**SQL Code For Various Analysis**

#### **1. Top 30 Bollers**

select Player\_Name, p.player\_id, sum(Bowling\_skill) as Skills from player p

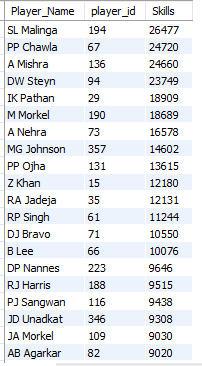
join ball\_by\_ball b on b.Bowler = p.Player\_Id

where b.Bowler = p.Player\_Id

group by Player\_Name, p.player\_id

order by Skills desc

limit 30;



#### **2. Top 35 Batsman**

select \* from (

select Player\_Name,

sum(Runs\_Scored) as total

from player p

join player\_match pm

on p.Player\_Id = pm.Player\_Id

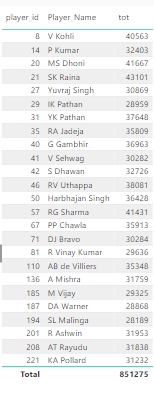
join batsman\_scored bs

on pm.Match\_Id = bs.Match\_Id

group by Player\_Name ) as A

order by total desc

limit 35;



#### **3. Top 5 Wicket-keepers**

select Player\_Name, count(Role\_Desc) AS Skill\_Points from player p

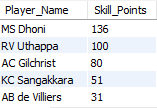
join player\_match pm on p.Player\_Id = pm.Player\_Id

join role r on r.role\_id = pm.role\_id

where r.role\_id = 2 or r.role\_id = 4

group by Player\_Name

order by Skill\_Points desc limit 5;



#### **4. Most Sixes by players**

select \* from

(select pm.match\_id, player\_name, count(runs\_scored) as total\_sixes

from batsman\_scored bs

join player\_match pm

on pm.match\_id = bs.match\_id

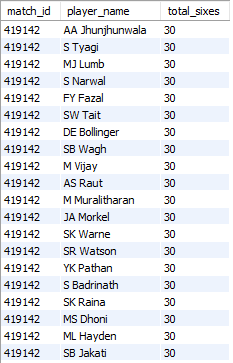
join player p

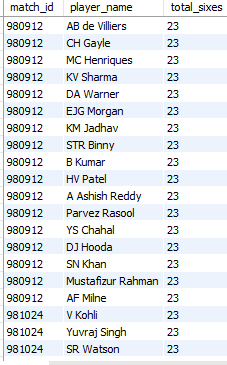
on p.Player\_id=pm.Player\_id

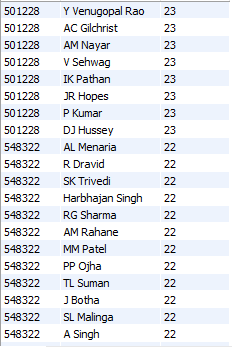
where Runs\_scored = 6

group by pm.match\_id, player\_name ) as A

order by total\_sixes desc;







#### **5. Most Fours by players**

select \* from (select pm.match\_id, player\_name, count(runs\_scored) as total\_four

from batsman\_scored bs

join player\_match pm

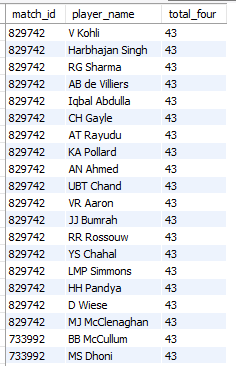
on pm.match\_id = bs.match\_id

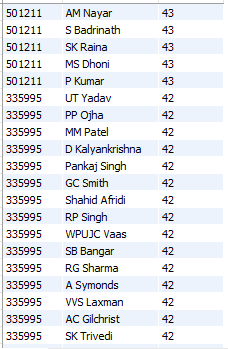
join player p on p.Player\_id=pm.Player\_id

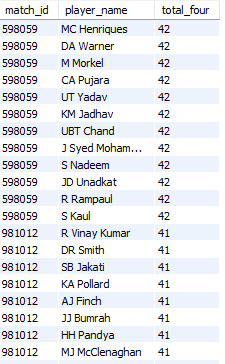
where Runs\_scored = 4

group by pm.match\_id, player\_name ) as A

order by total\_four desc;







#### **6. Win Margin:**

**SQL query :**

SELECT

Match\_ID,

Match\_Winner,

CASE

WHEN W.Win\_Type = 'runs' THEN M.Win\_Margin

WHEN W.Win\_Type = 'wickets' THEN M.Win\_Margin

ELSE NULL

END AS WinMargin

FROM

win\_by W

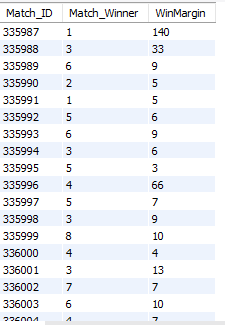
JOIN match\_ M

ON W.Win\_Id = M.Win\_Type

WHERE

W.Win\_Type IN ('Runs', 'Wickets');

* Query Gives : Win Margin



#### **7. Average Run Rate by Season:**

**SQL query :**

SELECT

Season\_Year ,

SUM(Runs\_Scored) / SUM(Over\_Id) AS AverageRunRate

FROM

batsman\_scored B

JOIN match\_ M

ON B.Match\_Id = M.Match\_Id

JOIN season S

ON M.Season\_Id = S.Season\_Id

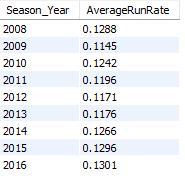
GROUP BY

Season\_Year

ORDER BY

Season\_Year;

* Query Gives : Average Run Rate by Season Year



#### **8. Win type** **:**

**SQL query :**

SELECT

Win\_Type,

COUNT(\*) AS DismissalCount

FROM

win\_by

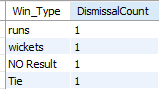
GROUP BY

Win\_Type

ORDER BY

DismissalCount DESC;

* Query Gives :Win Types



#### **9. Player Consistency:**

**SQL query :**

SELECT

Player\_Name,

Season\_Year,

AVG(Runs\_Scored) AS AverageRuns

FROM

player P

JOIN player\_match PM

ON P.Player\_Id = PM.Player\_Id

JOIN match\_ M

ON PM.Match\_Id = M.Match\_Id

JOIN season S

ON S.Season\_Id = M.Season\_Id

JOIN batsman\_scored BS

ON BS.Match\_Id = M.Match\_Id

GROUP BY

Player\_Name,

Season\_Year

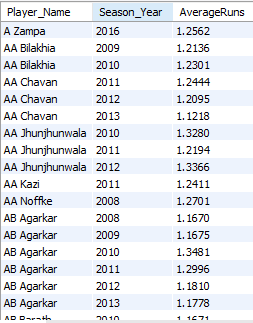
ORDER BY

Player\_Name,

Season\_Year;

* Query Gives :Player Consistency





#### **10. Performance by Venue:**

**SQL query :**

SELECT

Venue\_Name,

AVG(Runs\_Scored) AS AverageRuns

FROM

venue V

JOIN match\_ M

ON M.Venue\_Id = V.Venue\_Id

JOIN batsman\_scored B

ON M.Match\_Id = B.Match\_Id

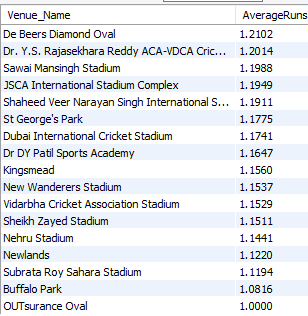
GROUP BY

Venue\_Name

ORDER BY

AverageRuns DESC;

* Query Gives :Performance by Venue





#### **11. Over Analysis:**

**SQL query :**

SELECT

W.Over\_Id,

COUNT(DISTINCT M.Match\_Id) AS TotalOvers,

SUM(Runs\_Scored) AS TotalRuns,

AVG(Runs\_Scored) AS AverageRunsPerOver

FROM

wicket\_taken W

JOIN match\_ M

ON W.Match\_Id = M.Match\_Id

JOIN batsman\_scored BS

ON BS.Match\_Id =M.Match\_Id

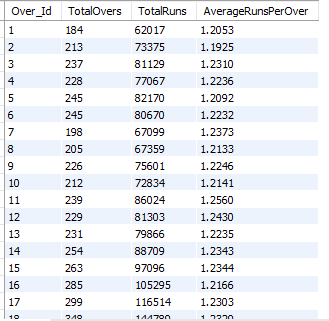
GROUP BY

W.Over\_Id

ORDER BY

W.Over\_Id;

* Query Gives : Over Analysis



#### **12. Bowler Types:**

**SQL query :**

SELECT

BS.Bowling\_skill , SUM(ER.Extra\_Runs) AS EXTRARUN ,

SUM(Runs\_Scored)

FROM

bowling\_style BS

JOIN extra\_runs ER

ON BS.Bowling\_Id = ER.Ball\_Id

JOIN batsman\_scored BS

ON BS.Ball\_Id = ER.Ball\_Id

;

* Query Gives : Bowler Types

#### **13. Performance in Final Matches:**

**SQL query :**

SELECT

Match\_Id , Innings\_No , SUM(Runs\_Scored) AS TOTAL\_RUN

FROM batsman\_scored

GROUP BY Match\_Id , Innings\_No

ORDER BY TOTAL\_RUN DESC;

* Query Gives : Performance in Finals

#### **14. Head-to-Head Records:**

**SQL query :**

SELECT

Team\_1 AS Team,

Team\_2 AS Opponent,

COUNT(\*) AS MatchesPlayed,

SUM(CASE WHEN Match\_Winner = Team\_1 THEN 1 ELSE 0 END) AS WinsForTeam,

SUM(CASE WHEN Match\_Winner = Team\_2 THEN 1 ELSE 0 END) AS WinsForOpponent

FROM

match\_

GROUP BY

Team\_1, Team\_2

UNION ALL

SELECT

Team\_2 AS Team,

Team\_1 AS Opponent,

COUNT(\*) AS MatchesPlayed,

SUM(CASE WHEN Match\_Winner = Team\_2 THEN 1 ELSE 0 END) AS WinsForTeam,

SUM(CASE WHEN Match\_Winner = Team\_1 THEN 1 ELSE 0 END) AS WinsForOpponent

FROM

match\_

GROUP BY

Team\_2, Team\_1;

* Query Gives : Head-to-Head Records

#### **15. Player Nationalities:**

**SQL query :**

SELECT

Country\_Name AS COUNTRY\_CODE,

COUNT(Player\_Id) AS NumberOfPlayers

FROM

player P

GROUP BY

Country\_Name

ORDER BY

NumberOfPlayers DESC;

* Query Gives : Player Nationalities

#### **16. Top Scores by Batting\_hand:**

**SQL query :**

SELECT

Batting\_hand,

Player\_Name,

Runs\_Scored,

Match\_Id

FROM

(SELECT

P.Batting\_hand,

P.Player\_Name,

BS.Runs\_Scored,

M.Match\_Id,

ROW\_NUMBER() OVER (PARTITION BY P.Batting\_hand ORDER BY BS.Runs\_Scored DESC) AS rank\_

FROM

player P

JOIN player\_match PM

ON P.Player\_Id = PM.Player\_Id

JOIN match\_ M

ON M.Match\_Id = PM.Match\_Id

JOIN batsman\_scored BS

ON M.Match\_Id = BS.Match\_Id

) ranked

WHERE

rank\_ = 1

ORDER BY Batting\_hand;

Query Gives : Top Scores by Batting\_hand

#### **17. Player Contributions:**

**SQL query :**

SELECT

P.Player\_Name,

T.Team\_Name,

SUM(BS.Runs\_Scored) AS TotalRuns,

SUM(Player\_Out) AS TotalWickets

FROM

player P

JOIN player\_match PM

ON P.Player\_Id = PM.Player\_Id

JOIN team T

ON T.Team\_Id = PM.Team\_Id

JOIN batsman\_scored BS

ON PM.Match\_Id = BS.Match\_Id

JOIN wicket\_taken W

ON W.Match\_Id = PM.Match\_Id

GROUP BY

P.Player\_Name,

T.Team\_Name

ORDER BY

TotalRuns DESC;

* Query Gives : Player Contributions

#### **18. Milestone Matches:**

**SQL query :**

WITH RankedMatches AS (

SELECT

P.Player\_Id,

Player\_Name,

M.Match\_Id,

Match\_Date,

ROW\_NUMBER() OVER (PARTITION BY Player\_Id ORDER BY Match\_Date) AS MatchNumber

FROM

player P

JOIN player\_match PM

ON P.Player\_Id = PM.Player\_Id

JOIN match\_ M

ON PM.Match\_Id = M.Match\_Id

)

SELECT

Player\_Id,

Player\_Name,

Match\_Id,

Match\_Date,

MatchNumber

FROM

RankedMatches

WHERE

MatchNumber IN (50, 100)

ORDER BY

Player\_Name, MatchNumber;

* Query Gives : Milestone Matches

#### **19. Win by Toss Decision:**

**SQL query :**

SELECT

TD.Toss\_Name,

COUNT(\*) AS TotalMatches,

SUM(CASE WHEN M.Match\_Winner = Toss\_Winner THEN 1 ELSE 0 END) AS WinsAfterWinningToss,

SUM(CASE WHEN M.Match\_Winner != Toss\_Winner THEN 1 ELSE 0 END) AS LossesAfterWinningToss,

SUM(CASE WHEN M.Match\_Winner = Toss\_Winner THEN 1 ELSE 0 END) \* 100.0 / COUNT(\*) AS WinPercentage

FROM

toss\_decision TD

JOIN match\_ M

ON TD.Toss\_Id = M.Toss\_Decide

GROUP BY

TD.Toss\_Name

ORDER BY

TD.Toss\_Name;

* Query Gives : Win by Toss Decision

#### **20. Match Locations:**

**SQL query :**

SELECT

Venue\_Name,

City\_Name,

COUNT(\*) AS TotalMatches,

COUNT(DISTINCT Season\_Year) AS SeasonsHosted,

COUNT(DISTINCT Team\_1) + COUNT(DISTINCT Team\_2) AS TeamsPlayed,

COUNT(DISTINCT Match\_Winner) AS UniqueWinners

FROM

venue V

JOIN city C

ON V.City\_Id = C.City\_Id

JOIN match\_ M

ON M.Venue\_Id = V.Venue\_Id

JOIN season S

ON M.Season\_Id = S.Season\_Id

GROUP BY

Venue\_Name,

City\_Name

ORDER BY

TotalMatches DESC;

* Query Gives : Match Locations

#### **21. Overseas Player Performances:**

**SQL query :**

SELECT

P.Player\_Name,

C.Country\_Name,

SUM(Runs\_Scored) AS TotalRuns,

SUM(Player\_Out) AS TotalWickets,

COUNT(M.Match\_Id) AS MatchesPlayed

FROM

player P

JOIN country C

ON P.Country\_Name = C.Country\_Id

JOIN player\_match PM

ON P.Player\_Id = PM.Player\_Id

JOIN match\_ M

ON M.Match\_Id = PM.Match\_Id

JOIN batsman\_scored BS

ON M.Match\_Id = BS.Match\_Id

JOIN wicket\_taken WT

ON M.Match\_Id = WT.Match\_Id

WHERE

C.Country\_Name != 'India'

GROUP BY

P.Player\_Name,

C.Country\_Name

ORDER BY

TotalRuns DESC, TotalWickets DESC;

* Query Gives : Overseas Player Performances

#### **22. Run Rate in Powerplay:**

**SQL query :**

SELECT

M.Match\_Id,

WT.Innings\_No,

SUM(Runs\_Scored) AS TotalRuns,

COUNT(WT.Ball\_Id) AS TotalBalls,

ROUND(SUM(Runs\_Scored) \* 6.0 / COUNT(WT.Ball\_Id), 2) AS PowerplayRunRate

FROM

match\_ M

JOIN wicket\_taken WT

ON M.Match\_Id = WT.Match\_Id

JOIN batsman\_scored BS

ON M.Match\_Id = BS.Match\_Id

WHERE

WT.Over\_Id BETWEEN 1 AND 6

GROUP BY

M.Match\_Id,

Innings\_No

ORDER BY

M.Match\_Id,

Innings\_No;

* Query Gives : Run Rate in Powerplay

#### **23. Death Over Analysis:**

**SQL query :**

SELECT

M.Match\_Id,

WT.Innings\_No,

SUM(Runs\_Scored) AS TotalRuns,

COUNT(WT.Ball\_Id) AS TotalBalls,

ROUND(SUM(Runs\_Scored) \* 6.0 / COUNT(Runs\_Scored), 2) AS DeathOverRunRate

FROM

match\_ M

JOIN wicket\_taken WT

ON M.Match\_Id = WT.Match\_Id

JOIN batsman\_scored BS

ON M.Match\_Id = BS.Match\_Id

WHERE

WT.Over\_Id BETWEEN 17 AND 20

GROUP BY

M.Match\_Id,

WT.Innings\_No

ORDER BY

M.Match\_Id,

WT.Innings\_No;

* Query Gives : Death Over Analysis

#### **24. Player Transfers:**

**SQL query :**

SELECT

Player\_Name,

Season\_Year,

Team\_1 AS From\_team,

Team\_2 AS To\_team

FROM

player P

JOIN

player\_match PM ON P.Player\_Id = PM.Player\_Id

JOIN match\_ M

ON PM.Match\_Id = M.Match\_Id

JOIN season S

ON S.Season\_Id = M.Season\_Id

ORDER BY

Season\_Year, Player\_Name;

* Query Gives :Player Transfers