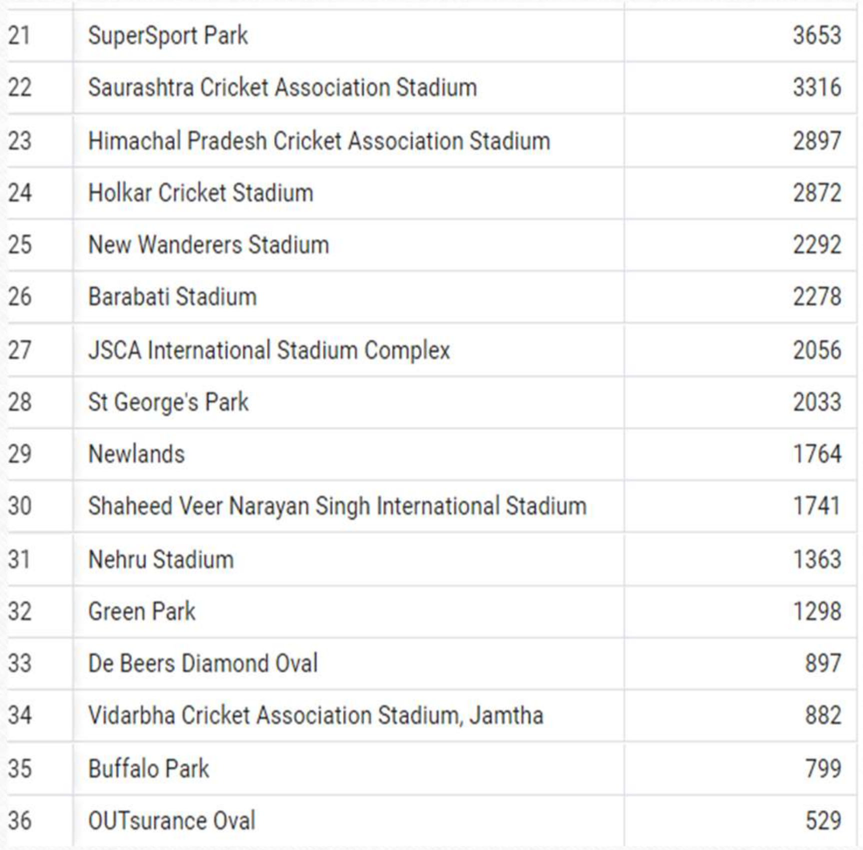
bowling_team character varying (255)	ball_result text	venue character varying (255)	match_date date
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	dot	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	other	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	dot	M Chinnaswamy Stadium	2008-04-18
Royal Challengers Bangalore	dot	M Chinnaswamy Stadium	2008-04-18

9. Write a query to fetch the total runs scored for each venue and order it in the descending order of total runs scored.

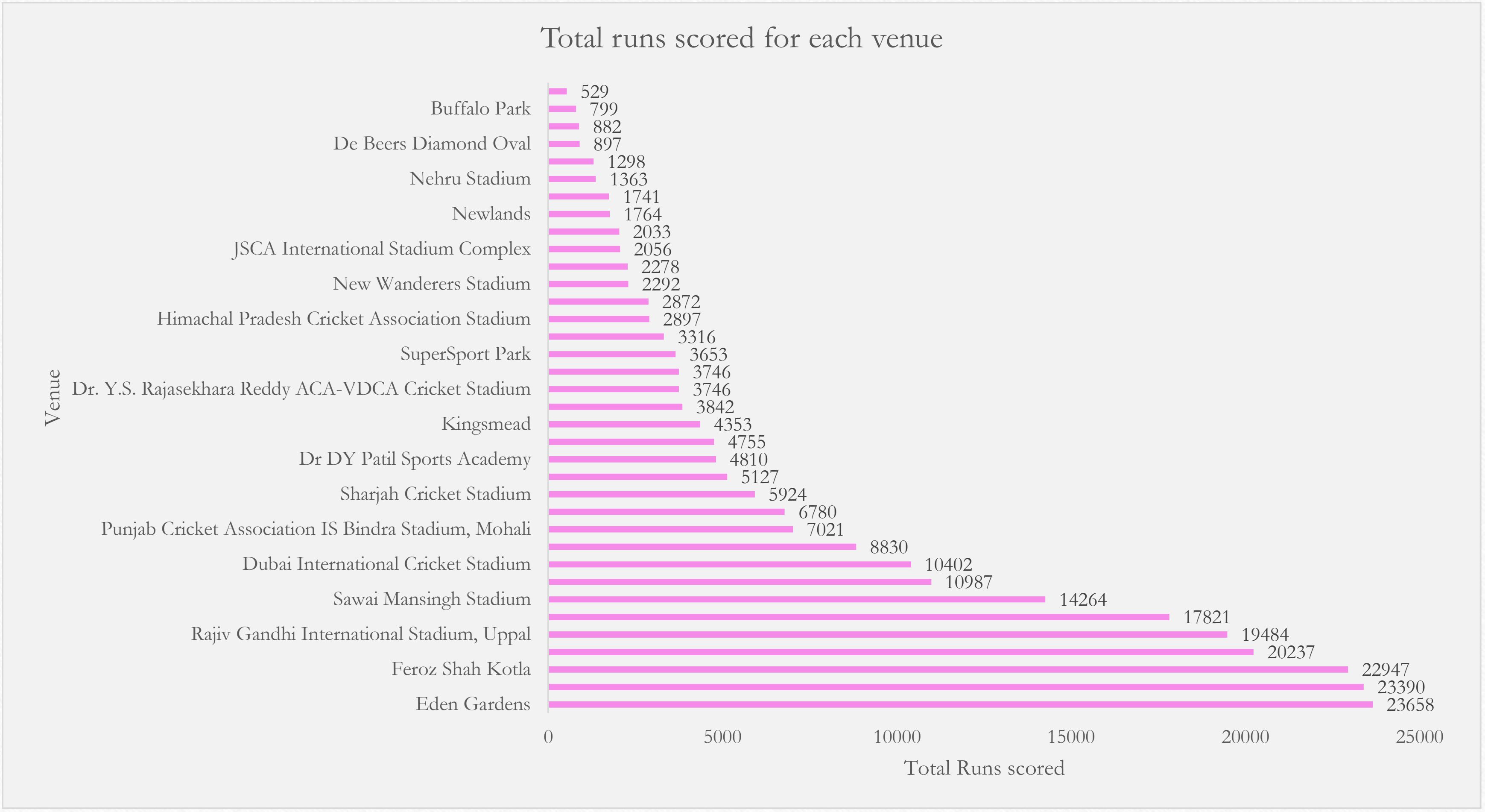
Total runs scored for each venue

```
select venue, sum(total_runs) as total_runs_scored
from deliveries_v03
group by venue
order by total_runs_scored desc;
```





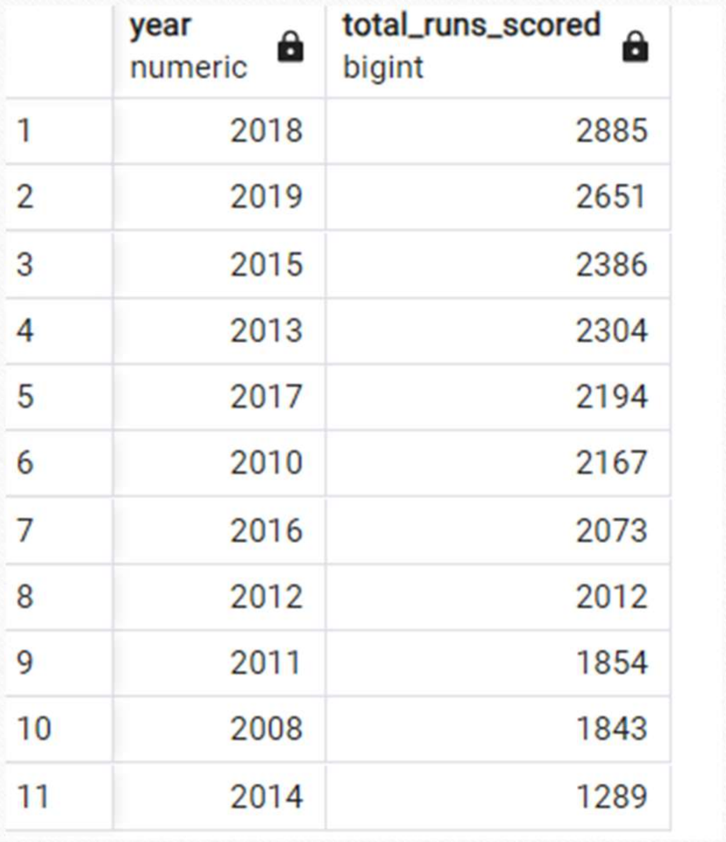
	venue character varying (255)	total_runs_scored bigint
1	Eden Gardens	23658
2	Wankhede Stadium	23390
3	Feroz Shah Kotla	22947
4	M Chinnaswamy Stadium	20237
5	Rajiv Gandhi International Stadium, Uppal	19484
6	MA Chidambaram Stadium, Chepauk	17821
7	Sawai Mansingh Stadium	14264
8	Punjab Cricket Association Stadium, Mohali	10987
9	Dubai International Cricket Stadium	10402
10	Sheikh Zayed Stadium	8830
11	Punjab Cricket Association IS Bindra Stadium, Moh...	7021
12	Maharashtra Cricket Association Stadium	6780
13	Sharjah Cricket Stadium	5924
14	M.Chinnaswamy Stadium	5127
15	Dr DY Patil Sports Academy	4810
16	Subrata Roy Sahara Stadium	4755
17	Kingsmead	4353
18	Brabourne Stadium	3842
19	Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadi...	3746
20	Sardar Patel Stadium, Motera	3746
21	SuperSport Park	3653
22	Saurashtra Cricket Association Stadium	3316
23	Himachal Pradesh Cricket Association Stadium	2897
24	Holkar Cricket Stadium	2872
25	New Wanderers Stadium	2292
26	Barabati Stadium	2278
27	JSCA International Stadium Complex	2056
28	St George's Park	2033
29	Newlands	1764
30	Shaheed Veer Narayan Singh International Stadium	1741
31	Nehru Stadium	1363
32	Green Park	1298
33	De Beers Diamond Oval	897
34	Vidarbha Cricket Association Stadium, Jamtha	882
35	Buffalo Park	799
36	OUTsurance Oval	529


Total runs scored for each venue
Venue
Buffalo Park
De Beers Diamond Oval
Nehru Stadium
Newlands
JSCA International Stadium Complex
New Wanderers Stadium
Himachal Pradesh Cricket Association Stadium
SuperSport Park
Dr. Y.S. Rajasekhara Reddy ACA-VDCA Cricket Stadium
Kingsmead
Dr DY Patil Sports Academy
Sharjah Cricket Stadium
Punjab Cricket Association IS Bindra Stadium, Mohali
Dubai International Cricket Stadium
Sawai Mansingh Stadium
Rajiv Gandhi International Stadium, Uppal
Feroz Shah Kotla
Eden Gardens
529
799
882
897
1298
1363
1741
1764
2033
2056
2278
2292
2872
2897
3316
3653
3746
3746
3842
4353
4755
4810
5127
5924
6780
7021
8830
10402
10987
14264
17821
19484
20237
22947
23390
23658
0
5000
10000
15000
20000
25000
Total Runs scored

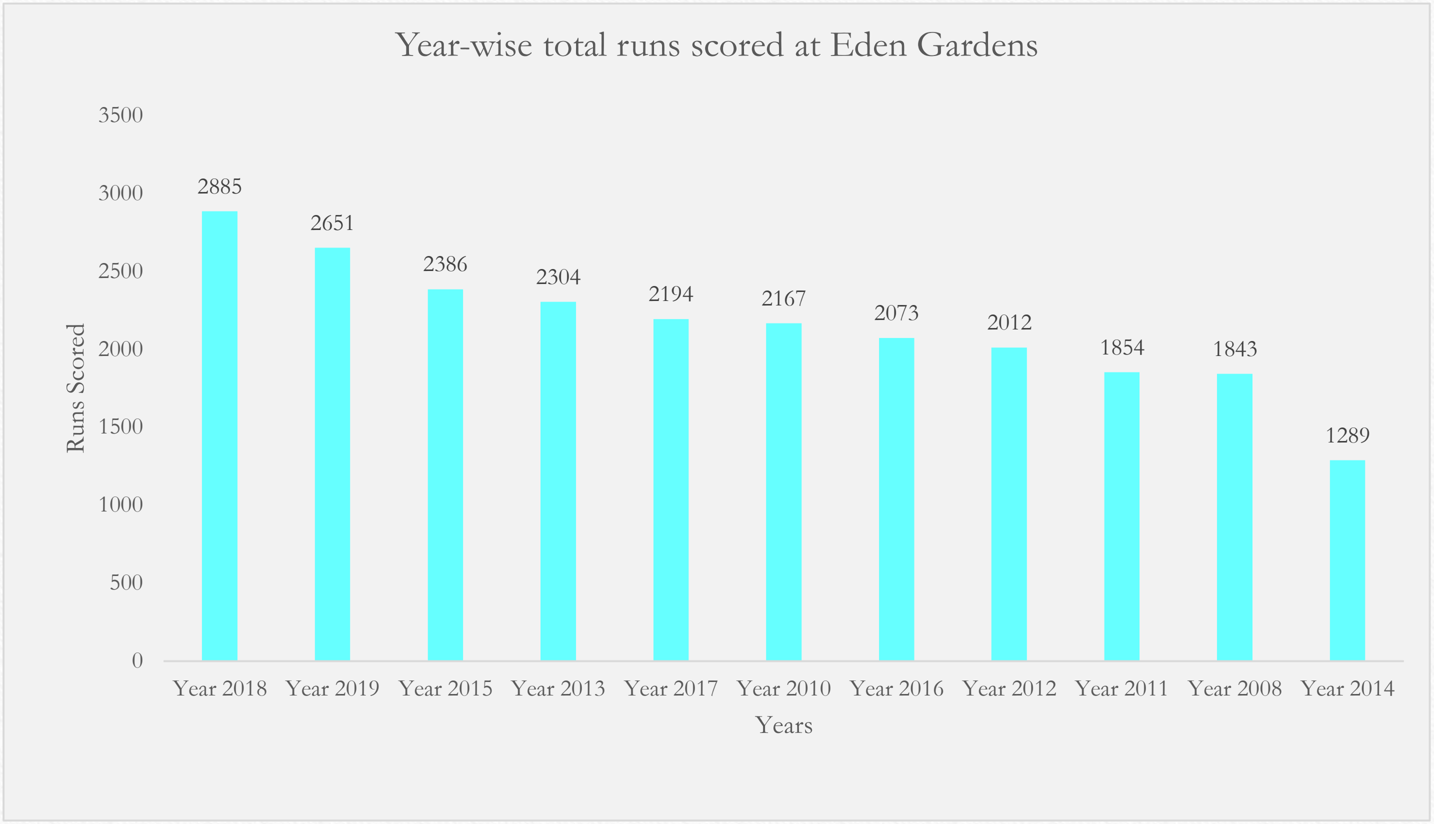
10. Write a query to fetch the year-wise total runs scored at Eden Gardens and order it in the descending order of total runs scored.

Year-wise total runs scored at Eden Gardens

```
select extract(year from match_date::date) as year,
       sum(total_runs) as total_runs_scored
from deliveries_v03
where venue = 'Eden Gardens'
group by year
order by total_runs_scored desc;
```



	year numeric	total_runs_scored bigint
1	2018	2885
2	2019	2651
3	2015	2386
4	2013	2304
5	2017	2194
6	2010	2167
7	2016	2073
8	2012	2012
9	2011	1854
10	2008	1843
11	2014	1289


Year-wise total runs scored at Eden Gardens
Runs Scored
3500
3000
2500
2000
1500
1000
500
0
2885
2651
2386
2304
2194
2167
2073
2012
1854
1843
1289
Year 2018
Year 2019
Year 2015
Year 2013
Year 2017
Year 2010
Year 2016
Year 2012
Year 2011
Year 2008
Year 2014
Years