IPL DATA CASE STUDY

```
Create database ipl;
use ipl;

CREATE TABLE matches (
    id INT PRIMARY KEY,
    season INT,
    city VARCHAR(50),
    date DATE,
    team1 VARCHAR(50),
    team2 VARCHAR(50),
    toss_winner VARCHAR(50),
    toss_decision VARCHAR(50),
    result VARCHAR(50),
    dl_applied INT,
    winner VARCHAR(50),
    win_by_runs INT,
    win_by_wickets INT,
    player_of_match VARCHAR(50),
    venue VARCHAR(100),
    umpire1 VARCHAR(50),
    umpire2 VARCHAR(50),
    umpire3 VARCHAR(50)
);
select * from matches;
CREATE TABLE deliveries(
   match_id         INTEGER  NOT NULL,
   inning           INTEGER  NOT NULL,
   batting_team     VARCHAR(50) NOT NULL,
   bowling_team     VARCHAR(50) NOT NULL,
   over_no          INTEGER  NOT NULL,
   ball             INTEGER  NOT NULL,
   batsman          VARCHAR(50) NOT NULL,
   non_striker      VARCHAR(50) NOT NULL,
   bowler           VARCHAR(50) NOT NULL,
   is_super_over_no BIT  NOT NULL,
   wide_runs        INTEGER  NOT NULL,
   bye_runs         INTEGER  NOT NULL,
   legbye_runs      INTEGER  NOT NULL,
   noball_runs      INTEGER  NOT NULL,
   penalty_runs     INTEGER  NOT NULL,
   batsman_runs     INTEGER  NOT NULL,
   extra_runs       INTEGER  NOT NULL,
   total_runs       INTEGER  NOT NULL,
   player_dismissed VARCHAR(50),
   dismissal_kind   VARCHAR(50),
   fielder          VARCHAR(50)
);
select * from deliveries;

#Q1
select player_of_match,count(*) as awards_count
from matches group by player_of_match
order by awards_count desc
limit 5;

#Q2
```

```
select season,winner as team,count(*) as matches_won
from matches group by season,winner;

#Q3

select avg(strike_rate) as average_strike_rate
from(
select batsman,(sum(total_runs)/count(ball))*100 as strike_rate
from deliveries group by batsman)as batsman_stats;

#Q4

select batting_first,count(*) as matches_won
from(
select case when win_by_runs>0 then team1
else team2
end as batting_first
from matches
where winner!="Tie") as batting_first_teams
group by batting_first;

#Q5

select batsman,(sum(batsman_runs)*100/count(*))
as strike_rate
from deliveries group by batsman
having sum(batsman_runs)>=200
order by strike_rate desc
limit 1;

#Q6

select batsman,count(*) as total_dismissals
from deliveries
where player_dismissed is not null
and bowler='SL Malinga'
group by batsman;

#Q7

select batsman,avg(case when batsman_runs=4 or batsman_runs=6
then 1 else 0 end)*100 as average_boundaries
from deliveries group by batsman;

#Q8
select season,batting_team,avg(fours+sixes) as average_boundaries
from(select season,match_id,batting_team,
sum(case when batsman_runs=4 then 1 else 0 end)as fours,
sum(case when batsman_runs=6 then 1 else 0 end) as sixes
from deliveries,matches
where deliveries.match_id=matches.id
group by season,match_id,batting_team) as team_bounsaries
group by season,batting_team;

#Q9
select season,batting_team,max(total_runs) as highest_partnership
from(select season,batting_team,partnership,sum(total_runs) as total_runs
from(select season,match_id,batting_team,over_no,
sum(batsman_runs) as partnership,sum(batsman_runs)+sum(extra_runs) as
total_runs
from deliveries,matches where deliveries.match_id=matches.id
group by season,match_id,batting_team,over_no) as team_scores
group by season,batting_team,partnership) as highest_partnership
group by season,batting_team;

#Q10
select m.id as match_no,d.bowling_team,
sum(d.extra_runs) as extras
from matches as m
join deliveries as d on d.match_id=m.id
where extra_runs>0
group by m.id,d.bowling_team;

#Q11

select m.id as match_no,d.bowler,count(*) as wickets_taken
from matches as m
join deliveries as d on d.match_id=m.id
where d.player_dismissed is not null
group by m.id,d.bowler
order by wickets_taken desc
limit 1;

#Q12

select m.city,case when m.team1=m.winner then m.team1
when m.team2=m.winner then m.team2
else 'draw'
end as winning_team,
count(*) as wins
from matches as m
join deliveries as d on d.match_id=m.id
where m.result!='Tie'
group by m.city,winning_team;

#Q13

select season,toss_winner,count(*) as toss_wins
from matches group by season,toss_winner;

#Q14
select player_of_match,count(*) as total_wins
from matches
where player_of_match is not null
group by player_of_match
order by total_wins desc;

#Q15

select m.id,d.inning,d.over_no,
avg(d.total_runs) as average_runs_per_over
from matches as m
join deliveries as d on d.match_id=m.id
group by m.id,d.inning,d.over_no;
```

```
#Q16
select m.season,m.id as match_no,d.batting_team,
sum(d.total_runs) as total_score
from matches as m
join deliveries as d on d.match_id=m.id
group by m.season,m.id,d.batting_team
order by total_score desc
limit 1;

#Q17
select m.season,m.id as match_no,d.batsman,
sum(d.batsman_runs) as total_runs
from matches as m
join deliveries as d on d.match_id=m.id
group by m.season,m.id,d.batsman
order by total_runs desc
limit 1;
```