



Database Management Systems Lab

Project Phase: 2

Group Members:

Sr.No.	Name	PRN	Branch	Division
1	Hemang Dave	21070122041	CSE	A2
2	Dishank Arora	21070122046	CSE	A2

Title of the project: IPL

About the project:

In the electrifying world of cricket, few spectacles rival the Indian Premier League (IPL) for its fusion of sporting brilliance, strategic acumen, and fervent fanfare. As each IPL season unfolds, it bequeaths a treasure trove of data, a digital mosaic that captures the heartbeats of players, the dynamics of teams, and the narratives of matches. In this dynamic landscape, a formidable challenge emerges – to distil this influx of information into actionable insights that illuminate the game's intricacies and guide decisions across the cricketing spectrum.

This voyage of data exploration embarks on a journey that traverses teams, players, matches, and the unseen forces that shape the destiny of cricketing battles. It is an odyssey through databases replete with team profiles, player statistics, match outcomes, and the very words that breathe life into the games – the commentaries. Through careful analysis, this endeavours to unlock the strategic nuances that lie beneath the surface, providing a compass for teams, players, broadcasters, and ardent fans alike.

At its core, this exploration scrutinizes the essence of IPL cricket – the teams. It delves into the annals of team performances across seasons, deciphering the codes of wins, losses, and the patterns that underlie triumphs. Beyond the collective, it profiles individual players, blending their skills, roles, and on-field prowess into a vivid canvas. In parallel, it uncovers the cryptic connections between venues and match outcomes, revealing the hidden stories that pitch conditions and enthusiastic crowds narrate.

This odyssey also navigates the enigmatic world of match officials, dissecting their influence on the course of the game and their role in shaping contentious moments. It contemplates the epic sagas of tournaments – their inception, evolution, and the rhythms that dictate their ebb



and flow. The melding of match commentaries with pivotal events elucidates the magic moments that reshape destinies within the span of an over.

In this symphony of data, rankings such as the Orange Cap and Purple Cap emerge as accolades that crown exceptional batting and bowling prowess, across seasons. Sponsorship connections are analysed to understand their potential impact on team dynamics and their resonance with the fans. The figureheads of coaches, too, command attention as the analysis dissects their influence on strategy and performance.

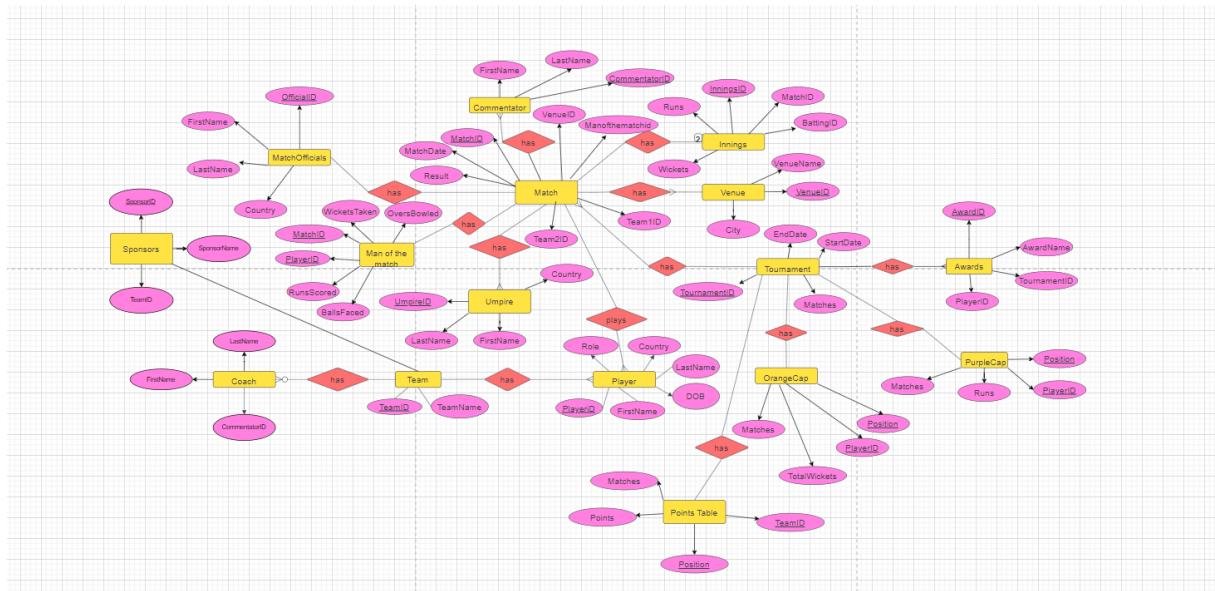
But this journey doesn't end at the surface – it plunges deeper into the intricate ecosystem. The points table becomes a testament to the rhythm of accumulation and the interplay of victories and defeats. It navigates the labyrinthine threads that link tournaments and their individual matches, revealing the hidden narratives that breathe life into the contests. It peels back layers to uncover the pulse of fans – the interplay between match outcomes, player feats, and the resonant metrics of fan engagement.

Intricacies extend further to encompass player workloads, the ebb and flow of form, and the tantalizing potential of predictive model. This voyage stands as a testament to the symbiosis of data and insight, a chronicle of how numbers fuse with narratives to forge a deeper understanding of the IPL's vibrant tapestry. In essence, this endeavour seeks not just to uncover facts, but to unearth the very soul of cricket as played in the IPL arena.

Addressed pain points:

In your sports project, points are a crucial component used for ranking and quantifying team and player performance. A dedicated PointsTable is employed to track these points, with each team's performance in matches contributing to their points tally. The points system drives the competitive spirit within the tournament and is vital for determining awards like "Player of the Match" and statistical accolades such as the "Orange Cap" for the highest run-scorer. Automated stored procedures or triggers likely manage the calculation and update of points in real-time, ensuring accurate rankings and data analysis. In summary, your sports project's points system forms the foundation for fair competition, ranking, and data-driven insights.

ER diagram:



Padma Shri by President of India)

Anomalies:

A database anomaly is a fault within a database, which can occur because of poor planning or when everything is stored in a flat database. These anomalies naturally occur and result in data that does not match the real-world the database purports to represent.

Anomalies occurring in our code:

Venues:

- Consider adding attributes like city, state, or country to provide more information about the venue.

Awards Table:

- Consider adding a date or season attribute to the Awards table. This would provide context for when the award was given.

Functional dependencies:

Functional dependency refers to a relationship that is present between attributes of any table that are dependent on each other. It helps in avoiding data redundancy and getting to know more about bad designs.

Functional dependencies of schema occurring in our code:

1. *Teams*:



- TeamID → TeamName

—TeamName → TeamID

2. *Players*:

- PlayerID → FirstName, LastName, DateOfBirth, Country, Role, TeamID
- FirstName, LastName → PlayerID
- DateOfBirth, Country, Role, TeamID → PlayerID

3. *Coaches*:

- CoachID → CoachName, TeamID, Specialization
- CoachName, TeamID, Specialization → CoachID

4. *Venues*:

- VenueID → VenueName, Stadium, State
- VenueName, Stadium, State → VenueID

5. *Matches*:

- MatchID → MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID
- MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID → MatchID

6. *Umpires*:

- UmpireID → FirstName, LastName, Country
- FirstName, LastName, Country → UmpireID

7. *MatchOfficials*:

- OfficialID → FirstName, LastName, Country
- FirstName, LastName, Country → OfficialID

8. *Tournaments*:

- TournamentID → TournamentName, StartDate, EndDate, WinnerID
- TournamentName, StartDate, EndDate, WinnerID → TournamentID

9. *TournamentMatches*:

- TournamentID, MatchID → (No non-trivial FDs as the combination of TournamentID and MatchID forms the primary key)

10. *Commentator*:

- CommentatorID → FirstName, LastName
- FirstName, LastName → CommentatorID

11. *Innings*:

- InningsID → MatchID, BattingTeamID, BowlingTeamID, TotalRuns, TotalWickets
- MatchID, BattingTeamID, BowlingTeamID, TotalRuns, TotalWickets → InningsID

12. *ManOfTheMatch*:



- MatchID, PlayerID → RunsScored, BallsFaced, WicketsTaken, OversBowled
—~~RunsScored, BallsFaced, WicketsTaken, OversBowled~~ → MatchID, PlayerID

13. *OrangeCap*:

- Position, PlayerID → Runs, NumberOfMatches
- Runs, NumberOfMatches → Position, PlayerID

14. *PurpleCap*:

- Position, PlayerID → Wickets, NumberOfMatches
- Wickets, NumberOfMatches → Position, PlayerID

15. *Sponsors*:

- SponsorID → SponsorName, TeamID
- SponsorName, TeamID → SponsorID

16. *PointsTable*:

- Position, TeamID → NumberOfMatches, Points
- NumberOfMatches, Points → Position, TeamID

17. *Awards*:

- TournamentID, AwardID → AwardName, PlayerID
- AwardName, PlayerID → TournamentID, AwardID

Normalization

A normalization procedure, which combines and splits tables, is usually sufficient to remove this. By normalizing the database, we reduce the likelihood of creating tables that generate anomalies. Normalization is necessary if you do not do it then the overall integrity of the data stored in the database will eventually degrade. Specifically, this is due to data anomalies.

Types of normalization

1. Unnormalized form or UNF:

It is the simplest database model also known as non-first normal form (NF2). A UNF model will suffer problems like data redundancy thus it lacks the efficiency of database normalization.

Padma Shri by President of India)

2. First Normal Form or 1NF:

A relation is in first normal form only if the relational table doesn't contain any multivalued attribute but contains only single-valued attributes.

3. Second Normal Form or 2NF:

A relation is in second normal form if:

- It is in first normal form or 1NF



Founder: Prof. Dr. S. B. Mujumdar, M.Sc.,Ph.D. (Awarded Padma Bhushan and

- It doesn't contain any partial dependencies. (It shouldn't have any non-prime attribute which is functionally dependent on any proper subset of the candidate key of the relation.).

4. Third Normal Form or 3NF:

Let R be the relational schema, $X \rightarrow Y$ any non-trivial functional dependency over R is in 3NF if:

- R should be in 2NF
- X should be candidate key or superkey, or
- Y should be prime attribute

Normalization occurring in our code:

- Attributes: VenueID, VenueName, Stadium, City, State,Country

Key: {VenueID}

Normalization: This is already in at least 1NF. It is also in 2NF because there are no partial dependencies.

Conclusion: Already in at least 2NF

The screenshot shows the MySQL Workbench interface with the 'venues' table selected. The table has columns: VenueID, VenueName, Stadium, and State. The data is as follows:

VenueID	VenueName	Stadium	State
1	Wankhede Stadium	Mumbai	Maharashtra
2	Eden Gardens	Kolkata	West Bengal
3	M. Chinnaswamy Stadium	Bengaluru	Karnataka
4	Arun Jaitley Stadium	Delhi	Delhi
5	Rajiv Gandhi International Cricket Stadium	Hyderabad	Andhra Pradesh

The SQL editor shows the creation of the 'venues' table with an additional 'State' column and a CASE statement for the 'State' column:

```

CREATE TABLE `venues` (
  `VenueID` int(11) NOT NULL,
  `VenueName` varchar(255) NOT NULL,
  `Stadium` varchar(255) NOT NULL,
  `State` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
ALTER TABLE `venues` ADD PRIMARY KEY (`VenueID`);
ALTER TABLE `venues` ADD COLUMN `State` varchar(255);
-- Update the "State" column with data for multiple venues in a single statement
UPDATE `venues` SET `State` = CASE
    WHEN `VenueID` = 1 THEN 'Maharashtra'
    WHEN `VenueID` = 2 THEN 'West Bengal'
    WHEN `VenueID` = 3 THEN 'Karnataka'
    WHEN `VenueID` = 4 THEN 'Delhi'
    WHEN `VenueID` = 5 THEN 'Andhra Pradesh'
    WHEN `VenueID` = 6 THEN 'Rajasthan'
    WHEN `VenueID` = 7 THEN 'Madhya Pradesh'
    WHEN `VenueID` = 8 THEN 'Tamilnadu'
END;
    
```

2. Awards:

Attributes: AwardID, AwardName, DateReceived, PlayerID, MatchID

Key: {AwardID} Normalization: This is already in at least 1NF. It's also in 2NF because there are no partial dependencies.



Conclusion: Already in at least 2NF

Thus, we have normalized this table.

Padma Shri by President of India)

What is MySQL:

MySQL is a relational database management system (RDBMS) developed by Oracle that is based on structured query language (SQL).

Implementation of the tables in MySQL:

show databases;

USE IPL;

-- Create the Teams table

```
CREATE TABLE Teams (
    TeamID INT PRIMARY KEY,
    TeamName VARCHAR(255) NOT NULL
);
```

-- Create the Players table

```
CREATE TABLE Players (
    PlayerID INT PRIMARY KEY,
    FirstName VARCHAR(255) NOT NULL,
    LastName VARCHAR(255) NOT NULL,
    DateOfBirth DATE,
    Country VARCHAR(255),
    Role VARCHAR(255),
    TeamID INT,
    FOREIGN KEY (TeamID) REFERENCES Teams (TeamID)
);
```

-- Create the Coaches table

```
CREATE TABLE Coaches (
    CoachID INT PRIMARY KEY,
    CoachName VARCHAR(255) NOT NULL,
```



TeamID INT,
Specialization VARCHAR(255),
FOREIGN KEY (TeamID) REFERENCES Teams (TeamID)
);

-- Create the Venues table
CREATE TABLE Venues (
VenueID INT PRIMARY KEY,
VenueName VARCHAR(255) NOT NULL,
Stadium varchar(255)

);

-- Create the Matches table
CREATE TABLE Matches (
MatchID INT PRIMARY KEY,
MatchDate DATE NOT NULL,
VenueID INT,
Team1ID INT,
Team2ID INT,
Result VARCHAR(255),
ManOfTheMatchID INT,
FOREIGN KEY (VenueID) REFERENCES Venues (VenueID),
FOREIGN KEY (Team1ID) REFERENCES Teams (TeamID),
FOREIGN KEY (Team2ID) REFERENCES Teams (TeamID),
FOREIGN KEY (ManOfTheMatchID) REFERENCES Players (PlayerID)
);

-- Create the Umpires table
CREATE TABLE Umpires (
UmpireID INT PRIMARY KEY,
FirstName VARCHAR(255) NOT NULL,
LastName VARCHAR(255) NOT NULL,
Country VARCHAR(255)
);

-- Create the MatchOfficials table
CREATE TABLE MatchOfficials (
OfficialID INT PRIMARY KEY,
FirstName VARCHAR(255) NOT NULL,
LastName VARCHAR(255) NOT NULL,
Country VARCHAR(255)
);

-- Create the Tournaments table



```
CREATE TABLE Tournaments (
    TournamentID INT PRIMARY KEY,
    TournamentName VARCHAR(255) NOT NULL,
    StartDate DATE NOT NULL,
    EndDate DATE NOT NULL
);

-- Create the TournamentMatches table
CREATE TABLE TournamentMatches (
    TournamentID INT,
    MatchID INT,
    PRIMARY KEY (TournamentID, MatchID),
    FOREIGN KEY (TournamentID) REFERENCES Tournaments (TournamentID),
    FOREIGN KEY (MatchID) REFERENCES Matches (MatchID)
);

-- Create the Commentaries table
CREATE TABLE Commentator (
    CommentatorID INT PRIMARY KEY,
    MatchID INT,
    FirstName Varchar(255),
    LastName Varchar(255),
    FOREIGN KEY (MatchID) REFERENCES Matches (MatchID)
);

-- Create the Innings table
CREATE TABLE Innings (
    InningsID INT PRIMARY KEY,
    MatchID INT,
    BattingTeamID INT,
    BowlingTeamID INT,
    TotalRuns INT,
    TotalWickets INT,
    FOREIGN KEY (MatchID) REFERENCES Matches (MatchID),
    FOREIGN KEY (BattingTeamID) REFERENCES Teams (TeamID),
    FOREIGN KEY (BowlingTeamID) REFERENCES Teams (TeamID)
);

-- Create the ManOfTheMatch table
CREATE TABLE ManOfTheMatch (
    MatchId INT,
    PlayerID INT,
    RunsScored INT,
    BallsFaced INT,
```



WicketsTaken INT,
OversBowled DECIMAL(4, 2),
PRIMARY KEY (MatchID, PlayerID),
FOREIGN KEY (MatchID) REFERENCES Matches (MatchID),
FOREIGN KEY (PlayerID) REFERENCES Players (PlayerID)
);

-- Create the OrangeCap table
CREATE TABLE OrangeCap (
Position INT,
PlayerID INT,
Runs INT,
NumberOfMatches INT,
PRIMARY KEY (Position, PlayerID),
FOREIGN KEY (PlayerID) REFERENCES Players (PlayerID)
);

-- Create the PurpleCap table
CREATE TABLE PurpleCap (
Position INT,
PlayerID INT,
Wickets INT,
NumberOfMatches INT,
PRIMARY KEY (Position, PlayerID),
FOREIGN KEY (PlayerID) REFERENCES Players (PlayerID)
);

-- Create the Sponsors table
CREATE TABLE Sponsors (
SponsorID INT PRIMARY KEY,
SponsorName VARCHAR(255) NOT NULL,
TeamID INT,
FOREIGN KEY (TeamID) REFERENCES Teams (TeamID)
);

-- Create the PointsTable table
CREATE TABLE PointsTable (
Position INT,
TeamID INT,
NumberOfMatches INT,
Points INT,
PRIMARY KEY (Position, TeamID),
FOREIGN KEY (TeamID) REFERENCES Teams (TeamID)
);



-- Create the Awards table

CREATE TABLE Awards (

```

AwardID INT PRIMARY KEY,
AwardName VARCHAR(255) NOT NULL,
PlayerID INT,
MatchID INT,
FOREIGN KEY (PlayerID) REFERENCES Players (PlayerID),
FOREIGN KEY (MatchID) REFERENCES Matches (MatchID)
);
```

show tables;

INSERT INTO Teams (TeamID, TeamName) VALUES

```

(1, 'Mumbai Indians'),
(2, 'Chennai Super Kings'),
(3, 'Kolkata Knight Riders'),
(4, 'Delhi Capitals'),
(5, 'Royal Challengers Bangalore'),
(6, 'Kings XI Punjab'),
(7, 'Sunrisers Hyderabad'),
(8, 'Rajasthan Royals');
```

INSERT INTO Players

```

VALUES
('1','Virat', 'Kohli', '1988-11-05', 'India', 'Batsman', 5),
('2','Rohit', 'Sharma', '1987-04-30', 'India', 'Batsman', 1),
('3','Kane', 'Williamson', '1990-08-08', 'New Zealand', 'Batsman', 7),
('4','MS', 'Dhoni', '1981-07-07', 'India', 'Wicket-Keeper', 2),
('5','David', 'Warner', '1986-10-27', 'Australia', 'Batsman', 7),
('6','AB', 'de Villiers', '1984-02-17', 'South Africa', 'Batsman', 5),
('7','Shikhar', 'Dhawan', '1985-12-05', 'India', 'Batsman', 6),
('8','Steve', 'Smith', '1989-06-02', 'Australia', 'Batsman', 8),
('9','Chris', 'Gayle', '1979-09-21', 'West Indies', 'Batsman', 6),
('10','KL', 'Rahul', '1992-04-18', 'India', 'Batsman', 6),
('11','Hardik', 'Pandya', '1993-10-11', 'India', 'All-Rounder', 1),
('12','Ravindra', 'Jadeja', '1988-12-06', 'India', 'All-Rounder', 2),
('13','Rashid', 'Khan', '1998-09-20', 'Afghanistan', 'Bowler', 7),
('14','Jasprit', 'Bumrah', '1993-12-06', 'India', 'Bowler', 1),
('15','Kagiso', 'Rabada', '1995-05-25', 'South Africa', 'Bowler', 4),
('16','Sunil', 'Narine', '1988-05-26', 'West Indies', 'All-Rounder', 3),
('17','Yuzvendra', 'Chahal', '1990-07-23', 'India', 'Bowler', 5),
('18','Andre', 'Russell', '1988-04-29', 'West Indies', 'All-Rounder', 3),
```



('19','Suresh', 'Raina', '1986-11-27', 'India', 'Batsman', 2),
('20','Ben', 'Stokes', '1991-06-04', 'England', 'All-Rounder', 8),
('21','Rahul', 'Tewatia', '1993-05-20', 'India', 'All-Rounder', 8),
('22','Mayank', 'Agarwal', '1991-02-16', 'India', 'Batsman', 6),
('23','Nicholas', 'Pooran', '1995-10-02', 'West Indies', 'Wicket-Keeper', 6),
('24','Chris', 'Jordan', '1988-10-04', 'England', 'Bowler', 6),
('25','Eoin', 'Morgan', '1986-09-10', 'England', 'Batsman', 3),
('26','Pat', 'Cummins', '1993-05-08', 'Australia', 'Bowler', 3),
('27','Dinesh', 'Karthik', '1985-06-01', 'India', 'Wicket-Keeper', 3),
('28','Nitish', 'Rana', '1994-12-27', 'India', 'Batsman', 3),
('29','Shreyas', 'Iyer', '1994-12-06', 'India', 'Batsman', 4),
('30','Shubman', 'Gill', '1999-09-08', 'India', 'Batsman', 3),
('31','Prithvi', 'Shaw', '1999-11-09', 'India', 'Batsman', 4),
('32','Ajinkya', 'Rahane', '1988-06-06', 'India', 'Batsman', 8),
('33','Marcus', 'Stoinis', '1989-08-16', 'Australia', 'All-Rounder', 4),
('34','Ishan', 'Kishan', '1998-07-18', 'India', 'Batsman', 1),
('35','Trent', 'Boult', '1989-07-22', 'New Zealand', 'Bowler', 1),
('36','Quinton', 'de Kock', '1992-12-17', 'South Africa', 'Wicket-Keeper', 1),
('37','Dwayne', 'Bravo', '1983-10-07', 'West Indies', 'All-Rounder', 2),
('38','Kieron', 'Pollard', '1987-05-12', 'West Indies', 'All-Rounder', 1),
('39','Imran', 'Tahir', '1979-03-27', 'South Africa', 'Bowler', 2),
('40','Faf', 'du Plessis', '1984-07-13', 'South Africa', 'Batsman', 2),
('41','Ambati', 'Rayudu', '1985-09-23', 'India', 'Batsman', 1),
('42','Deepak', 'Chahar', '1992-08-07', 'India', 'Bowler', 2),
('43','Shardul', 'Thakur', '1991-10-16', 'India', 'Bowler', 2),
('44','Ravichandran', 'Ashwin', '1986-09-17', 'India', 'Bowler', 2),
('45','Mohammed', 'Shami', '1990-09-03', 'India', 'Bowler', 4),
('46','Rishabh', 'Pant', '1997-10-04', 'India', 'Wicket-Keeper', 4),
('47','Alex', 'Hales', '1989-01-03', 'England', 'Batsman', 4),
('48','Tom', 'Curran', '1995-03-12', 'England', 'All-Rounder', 8),
('49','Rohit', 'Kumar', '1990-12-01', 'India', 'Bowler', 4),
('50','Yusuf', 'Pathan', '1982-11-17', 'India', 'All-Rounder', 4),
('51','Colin', 'Munro', '1987-03-11', 'New Zealand', 'Batsman', 4),
('52','Manish', 'Pandey', '1989-09-10', 'India', 'Batsman', 7),
('53','Bhuvneshwar', 'Kumar', '1990-02-05', 'India', 'Bowler', 7),
('54','Sandeep', 'Sharma', '1993-05-18', 'India', 'Bowler', 7),
('55','Martin', 'Guptill', '1986-09-30', 'New Zealand', 'Batsman', 1),
('56','Jason', 'Roy', '1990-07-21', 'England', 'Batsman', 4),
('57','Lungi', 'Ngidi', '1996-03-29', 'South Africa', 'Bowler', 2),
('58','Moeen', 'Ali', '1987-06-18', 'England', 'All-Rounder', 2),
('59','Varun', 'Chakravarthy', '1991-08-29', 'India', 'Bowler', 3),
('60','Arjun', 'Tendulkar', '1999-09-24', 'India', 'Bowler', 1),
('61','Devdutt', 'Padikkal', '2000-07-07', 'India', 'Batsman', 5),
('62','Chris', 'Woakes', '1989-03-02', 'England', 'All-Rounder', 4),
('63','Shimron', 'Hetmyer', '1996-12-26', 'West Indies', 'Batsman', 5),



('64', 'Ravindra', 'Jadeja', '1988-12-06', 'India', 'All-Rounder', 2),
('65', 'Sanju', 'Samson', '1994-11-11', 'India', 'Wicket-Keeper', 8),
('66', 'Jos', 'Buttler', '1990-09-08', 'England', 'Wicket-Keeper', 8),
('67', 'Kuldeep', 'Yadav', '1994-12-14', 'India', 'Bowler', 3),
('68', 'Suryakumar', 'Yadav', '1990-09-14', 'India', 'Batsman', 1),
('69', 'Deepak', 'Hooda', '1995-04-19', 'India', 'All-Rounder', 6),
('70', 'Harbhajan', 'Singh', '1980-07-03', 'India', 'Bowler', 2),
('71', 'Murugan', 'Ashwin', '1990-09-08', 'India', 'Bowler', 4),
('72', 'Rahul', 'Chahar', '1999-08-04', 'India', 'Bowler', 1),
('73', 'Manan', 'Vohra', '1993-07-18', 'India', 'Batsman', 6),
('74', 'Krishnappa', 'Gowtham', '1988-10-20', 'India', 'All-Rounder', 8),
('75', 'Mohammad', 'Nabi', '1985-01-01', 'Afghanistan', 'All-Rounder', 7),
('76', 'Anrich', 'Nortje', '1993-11-16', 'South Africa', 'Bowler', 4),
('77', 'Riley', 'Meredith', '1996-12-21', 'Australia', 'Bowler', 6),
('78', 'Sheldon', 'Cottrell', '1989-08-19', 'West Indies', 'Bowler', 6),
('79', 'T.', 'Natarajan', '1991-05-27', 'India', 'Bowler', 7),
('80', 'Isuru', 'Udama', '1988-02-17', 'Sri Lanka', 'All-Rounder', 5),
('81', 'Josh', 'Hazlewood', '1991-01-08', 'Australia', 'Bowler', 2),
('82', 'Rahul', 'Tripathi', '1991-03-02', 'India', 'Batsman', 3),
('83', 'Jaydev', 'Unadkat', '1991-10-18', 'India', 'Bowler', 8),
('84', 'Shivam', 'Dube', '1993-06-26', 'India', 'All-Rounder', 5),
('85', 'Naveen', 'Ul Haq', '2000-02-28', 'Afghanistan', 'Bowler', 5),
('86', 'Ravi', 'Bishnoi', '2000-08-05', 'India', 'Bowler', 6),
('87', 'Saurabh', 'Tiwary', '1989-12-30', 'India', 'Batsman', 1),
('88', 'Jayant', 'Yadav', '1989-01-22', 'India', 'Bowler', 2),
('89', 'Manjot', 'Kalra', '1999-01-15', 'India', 'Batsman', 4),
('90', 'Sherfane', 'Rutherford', '1998-08-15', 'West Indies', 'Batsman', 1),
('91', 'Prasidh', 'Krishna', '1996-02-19', 'India', 'Bowler', 8),
('92', 'Siddharth', 'Kaul', '1990-05-19', 'India', 'Bowler', 7),
('93', 'Shahbaz', 'Ahmed', '1994-12-02', 'India', 'Bowler', 4),
('94', 'Tushar', 'Deshpande', '1995-05-15', 'India', 'Bowler', 4),
('95', 'Abdul', 'Samad', '2001-10-28', 'India', 'All-Rounder', 7),
('96', 'Yashasvi', 'Jaiswal', '2001-12-28', 'India', 'Batsman', 8),
('97', 'Anukul', 'Roy', '1998-12-30', 'India', 'All-Rounder', 3),
('98', 'Kamlesh', 'Nagarkoti', '1999-12-28', 'India', 'Bowler', 3),
('99', 'Mujeeb', 'Ur Rahman', '2001-03-28', 'Afghanistan', 'Bowler', 6),
('100', 'Tom', 'Banton', '1998-11-11', 'England', 'Batsman', 7),
('101', 'Rasikh', 'Salam', '2001-02-21', 'India', 'Bowler', 8),
('102', 'Pravin', 'Dubey', '1993-06-05', 'India', 'Bowler', 2),
('103', 'Priyam', 'Garg', '2001-11-30', 'India', 'Batsman', 7),
('104', 'Jalaj', 'Saxena', '1986-11-15', 'India', 'All-Rounder', 4);

SYMBIOSIS INTERNATIONAL

(Established under
UGC Act 1956) Re



(DEEMED UNIVERSITY)

section 3 of the
- accredited by

NAAC with 'A' Grade

Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

VALUES

('105','Adam','Zampa','1992-03-31','Australia','Bowler',5),
('106','Parthiv','Patel','1985-03-09','India','Wicket-Keeper',5),
('107','Aaron','Finch','1986-09-17','Australia','Batsman',5),
('108','Mohammed','Siraj','1994-04-13','India','Bowler',5);

INSERT INTO Coaches

VALUES

('1','Stephen Fleming', 1, 'Head Coach'),
(2,'Michael Hussey', 1, 'Batting Coach'),
(3,'Andy Flower', 1, 'Bowling Coach'),
(4,'Anil Kumble', 2, 'Head Coach'),
(5,'Jonty Rhodes', 2, 'Fielding Coach'),
(6,'Sanjay Bangar', 2, 'Batting Coach'),
(7,'Brendon McCullum', 3, 'Head Coach'),
(8,'Abhishek Nayar', 3, 'Batting Coach'),
(9,'Kyle Mills', 3, 'Bowling Coach'),
(10,'Ricky Ponting', 4, 'Head Coach'),
(11,'Pravin Amre', 4, 'Batting Coach'),
(12,'Ryan Harris', 4, 'Bowling Coach'),
(13,'Trevor Bayliss', 4, 'Head Coach'),
(14,'Simon Katich', 5, 'Head Coach'),
(15,'VVS Laxman', 5, 'Mentor'),
(16,'Tom Moody', 6, 'Head Coach'),
(17,'Adam Gilchrist', 6, 'Batting Coach'),
(18,'Daniel Vettori', 6, 'Bowling Coach'),
(19,'Shane Warne', 7, 'Mentor'),
(20,'Andrew McDonald', 7, 'Head Coach'),
(21,'Amol Muzumdar', 7, 'Batting Coach'),
(22,'Mike Hesson', 8, 'Batting Coach'),
(23,'Simon Katich', 8, 'Head Coach'),
(24,'Adam Griffith', 8, 'Bowling Coach');

select * from Coaches where TeamID="2";

INSERT INTO Venues

VALUES

('1', 'Wankhede Stadium', 'Mumbai'),
(2, 'Eden Gardens', 'Kolkata'),
(3, 'M. Chinnaswamy Stadium', 'Bengaluru'),
(4, 'Arun Jaitley Stadium', 'Delhi'),
(5, 'Rajiv Gandhi International Cricket Stadium', 'Hyderabad'),
(6, 'Sawai Mansingh Stadium', 'Jaipur'),



('7', 'Holkar Cricket Stadium', 'Indore'),
— ('8', 'M. A. Chidambaram Stadium', 'Chennai'),
— ('9', 'JSCA International Stadium Complex', 'Ranchi'),
— ('10', 'Barabati Stadium', 'Cuttack'),
— ('11', 'Green Park', 'Kanpur');

select * from Venues where VenueID='5';

INSERT INTO Umpires
VALUES
(1, 'Richard', 'Kettleborough', 'England'),
(2, 'Nitin', 'Menon', 'India'),
(3, 'Anil', 'Chaudhary', 'India'),
(4, 'Bruce', 'Oxenford', 'Australia'),
(5, 'Chris', 'Gaffaney', 'New Zealand'),
(6, 'Rod', 'Tucker', 'Australia'),
(7, 'Marius', 'Erasmus', 'South Africa'),
(8, 'Aleem', 'Dar', 'Pakistan'),
(9, 'Paul', 'Reiffel', 'Australia'),
(10, 'Sundaram', 'Ravi', 'India'),
(11, 'Nigel', 'Llong', 'England'),
(12, 'Kumar', 'Dharmasena', 'Sri Lanka'),
(13, 'Ian', 'Gould', 'England'),
(14, 'S. Ravi', 'Sundaram', 'India'),
(15, 'Michael', 'Gough', 'England');

select * from Umpires where UmpireID='5';

INSERT INTO MatchOfficials
VALUES
(1, 'David', 'Boon', 'Australia'),
(2, 'Ranjan', 'Madugalle', 'Sri Lanka'),
(3, 'Chris', 'Broad', 'England'),
(4, 'Jeff', 'Crowe', 'New Zealand'),
(5, 'Javagal', 'Srinath', 'India');

INSERT INTO Sponsors
VALUES
(1, 'Gulf Oil', 2),
(2, 'ACC', 6),
(3, 'Royal Challenge', 5),
(4, 'Jio', 1),
(5, 'JOY', 3),



(6, 'JSW Paints', '4'),

(7, 'TCL', '7'),
(8, 'Luminous Power Technologies', 8);

select * from Sponsors where TeamID='5';

INSERT INTO Tournaments
VALUES
(1, 'IPL 2020', '2020-09-19', '2020-11-10');

select * from Matches;

SET SQL_SAFE_UPDATES = 0;
-- Insert your script

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result,
ManOfTheMatchID) VALUES
(1, '2020-04-05', 1, 2, 1, 'CSK', 4),
(2, '2020-04-06', 2, 8, 5, 'KKR', 30),
(3, '2020-04-07', 3, 3, 6, 'KXIP', 23),
(4, '2020-04-08', 4, 7, 4, 'DC', 46);
-- Add more matches as needed
;

-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result,
ManOfTheMatchID) VALUES
(5, '2020-04-09', 5, 5, 2, 'RCB', 1),
(6, '2020-04-10', 6, 6, 1, 'MI', 14),
(7, '2020-04-11', 7, 4, 8, 'RR', 20),
(8, '2020-04-12', 8, 7, 3, 'SRH', 13),
(9, '2020-04-13', 1, 2, 6, 'KXIP', 7),
(10, '2020-04-14', 2, 5, 4, 'RCB', 6);
-- Add more matches as needed
;

-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result,
ManOfTheMatchID) VALUES
(11, '2020-04-15', 3, 1, 7, 'MI', 14),
(12, '2020-04-16', 4, 8, 3, 'RR', 48),
(13, '2020-04-17', 5, 4, 2, 'DC', 33),
(14, '2020-04-18', 6, 7, 6, 'SRH', 75),
(15, '2020-04-19', 7, 3, 5, 'KKR', 16),
(16, '2020-04-20', 8, 8, 1, 'RR', 32);
-- Add more matches as needed



;

~~-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs~~

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES

(17, '2020-04-21', 1, 2, 7, 'CSK', 40),
(18, '2020-04-22', 2, 4, 3, 'KKR', 30),
(19, '2020-04-23', 3, 6, 8, 'KXIP', 9),
(20, '2020-04-24', 4, 5, 1, 'MI', 14),
(21, '2020-04-25', 5, 3, 2, 'KKR', 25),
(22, '2020-04-26', 6, 8, 7, 'SRH', 52);

~~-- Add more matches as needed~~

;

~~-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs~~

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES

(23, '2020-04-27', 7, 1, 4, 'MI', 14),
(24, '2020-04-28', 8, 5, 6, 'RCB', 1),
(25, '2020-04-29', 1, 2, 8, 'CSK', 64),
(26, '2020-04-30', 2, 3, 1, 'MI', 2),
(27, '2020-05-01', 3, 7, 5, 'SRH', 53),
(28, '2020-05-02', 4, 4, 6, 'DC', 31);

~~-- Add more matches as needed~~

;

~~-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs~~

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES

(29, '2020-05-03', 5, 1, 2, 'CSK', 19),
(30, '2020-05-04', 6, 5, 8, 'RCB', 17),
(31, '2020-05-05', 7, 6, 3, 'KKR', 16),
(32, '2020-05-06', 8, 4, 7, 'DC', 46),
(33, '2020-05-07', 1, 2, 5, 'CSK', 42),
(34, '2020-05-08', 2, 1, 6, 'MI', 14);

~~-- Add more matches as needed~~

;

~~-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs~~

INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES

(35, '2020-05-09', 3, 8, 4, 'RR', 65),
(36, '2020-05-10', 4, 3, 7, 'SRH', 54),
(37, '2020-05-11', 5, 6, 2, 'KXIP', 10),
(38, '2020-05-12', 6, 4, 5, 'DC', 15),
(39, '2020-05-13', 7, 7, 1, 'SRH', 5),
(40, '2020-05-14', 8, 3, 8, 'KKR', 27);

~~-- Add more matches as needed~~

;



-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

~~INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES~~

(41, '2020-05-15', 1, 2, 4, 'DC', 15),
 (42, '2020-05-16', 2, 6, 7, 'KXIP', 10),
 (43, '2020-05-17', 3, 5, 3, 'RCB', 1),
 (44, '2020-05-18', 4, 1, 8, 'MI', 14),
 (45, '2020-05-19', 5, 7, 2, 'SRH', 5),
 (46, '2020-05-20', 6, 3, 4, 'DC', 15);

-- Add more matches as needed

;

-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

~~INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES~~

(47, '2020-05-21', 7, 8, 6, 'KXIP', 7),
 (48, '2020-05-22', 8, 1, 5, 'MI', 11),
 (49, '2020-05-23', 1, 2, 3, 'KKR', 30),
 (50, '2020-05-24', 2, 7, 8, 'RR', 66),
 (51, '2020-05-25', 3, 4, 1, 'DC', 29),
 (52, '2020-05-26', 4, 6, 5, 'RCB', 1);

-- Add more matches as needed

;

-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

~~INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES~~

(53, '2020-05-27', 5, 8, 2, 'CSK', 39),
 (54, '2020-05-28', 6, 1, 3, 'MI', 14);

-- Insert more data into the Matches table for IPL 2020 with respect to Team IDs

~~INSERT INTO Matches (MatchID, MatchDate, VenueID, Team1ID, Team2ID, Result, ManOfTheMatchID) VALUES~~

(55, '2020-05-29', 7, 5, 7, 'RCB', 61),
 (56, '2020-05-30', 8, 6, 4, 'KXIP', 10),
 (57, '2020-05-31', 8, 1, 4, 'MI', 2),
 (58, '2020-06-2', 8, 5, 7, 'SRH', 5),
 (59, '2020-06-4', 8, 7, 4, 'DC', 33),
 (60, '2020-06-6', 8, 1, 4, 'MI', 14);

SELECT * FROM Matches;

UPDATE MATCHES

SET RESULT = 'RR'

WHERE MATCHID = 19;



```
SELECT * FROM Matches WHERE RESULT ='KXIP';
```

```
SELECT * FROM MATCHES;
```

```
INSERT INTO PointsTable (Position, TeamID, NumberOfMatches, Points) VALUES  
(1, 1, 14, 18), -- Mumbai Indians  
(2, 4, 14, 16), -- DC  
(3, 7, 14, 14), -- SRH  
(4, 5, 14, 14), -- RCB  
(5, 3, 14, 14), -- Kolkata Knight Riders  
(6, 6, 14, 12), -- Kings XI Punjab  
(7, 2, 14, 12), -- CSK  
(8, 8, 14, 12); -- Rajasthan Royals
```

```
SELECT * FROM PointsTable;
```

```
INSERT INTO OrangeCap (Position, PlayerID, Runs, NumberOfMatches) VALUES  
(1, 10, 600, 14), -- K1 rahul  
(2, 7, 520, 14), -- DHAWAN  
(3, 5, 490, 14), -- WARNER  
(4, 29, 450, 16), -- IYER  
(5, 34, 410, 16); -- KISHAN
```

```
INSERT INTO PurpleCap (Position, PlayerID, Wickets, NumberOfMatches) VALUES  
(1, 15, 28, 16), -- rABADA  
(2, 14, 24, 16), -- BUMRAH  
(3, 35, 23, 16), -- BOULT  
(4, 76, 21, 14), -- NORTJE  
(5, 17, 20, 15); -- CHAHAL
```

```
-- Insert data into the TournamentMatches table for MatchID values from 1 to 10
```

```
INSERT INTO TournamentMatches (TournamentID, MatchID) VALUES
```

```
(1, 1),  
(1, 2),  
(1, 3),  
(1, 4),  
(1, 5),  
(1, 6),  
(1, 7),  
(1, 8),  
(1, 9),  
(1, 10);
```

```
-- Insert data into the TournamentMatches table for MatchID values from 10 to 40
```



INSERT INTO TournamentMatches (TournamentID, MatchID) VALUES

(1, 11),

(1, 12),

(1, 13),

(1, 14),

(1, 15),

(1, 16),

(1, 17),

(1, 18),

(1, 19),

(1, 20),

(1, 21),

(1, 22),

(1, 23),

(1, 24),

(1, 25),

(1, 26),

(1, 27),

(1, 28),

(1, 29),

(1, 30),

(1, 31),

(1, 32),

(1, 33),

(1, 34),

(1, 35),

(1, 36),

(1, 37),

(1, 38),

(1, 39),

(1, 40);

-- Insert data into the TournamentMatches table for MatchID values from 41 to 60

INSERT INTO TournamentMatches (TournamentID, MatchID) VALUES

(1, 41),

(1, 42),

(1, 43),

(1, 44),

(1, 45),

(1, 46),

(1, 47),

(1, 48),

(1, 49),

(1, 50),

(1, 51),



(1, 52),
(1, 53),
(1, 54),
(1, 55),
(1, 56),
(1, 57),
(1, 58),
(1, 59),
(1, 60);

ALTER TABLE Commentator
DROP COLUMN MatchID;

SET SQL_SAFE_UPDATES = 0;
ALTER TABLE commentator
DROP FOREIGN KEY MatchID;

desc commentator;
show create table commentator;

UPDATE commentator
SET MatchID = ";
drop table commentator;

CREATE TABLE Commentator (
CommentatorID INT PRIMARY KEY,
FirstName Varchar(255),
LastName Varchar(255)
);

ALTER TABLE Matches
ADD COLUMN CommentatorID INT,
ADD CONSTRAINT fk_commentator
FOREIGN KEY (CommentatorID)
REFERENCES commentator(CommentatorID);
desc commentator;
ALTER TABLE Matches
DROP COLUMN Commentator_ID;



INSERT INTO Commentator (CommentatorID, FirstName, LastName)

VALUES

(1, 'Jatin', 'Sapru'),
(2, 'Gaurav', 'Kapoor'),
(3, 'Ravi', 'Shastri'),
(4, 'Ajit', 'Agarkar'),
(5, 'Sunil', 'Gavaskar'),
(6, 'Ian', 'Bishop'),
(7, 'Harsha', 'Bhogle');

UPDATE Matches

SET CommentatorID = FLOOR(1 + RAND() * 7);

drop table manofthematch;

ALTER TABLE ManOfTheMatch

CHANGE COLUMN InningsID MatchID INT;

INSERT INTO ManOfTheMatch (MatchID, PlayerID, RunsScored, BallsFaced,

WicketsTaken,

OversBowled)

VALUES

(1, 4, 72, 48, 0, 0.0),
(2, 30, 97, 62, 0, 0.0),
(3, 23, 92, 40, 0, 0.0),
(4, 46, 71, 25, 0, 0.0),
(5, 1, 83, 55, 0, 0.0),
(6, 14, 4, 3, 4, 4.0),
(7, 20, 45, 30, 2, 3.0),
(8, 13, 0, 0, 4, 4.0),
(9, 7, 62, 30, 0, 0.0),
(10, 6, 95, 33, 0, 0),
(11, 14, 0, 0, 3, 4.0),
(12, 48, 22, 9, 3, 4.0),
(13, 33, 75, 45, 1, 2.0),
(14, 75, 30, 18, 3, 4.0),
(15, 16, 52, 27, 3, 4.0),
(16, 32, 72, 55, 0, 0.0),
(17, 40, 69, 36, 0, 0.0),
(18, 30, 62, 40, 0, 0.0),
(19, 9, 127, 48, 0, 1.0),
(20, 14, 3, 4, 5, 4.0),
(21, 25, 60, 35, 0, 0.0),
(22, 52, 48, 20, 0, 0.0),
(23, 14, 0, 0, 5, 4.0),
(24, 1, 70, 40, 0, 0.0),
(25, 64, 40, 25, 3, 4.0),



(26, 2, 87, 47, 0, 0.0),
~~(27, 53, 25, 18, 4, 4.0)~~,
(28, 31, 80, 50, 0, 0.0),
(29, 19, 87, 25, 1, 2.0),
(30, 17, 0, 0, 3, 4.0),
(31, 16, 40, 30, 2, 4.0),
(32, 46, 120, 65, 0, 0.0),
(33, 42, 20, 15, 3, 4.0),
(34, 14, 0, 0, 3, 4.0),
(35, 65, 87, 45, 0, 0.0),
(36, 54, 0, 0, 3, 4.0),
(37, 10, 110, 62, 0, 0.0),
(38, 15, 0, 0, 6, 4.0),
(39, 5, 125, 72, 0, 0.0),
(40, 27, 60, 35, 0, 0.0),
(41, 15, 0, 0, 4, 4.0),
(42, 10, 82, 50, 0, 0.0),
(43, 1, 102, 56, 0, 0.0),
(44, 14, 0, 0, 4, 4.0),
(45, 5, 80, 45, 0, 0.0),
(46, 15, 0, 0, 3, 4.0),
(47, 7, 67, 32, 0, 0.0),
(48, 11, 80, 50, 1, 4.0),
(49, 30, 83, 30, 0, 0.0),
(50, 66, 68, 28, 0, 0.0),
(51, 29, 60, 35, 0, 0.0),
(52, 1, 72, 42, 1, 1.0),
(53, 39, 0, 0, 3, 4.0),
(54, 14, 0, 0, 6, 4.0),
(55, 61, 75, 45, 0, 0.0),
(56, 10, 69, 35, 0, 0.0),
(57, 2, 80, 45, 1, 2.0),
(58, 5, 87, 48, 0, 0.0),
(59, 33, 53, 28, 2, 3.0),
(60, 14, 0, 2, 3, 4.0);

UPDATE ManOfTheMatch
SET PlayerID = 28
WHERE MatchID = 3;

INSERT INTO Innings (InningsID, MatchID, BattingTeamID, BowlingTeamID, TotalRuns,
TotalWickets)
VALUES
(1, 1, 2, 1, 200, 4),
(2, 1, 1, 2, 180, 6),



(3, 2, 8, 5, 210, 2),
~~(4, 2, 5, 8, 211, 5),~~
(5, 3, 3, 6, 190, 3),
(6, 3, 6, 3, 172, 10),
(7, 4, 7, 4, 190, 3),
(8, 4, 4, 7, 191, 3),
(9, 5, 5, 2, 180, 3),
(10, 5, 2, 5, 132, 9),
(11, 6, 6, 1, 120, 3),
(12, 6, 1, 6, 123, 0),
(13, 7, 4, 8, 167, 3),
(14, 7, 8, 4, 170, 6),
(15, 8, 3, 7, 200, 6),
(16, 8, 7, 3, 202, 5),
(17, 9, 2, 6, 144, 3),
(18, 9, 6, 2, 147, 3),
(19, 10, 5, 4, 230, 6),
(20, 10, 4, 5, 130, 9),
(21, 11, 7, 1, 134, 9),
(22, 11, 1, 7, 136, 4),
(23, 12, 8, 3, 179, 3),
(24, 12, 3, 8, 178, 3),
(25, 13, 2, 4, 190, 3),
(26, 13, 4, 2, 191, 5),
(27, 14, 7, 6, 187, 6),
(28, 14, 6, 7, 179, 9),
(29, 15, 3, 5, 120, 9),
(30, 15, 5, 3, 49, 10),
(31, 16, 8, 1, 200, 6),
(32, 16, 1, 8, 178, 8),
(33, 17, 2, 7, 156, 5),
(34, 17, 7, 2, 147, 9),
(35, 18, 4, 3, 211, 9),
(36, 18, 3, 4, 215, 7),
(37, 19, 6, 8, 100, 10),
(38, 19, 8, 6, 102, 3),
(39, 20, 5, 1, 130, 10),
(40, 20, 1, 5, 131, 5),
(41, 21, 2, 3, 170, 3),
(42, 21, 3, 2, 172, 2),
(43, 22, 7, 8, 186, 8),
(44, 22, 8, 7, 143, 10),
(45, 23, 4, 1, 187, 3),
(46, 23, 1, 4, 188, 6),
(47, 24, 6, 5, 163, 8),



(48, 24, 5, 6, 144, 10),
~~(49, 25, 2, 8, 210, 4),~~
(50, 25, 8, 2, 190, 7),
(51, 26, 1, 3, 170, 3),
(52, 26, 3, 1, 152, 9),
(53, 27, 5, 7, 184, 3),
(54, 27, 7, 5, 185, 8),
(55, 28, 6, 4, 203, 6),
(56, 28, 4, 6, 204, 9),
(57, 29, 2, 1, 230, 3),
(58, 29, 1, 2, 135, 9),
(59, 30, 8, 5, 156, 6),
(60, 30, 5, 8, 157, 9),
(61, 31, 3, 6, 181, 8),
(62, 31, 6, 3, 128, 9),
(63, 32, 4, 7, 198, 3),
(64, 32, 7, 4, 156, 6),
(65, 33, 2, 5, 167, 4),
(66, 33, 5, 2, 162, 6),
(67, 34, 6, 1, 119, 10),
(68, 34, 1, 6, 122, 2),
(69, 35, 4, 8, 168, 7),
(70, 35, 8, 4, 172, 3),
(71, 36, 3, 7, 177, 4),
(72, 36, 7, 3, 180, 2),
(73, 37, 2, 6, 150, 8),
(74, 37, 6, 2, 151, 3),
(75, 38, 4, 5, 172, 3),
(76, 38, 5, 4, 125, 9),
(77, 39, 7, 1, 183, 3),
(78, 39, 1, 7, 142, 10),
(79, 40, 3, 8, 176, 5),
(80, 40, 8, 3, 142, 8),
(81, 41, 2, 4, 143, 8),
(82, 41, 4, 2, 146, 3),
(83, 42, 6, 7, 220, 3),
(84, 42, 7, 6, 180, 5),
(85, 43, 3, 5, 176, 4),
(86, 43, 5, 3, 187, 2),
(87, 44, 8, 1, 125, 9),
(88, 44, 1, 8, 127, 1),
(89, 45, 2, 7, 138, 9),
(90, 45, 7, 2, 140, 3),
(91, 46, 4, 3, 188, 4),
(92, 46, 3, 4, 191, 3),



(93, 47, 8, 6, 147, 6),
~~(94, 47, 6, 8, 149, 2),~~
(95, 48, 5, 1, 186, 5),
(96, 48, 1, 5, 189, 3),
(97, 49, 2, 3, 171, 6),
(98, 49, 3, 2, 174, 3),
(99, 50, 7, 8, 192, 3),
(100, 50, 8, 7, 194, 2),
(101, 51, 4, 1, 225, 2),
(102, 51, 1, 4, 170, 5),
(103, 52, 5, 6, 207, 3),
(104, 52, 6, 5, 177, 3),
(105, 53, 2, 8, 176, 3),
(106, 53, 8, 2, 125, 9),
(107, 54, 3, 1, 141, 7),
(108, 54, 1, 3, 143, 2),
(109, 55, 5, 7, 190, 3),
(110, 55, 7, 5, 192, 3),
(111, 56, 6, 4, 163, 3),
(112, 56, 4, 6, 140, 8),
(113, 57, 4, 1, 197, 3),
(114, 57, 1, 4, 199, 2),
(115, 58, 5, 7, 148, 4),
(116, 58, 7, 5, 152, 2),
(117, 59, 4, 7, 222, 5),
(118, 59, 7, 4, 190, 8),
(119, 60, 4, 1, 158, 6),
(120, 60, 1, 4, 161, 2);

select * from matches;

drop table awards;

```
CREATE TABLE Awards (
    TournamentID INT,
    AwardID INT PRIMARY KEY,
    AwardName VARCHAR(255) NOT NULL,
    PlayerID INT,
    FOREIGN KEY (TournamentID) REFERENCES Tournaments(TournamentID),
    FOREIGN KEY (PlayerID) REFERENCES Players (PlayerID)
);
```

```
INSERT INTO Awards (TournamentID, AwardID, AwardName, PlayerID)
VALUES
```



(1, 1, 'Most Valuable Player', 14),
(1, 2, 'Orange Cap', 10),
 (1, 3, 'Purple Cap', 15),
 (1, 4, 'Catch Of The Season', 64),
 (1, 5, 'Best Fielder', 19),
 (1, 6, 'Most Sixes', 2),
 (1, 7, 'Best Emerging Player', 30);

select * from venues;

alter table venues
 add column State varchar(255);

-- Update the "State" column with data for multiple venues in a single statement

UPDATE Venues

SET State = CASE

WHEN VenueID = 1 THEN 'Maharashtra'
 WHEN VenueID = 2 THEN 'West Bengal'
 WHEN VenueID = 3 THEN 'Karnataka'
 WHEN VenueID = 4 THEN 'Delhi'
 WHEN VenueID = 5 THEN 'Andhra Pradesh'
 WHEN VenueID = 6 THEN 'Rajasthan'
 WHEN VenueID = 7 THEN 'Madhya Pradesh'
 WHEN VenueID = 8 THEN 'Tamilnadu'
 WHEN VenueID = 9 THEN 'Jharkhand'
 WHEN VenueID = 10 THEN 'Odisha'
 WHEN VenueID = 11 THEN 'Uttar Pradesh'

-- Add more cases for other venues

ELSE 'Unknown'
 END

WHERE VenueID IN (1,2,3,4,5,6,7,8,9,10,11); -- Specify the VenueIDs you want to update

ALTER TABLE Awards
 ADD COLUMN Ceremony_Date VARCHAR(255) DEFAULT '2020-06-06';

select * from awards;

DELIMITER //

```
CREATE FUNCTION CalculatePlayerAgeByID (player_id INT)
RETURNS INT
DETERMINISTIC
BEGIN
```



DECLARE dob DATE;

DECLARE age INT;

-- Get date of birth

SELECT DateOfBirth INTO dob FROM Players WHERE PlayerID = player_id;

-- Calculate age

SET age = YEAR(CURRENT_DATE()) - YEAR(dob) -
 (DATE_FORMAT(CURRENT_DATE(), '%m%d') < DATE_FORMAT(dob, '%m%d'));

RETURN age;

END//

DELIMITER ;

select CalculatePlayerAgeByID (2);

DELIMITER \$\$

CREATE FUNCTION CalculateTotalRunsByTeamID(team_id INT)

RETURNS INT

DETERMINISTIC

BEGIN

DECLARE total_runs INT;

SELECT SUM(TotalRuns)

INTO total_runs

FROM Innings

WHERE BattingTeamID = team_id;

RETURN total_runs;

END\$\$

DELIMITER ;

select CalculateTotalRunsByTeamID(1);

DELIMITER \$\$

CREATE FUNCTION CalculateTotalWicketsByTeamID(team_id INT)

RETURNS INT



DETERMINISTIC

BEGIN

```
DECLARE total_wickets INT;

SELECT SUM(TotalWickets)
INTO total_wickets
FROM Innings
WHERE BowlingTeamID = team_id;

RETURN total_wickets;
END$$

DELIMITER ;

select CalculateTotalWicketsByTeamID(2);

ALTER TABLE Tournaments
ADD Column WinnerID INT,
ADD FOREIGN KEY (WinnerID) REFERENCES Teams(TeamID);

select * from Tournaments;

DROP TABLE Tournaments;

SET SQL_SAFE_UPDATES = 0;

ALTER TABLE Tournaments
ADD Column WINNER varchar(255) AFTER EndDate;
```

show triggers;

DELIMITER \$\$

```
CREATE TRIGGER UpdateTournamentWinner20
AFTER UPDATE ON Matches
FOR EACH ROW
BEGIN
IF NEW.MatchID = 60 AND NEW.Result IS NOT NULL THEN
UPDATE Tournaments
SET Winner = NEW.Result
WHERE TournamentID = 1; -- Adjust the condition as needed
END IF;
END$$
```



DELIMITER ;

UPDATE Tournaments

SET WinnerID = 1

WHERE TournamentID = 1;

```
SELECT P.FirstName, P.LastName
FROM Players P
INNER JOIN ManOfTheMatch M
ON P.PlayerID = M.PlayerID
GROUP BY P.PlayerID
HAVING COUNT(*) > 3;
```

```
SELECT P.FirstName, P.LastName, O.Runs
FROM Players P
INNER JOIN OrangeCap O
ON P.PlayerID = O.PlayerID
ORDER BY O.Runs DESC
LIMIT 5;
```

```
SELECT T.TeamName, SUM(I.TotalRuns) AS TotalRuns, SUM(I.TotalWickets) AS
TotalWickets
FROM Teams T
LEFT JOIN Innings I ON T.TeamID = I.BattingTeamID
LEFT JOIN Matches M ON I.MatchID = M.MatchID
LEFT JOIN TournamentMatches TM ON M.MatchID = TM.MatchID
WHERE TM.TournamentID = 1
GROUP BY T.TeamID;
```

```
SELECT T.TeamName, PT.Points
FROM Teams T
INNER JOIN PointsTable PT ON T.TeamID = PT.TeamID
ORDER BY PT.Points DESC
LIMIT 4;
```

```
SELECT P.Role, AVG(MM.OversBowled) AS AvgOversBowled
FROM Players P
LEFT JOIN ManOfTheMatch MM ON P.PlayerID = MM.PlayerID
GROUP BY P.Role;
```

DELIMITER \$\$

```
CREATE PROCEDURE IncreaseTeamPoints(IN teamID INT)
```



BEGIN

— Check if the team exists in PointsTable —

DECLARE teamExists INT;

SELECT COUNT(*) INTO teamExists FROM PointsTable WHERE TeamID = teamID;

-- If the team exists, update the points

IF teamExists > 0 THEN

 UPDATE PointsTable

 SET Points = Points + 2

 WHERE TeamID = teamID;

 SELECT 'Points increased by 2' AS Result;

ELSE

 SELECT 'Team not found in PointsTable' AS Result;

END IF;

END\$\$

DELIMITER ;

CALL IncreaseTeamPoints(1);

select * from venues;

DELIMITER \$\$

CREATE PROCEDURE AddVenue1(IN venueID INT, IN venueName VARCHAR(255), IN stadium VARCHAR(255))

BEGIN

 INSERT INTO Venues (VenueID, VenueName, Stadium) VALUES (venueID, venueName, stadium);

END\$\$

DELIMITER ;

CALL AddVenue1(12,'Narendra Modi Stadium', 'Ahmedabad');

Table outputs:



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench Local instance MySQL80 x

Navigator: MANAGEMENT, INSTANCE, PERFORMANCE, Administration, Schemas, Information.

IPL* x

```
1082
1083
1084 • select * from teams;
1085
1086
1087
```

Result Grid Filter Rows Edit Export/Import Wrap Cell Content Result Grid Form Editor Field Types Query Stats

TeamID	TeamName
1	Mumbai Indians
2	Chennai Super Kings
3	Kolkata Knight Riders
4	Delhi Capitals
5	Royal Challengers Bangalore
6	Kings XI Punjab
7	Sunrisers Hyderabad
8	Rajasthan Royals

Output: Action Output # Time Action Message Duration / Fetch
 146 23:14:32 CALL AddVenue(12,'Narendra Modi Stadium', 'Ahmedabad') 1 row(s) affected 0.016 sec
 147 23:14:41 select * from venue LIMIT 0, 1000 Error Code: 1146. Table 'ipl.venue' doesn't exist 0.015 sec
 148 23:14:47 select * from venues LIMIT 0, 1000 12 rows(s) returned 0.000 sec / 0.000 sec
 149 23:33:35 select * from teams LIMIT 0, 1000 0.000 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide

MySQL Workbench Local instance MySQL80 x

Navigator: MANAGEMENT, INSTANCE, PERFORMANCE, Administration, Schemas, Information.

IPL* x

```
1082
1083
1084 • select * from players;
1085
1086
1087
```

Result Grid Filter Rows Edit Export/Import Wrap Cell Content Result Grid Form Editor Field Types Query Stats

PlayerID	FirstName	LastName	DateOfBirth	Country	Role	TeamID
1	Wrat	Kohli	1988-11-05	India	Batsman	5
2	Rohit	Sharma	1987-04-30	India	Batsman	1
3	Kane	Williamson	1990-09-08	New Zealand	Batsman	7
4	MS	Dhoni	1981-07-07	India	Wicket-Keeper	2
5	David	Warner	1986-10-27	Australia	Batsman	7
6	AB	de Villiers	1984-02-17	South Africa	Batsman	5
7	Shikhar	Dhawal	1985-12-05	India	Batsman	6
8	Steve	Smith	1989-06-02	Australia	Batsman	8
9	Chris	Gayle	1979-09-21	West Indies	Batsman	6
10	KL	Rahul	1992-04-18	India	Batsman	6
11	Hardik	Pandya	1993-10-11	India	All-Rounder	1
12	Ravindra	Jadeja	1988-12-06	India	All-Rounder	2
13	Rashid	Khan	1998-09-20	Afghanistan	Bowler	7
14	Jasprit	Bumrah	1993-12-06	India	Bowler	1
15	Kagiso	RBabda	1995-05-25	South Africa	Bowler	4
16	Sunil	Narine	1988-05-26	West Indies	All-Rounder	3

Output: Action Output # Time Action Message Duration / Fetch
 147 23:14:41 select * from venue LIMIT 0, 1000 Error Code: 1146. Table 'ipl.venue' doesn't exist 0.015 sec
 148 23:14:47 select * from venues LIMIT 0, 1000 12 rows(s) returned 0.000 sec / 0.000 sec
 149 23:33:35 select * from teams LIMIT 0, 1000 0.000 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide

18°C Clear ENG 11:34 PM IN 10/28/2023



Founder: Prof. Dr. S. B. Mujumdar, M.Sc.,Ph.D. (Awarded Padma Bhushan and

MySQL Workbench

Local instance MySQL80

Navigator: MANAGEMENT, INSTANCE, PERFORMANCE, Administration, Schemas, Information.

IPL* (Query Editor):

```
1082
1083
1084 • select * from coaches;
1085
1086
1087
```

Result Grid:

CoachID	CoachName	TeamID	Specialization
1	Stephen Fleming	1	Head Coach
2	Michael Hussey	1	Batting Coach
3	Andy Flower	1	Bowling Coach
4	Anil Kumble	2	Head Coach
5	Jonny Rhodes	2	Fielding Coach
6	Sanjay Bangar	2	Batting Coach
7	Brendon McCullum	3	Head Coach
8	Abhishek Nayar	3	Batting Coach
9	Kyle Mills	3	Bowling Coach
10	Ricky Ponting	4	Head Coach
11	Pravin Amre	4	Batting Coach
12	Ryan Harris	4	Bowling Coach
13	Trevor Bayliss	4	Head Coach
14	Simon Katich	5	Head Coach
15	VVS Laxman	5	Mentor
16	Tom Moody	6	Head Coach

Output:

#	Time	Action	Message	Duration / Fetch
148	23:14:47	select * from venues LIMIT 0, 1000	12 row(s) returned	0.000 sec / 0.000 sec
149	23:33:35	select * from teams LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
150	23:34:43	select * from players LIMIT 0, 1000	108 row(s) returned	0.000 sec / 0.000 sec
151	23:34:55	select * from coaches LIMIT 0, 1000	24 model(s) returned	0.000 sec / 0.000 sec

MySQL Workbench

Local instance MySQL80

Navigator: MANAGEMENT, INSTANCE, PERFORMANCE, Administration, Schemas, Information.

IPL* (Query Editor):

```
1082
1083
1084 • select * from venues;
1085
1086
1087
```

Result Grid:

VenueID	VenueName	Stadu	State
1	Wankhede Stadium	Mumbai	Maharashtra
2	Eden Gardens	Kolkata	West Bengal
3	M. Chinnaswamy Stadium	Bengaluru	Karnataka
4	Arun Jaitley Stadium	Delhi	Delhi
5	Rajiv Gandhi International Cricket Stadium	Hyderabad	Andhra Pradesh
6	Sawai Mansingh Stadium	Jaipur	Rajasthan
7	Holkar Cricket Stadium	Indore	Madhya Pradesh
8	M. A. Chidambaram Stadium	Chennai	Tamilnadu
9	JSCA International Stadium Complex	Ranchi	Jharkhand
10	Barabati Stadium	Cuttack	Odisha
11	Green Park	Karapur	Uttar Pradesh
12	Narendra Modi Stadium	Ahmedabad	Gujarat

Output:

#	Time	Action	Message	Duration / Fetch
149	23:33:35	select * from teams LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
150	23:34:43	select * from players LIMIT 0, 1000	108 row(s) returned	0.000 sec / 0.000 sec
151	23:34:55	select * from coaches LIMIT 0, 1000	24 model(s) returned	0.000 sec / 0.000 sec
152	23:35:04	select * from venues LIMIT 0, 1000		0.015 sec / 0.000 sec



NAAC with 'A' Grade

Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information

No object selected Object Info Session

Type here to search

MySQL Workbench Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information

No object selected Object Info Session

Type here to search

Output

Action Output # Time Action Message Duration / Fetch
 151 23:34:55 select * from coaches LIMIT 0, 1000 24 row(s) returned 0.000 sec / 0.000 sec
 152 23:35:04 select * from venues LIMIT 0, 1000 12 row(s) returned 0.015 sec / 0.000 sec
 153 23:35:11 select * from matches LIMIT 0, 1000 60 row(s) returned 0.000 sec / 0.000 sec
 154 23:35:16 select * from umpires LIMIT 0, 1000 15 row(s) returned 0.015 sec / 0.000 sec

Output

Action Output # Time Action Message Duration / Fetch
 151 23:34:55 select * from coaches LIMIT 0, 1000 24 row(s) returned 0.000 sec / 0.000 sec
 152 23:35:04 select * from venues LIMIT 0, 1000 12 row(s) returned 0.015 sec / 0.000 sec
 153 23:35:11 select * from matches LIMIT 0, 1000 60 row(s) returned 0.000 sec / 0.000 sec
 154 23:35:16 select * from umpires LIMIT 0, 1000 15 row(s) returned 0.015 sec / 0.000 sec



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench

Local instance MySQL80

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

No object selected

Information

matchofficials 66

Output

Action Output

#	Time	Action	Message	Duration / Fetch
153	23:35:11	select * from matches LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
154	23:35:16	select * from umpires LIMIT 0, 1000	15 row(s) returned	0.015 sec / 0.000 sec
155	23:35:37	select * from matches LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
156	23:35:51	select * from matchofficials LIMIT 0, 1000		0.000 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide

MySQL Workbench

Local instance MySQL80

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

No object selected

tournaments 67

Output

Action Output

#	Time	Action	Message	Duration / Fetch
154	23:35:16	select * from matches LIMIT 0, 1000	15 row(s) returned	0.015 sec / 0.000 sec
155	23:35:37	select * from tournaments LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
156	23:35:51	select * from matchofficials LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
157	23:36:02	select * from tournaments LIMIT 0, 1000		0.015 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information

No object selected

Object Info Session

Type here to search

MySQL Workbench Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information

No object selected

Object Info Session

Type here to search

Result Grid

TournamentID MatchID

TournamentID	MatchID
1	1
1	2
1	3
1	4
1	5
1	6
1	7
1	8
1	9
1	10
1	11
1	12
1	13
1	14
1	15
1	16

tournamentmatches 68 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
155	23:35:37	select * from matches LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
156	23:35:51	select * from matchofficials LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
157	23:36:02	select * from tournaments LIMIT 0, 1000	1 row(s) returned	0.015 sec / 0.000 sec
158	23:36:07	select * from tournamentmatches LIMIT 0, 1000	60 rows(s) returned	0.016 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

CommentatorID FirstName LastName

CommentatorID	FirstName	LastName
1	Jatin	Sapru
2	Gaurav	Kapoor
3	Ravi	Shastri
4	Ajit	Agarkar
5	Sunil	Gavaskar
6	Ian	Bishop
7	Harsha	Bhogle

commentator 69 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
156	23:35:51	select * from matchofficials LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
157	23:36:02	select * from tournaments LIMIT 0, 1000	1 row(s) returned	0.015 sec / 0.000 sec
158	23:36:07	select * from tournamentmatches LIMIT 0, 1000	60 rows(s) returned	0.016 sec / 0.000 sec
159	23:36:19	select * from commentator LIMIT 0, 1000		0.000 sec / 0.000 sec

meet.google.com is sharing your screen. Stop sharing Hide

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Context Help Snippets

ENG 11:36 PM IN 10/28/2023



Founder: Prof. Dr. S. B. Mujumdar, M.Sc.,Ph.D. (Awarded Padma Bhushan and

MySQL Workbench

IPL

```
1082
1083
1084 • select * from innings;
1085
1086
1087
```

InningsID	MatchID	BattingTeamID	BowlingTeamID	TotalRuns	TotalWickets
1	1	1	2	200	4
2	1	1	2	180	6
3	2	8	5	210	2
4	2	5	8	211	5
5	3	3	6	190	3
6	3	6	3	172	10
7	4	7	4	190	3
8	4	4	7	191	3
9	5	5	2	180	3
10	5	2	5	132	9
11	6	6	1	120	3
12	6	1	6	123	0
13	7	4	8	167	3
14	7	8	4	170	6
15	8	3	7	200	6
16	8	7	3	202	5

Output

#	Time	Action	Message	Duration / Fetch
157	23:36:02	select * from tournaments LIMIT 0, 1000	1 row(s) returned	0.015 sec / 0.000 sec
158	23:36:07	select * from tournamentmatches LIMIT 0, 1000	60 row(s) returned	0.016 sec / 0.000 sec
159	23:36:19	select * from commentator LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
160	23:36:29	select * from innings LIMIT 0, 1000	120 row(s) returned	0.000 sec / 0.000 sec

Object Info Session

MySQL Workbench

orangecap

```
1082
1083
1084 • select * from orangecaps;
1085
1086
1087
```

Position	PlayerID	Runs	NumberOfMatches
1	10	600	14
2	7	520	14
3	5	490	14
4	29	450	16
5	34	410	16
*	*	*	*

Output

#	Time	Action	Message	Duration / Fetch
159	23:36:19	select * from commentator LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
160	23:36:29	select * from innings LIMIT 0, 1000	120 row(s) returned	0.000 sec / 0.000 sec
161	23:36:38	select * from manthematch LIMIT 0, 1000	60 rows(s) returned	0.000 sec / 0.000 sec
162	23:36:47	select * from orangecap LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

Object Info Session



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench

IPL*

```
1082
1083
1084 • select * from purplecap;
1085
1086
1087
```

Position	PlayerID	Wickets	NumberOfMatches
1	15	28	16
2	14	24	16
3	35	23	16
4	76	21	14
5	17	20	15
*	NULL	NULL	NULL

purplecap 73

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
160	23:36:29	select * from innings LIMIT 0, 1000	120 row(s) returned	0.000 sec / 0.000 sec
161	23:36:38	select * from manofthematch LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
162	23:36:47	select * from orangecap LIMIT 0, 1000	5 min(s) returned	0.000 sec / 0.000 sec
163	23:36:55	select * from purplecap LIMIT 0, 1000	5 min(s) returned	0.015 sec / 0.000 sec

sponsors74

No object selected

Action Output

#	Time	Action	Message	Duration / Fetch
161	23:36:38	select * from manofthematch LIMIT 0, 1000	60 row(s) returned	0.000 sec / 0.000 sec
162	23:36:47	select * from orangecap LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
163	23:36:55	select * from purplecap LIMIT 0, 1000	5 min(s) returned	0.015 sec / 0.000 sec
164	23:37:04	select * from sponsors LIMIT 0, 1000	5 min(s) returned	0.000 sec / 0.000 sec



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

MySQL Workbench Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information Object Info Session

IPL* 1082 1083 1084 • select * from pointstable; 1085 1086 1087

Result Grid Filter Rows Edit Export/Import Wrap Cell Content Result Grid Form Editor Field Types Query Stats

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Position	TeamID	NumberOfMatches	Points
1	1	14	20
2	4	14	18
3	7	14	16
4	5	14	16
5	3	14	16
6	6	14	14
7	2	14	14
8	8	14	14
MAX	MAX	MAX	MAX

pointstable 75 x Output Action Output # Time Action Message Duration / Fetch 162 23:36:47 select * from orangecap LIMIT 0, 1000 5 row(s) returned 0.000 sec / 0.000 sec 163 23:36:55 select * from purplecap LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec 164 23:37:04 select * from sponsors LIMIT 0, 1000 8 row(s) returned 0.000 sec / 0.000 sec 165 23:37:14 select * from pointatable LIMIT 0, 1000 8 row(s) returned 0.000 sec / 0.000 sec

MySQL Workbench Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information Object Info Session

IPL* 1082 1083 1084 • select * from awards; 1085 1086 1087

Result Grid Filter Rows Edit Export/Import Wrap Cell Content Result Grid Form Editor Field Types Query Stats

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

TournamentID	AwardID	AwardName	PlayerID	Ceremony_Date
1	1	Most Valuable Player	14	2020-06-06
1	2	Orange Cap	10	2020-06-06
1	3	Purple Cap	15	2020-06-06
1	4	Catch Of The Season	64	2020-06-06
1	5	Best Fielder	19	2020-06-06
1	6	Most Sixes	2	2020-06-06
1	7	Best Emerging Player	30	2020-06-06
MAX	MAX	MAX	MAX	MAX

awards 76 x Output Action Output # Time Action Message Duration / Fetch 163 23:36:55 select * from purplecap LIMIT 0, 1000 5 row(s) returned 0.015 sec / 0.000 sec 164 23:37:04 select * from sponsors LIMIT 0, 1000 8 row(s) returned 0.000 sec / 0.000 sec 165 23:37:14 select * from pointatable LIMIT 0, 1000 8 row(s) returned 0.000 sec / 0.000 sec 166 23:37:23 select * from awards LIMIT 0, 1000 8 row(s) returned 0.000 sec / 0.000 sec

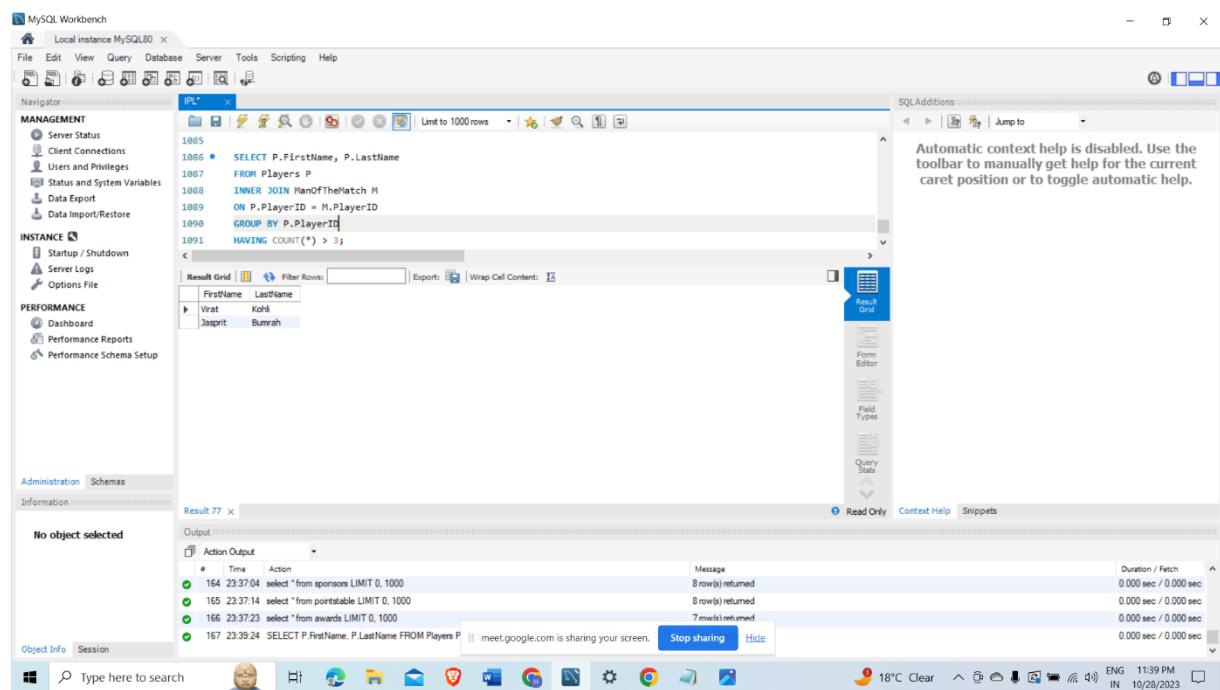


Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

Padma Shri by President of India)

Queries implemented:

Query 1: Find the names of players who have received the "Man of the Match" award more than 3 times.



The screenshot shows the MySQL Workbench interface with the following details:

- File Bar:** File, Edit, View, Query, Database, Server, Tools, Scripting, Help.
- Navigator:** MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup).
- SQL Editor:** Shows the following SQL query:

```
1085
1086 •  SELECT P.FirstName, P.LastName
1087   FROM Players P
1088   INNER JOIN ManOfTheMatch M
1089     ON P.PlayerID = M.PlayerID
1090   GROUP BY P.PlayerID
1091   HAVING COUNT(*) > 3;
```
- Result Grid:** Displays the results of the query:

FirstName	LastName
Virat	Kohli
Jasprit	Bumrah
- Output Window:** Shows the execution log:

#	Time	Action	Message	Duration / Fetch
164	23:37:04	select * from sponsors LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
165	23:37:14	select * from pointstable LIMIT 0, 1000	8 row(s) returned	0.000 sec / 0.000 sec
166	23:37:23	select * from awards LIMIT 0, 1000	7 row(s) returned	0.000 sec / 0.000 sec
167	23:39:24	SELECT P.FirstName, P.LastName FROM Players P	meet.google.com is sharing your screen. Stop sharing Hide	0.000 sec / 0.000 sec
- System Tray:** Shows the date and time as 10/28/2023, 11:39 PM, and the weather as 18°C Clear.



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

Query 2: List the top 2 run-scorers in the OrangeCap table.

```

SELECT P.FirstName, P.LastName, O.Runs
FROM Players P
INNER JOIN OrangeCap O
ON P.PlayerID = O.PlayerID
ORDER BY O.Runs DESC
LIMIT 2;
    
```

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows various database management options like Server Status, Client Connections, and Data Export.
- SQL Editor:** Contains the executed SQL query.
- Result Grid:** Displays the results of the query, showing two rows:

FirstName	LastName	Runs
KL	Rahul	600
Shikhar	Dhawan	520
- Output Window:** Shows the execution log with 4 entries, all completed in 0.000 sec / 0.000 sec.
- Session Bar:** Includes system icons and status information (18°C Clear, ENG 11:40 PM, IN 10/28/2023).

Query 3: Calculate the total number of runs and wickets for each team in a specific tournament.

```

SELECT T.TeamName, SUM(I.TotalRuns) AS TotalRuns, SUM(I.TotalWickets) AS TotalWickets
FROM Teams T
LEFT JOIN Innings I ON T.TeamID = I.BattingTeamID
LEFT JOIN Matches M ON I.MatchID = M.MatchID
LEFT JOIN TournamentMatches TM ON M.MatchID = TM.MatchID
WHERE TM.TournamentID = 1
GROUP BY T.TeamID;
    
```

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows various database management options like Server Status, Client Connections, and Data Export.
- SQL Editor:** Contains the executed SQL query.
- Result Grid:** Displays the results of the query, showing 8 rows:

TeamName	TotalRuns	TotalWickets
Mumbai Indians	2494	68
Chennai Super Kings	2377	72
Kolkata Knight Riders	2443	73
Delhi Capitals	3095	85
Royal Challengers Bangalore	2490	88
Kings XI Punjab	2191	82
Sunrisers Hyderabad	2796	83
Rajasthan Royals	2255	80
- Output Window:** Shows the execution log with 5 entries, all completed in 0.000 sec / 0.000 sec.
- Session Bar:** Includes system icons and status information (18°C Clear, ENG 11:41 PM, IN 10/28/2023).

Query 4: List the top 3 teams with the highest points in the PointsTable.



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Displays a query for selecting team names and points from the Teams and PointsTable tables.
- Result Grid:** Shows the results for the query, listing Mumbai Indians (20), Delhi Capitals (18), and Sunrisers Hyderabad (16).
- Session History:** Shows a list of recent actions, including SELECT statements for players and their statistics.

Query 5: Calculate the average number of overs bowled by players in each role.

The screenshot shows the MySQL Workbench interface with the following details:

- Query Editor:** Displays a query for calculating the average number of overs bowled by players in each role.
- Result Grid:** Shows the results for the query, listing Batman (0.214286), Wicket-Keeper (0.000000), All-Rounder (3.555556), and Bowler (4.000000).
- Session History:** Shows a list of recent actions, including SELECT statements for players and their statistics.



Padma Shri by President of India)

Function:

A function is a set of SQL statements, that you can pass parameters into, and they perform some operation and return a single value.

Functions:

1. Function to print total runs made by a team.

```
CREATE FUNCTION TotalRuns (@TeamID INT)
RETURNS INT
AS
BEGIN
    RETURN (SELECT SUM(Runs) FROM MatchLineups WHERE TeamID = @TeamID)
END
```

2. Function to print total runs made by a team.



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

```

CREATE FUNCTION CalculateTotalWicketsByTeamID(team_id INT)
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE total_wickets INT;
    SELECT sum(TotalWickets)
    INTO total_wickets
    FROM bowlers
    WHERE BowlingTeamID = team_id;
    RETURN total_wickets;
END;
DELIMITER ;
    
```

3. Print Player Age

```

CREATE FUNCTION CalculatePlayerAgeByID(player_id INT)
RETURNS INT
DETERMINISTIC
BEGIN
    DECLARE dob DATE;
    DECLARE age INT;
    -- Get Date of birth
    SELECT DateOfBirth INTO dob FROM Players WHERE PlayerID = player_id;
    -- Calculate age
    SET age = YEAR(CURRENT_DATE()) - YEAR(dob) - (DATE_FORMAT(CURRENT_DATE(), '%Y') < DATE_FORMAT(dob, '%Y')) ;
    RETURN age;
END;
    
```

Padma Shri by President of India)



Founder: Prof. Dr. S. B. Mujumdar, M.Sc., Ph.D. (Awarded Padma Bhushan and

A stored procedure is a set of pre-compiled Structured Query Languages (SQL), so it can be reused and shared by multiple programs. It can access or modify data in a database.

Procedure:

1. Procedure to Increase Team Points by 2.

```

MySQL Workbench - Local instance MySQL80 X
File Edit View Query Database Server Tools Scripting Help
Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information
No object selected
Object Info Session
Type here to search 18°C Clear ENG 11:47 PM IN 10/28/2023

IPL% x SQLAdditions
CREATE PROCEDURE IncreaseTeamPoints(IN teamID INT)
BEGIN
    -- Check if the team exists in PointsTable
    DECLARE teamExists INT;
    SELECT COUNT(*) INTO teamExists FROM PointsTable WHERE TeamID = teamID;

    -- If the team exists, update the points
    IF teamExists > 0 THEN
        UPDATE PointsTable
        SET Points = Points + 2
        WHERE TeamID = teamID;
        SELECT 'Points increased by 2' AS Result;
    ELSE
        SELECT 'Team not found in PointsTable' AS Result;
    END IF;
END
    
```

Result Grid | Filter Rows: [] Export: [] Wrap Cell Content: []

Result

Points increased by 2

Result 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalRunsByTeamID(1) LIMIT 0, 1000	172 23:45:44	1 row(s) returned	0.000 sec / 0.000 sec
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.016 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec

Result 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalRunsByTeamID(1) LIMIT 0, 1000	172 23:45:44	1 row(s) returned	0.000 sec / 0.000 sec
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.016 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec

Result 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalRunsByTeamID(1) LIMIT 0, 1000	172 23:45:44	1 row(s) returned	0.000 sec / 0.000 sec
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.016 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec

2. Procedure to add a venue.

```

MySQL Workbench - Local instance MySQL80 X
File Edit View Query Database Server Tools Scripting Help
Navigator: MANAGEMENT INSTANCE PERFORMANCE Administration Schemas Information
No object selected
Object Info Session
Type here to search 18°C Clear ENG 11:48 PM IN 10/28/2023

IPL% x SQLAdditions
select * from venues;
DELIMITER $$
CREATE PROCEDURE AddVenue1(IN venueID INT, IN venueName VARCHAR(255), IN stadium VARCHAR(255))
BEGIN
    INSERT INTO Venues (VenueID, VenueName, Stadium) VALUES (venueID, venueName, stadium);
END$$
DELIMITER ;
    
```

Result Grid | Filter Rows: [] Export: [] Wrap Cell Content: []

venues

TournamentID	AwardID	AwardName	PlayerID	Ceremony_Date
1	4	Catch Of The Season	64	2020-06-06
1	5	Best Fielder	19	2020-06-06
1	6	Most Sixes	2	2020-06-06
1	7	Best Emerging Player	30	2020-06-06

awards 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.000 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec
select * from awards LIMIT 0, 1000	176 23:48:18	1 row(s) returned	0.000 sec / 0.000 sec

awards 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.000 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec
select * from awards LIMIT 0, 1000	176 23:48:18	1 row(s) returned	0.000 sec / 0.000 sec

awards 85 x

Action Output

Action	Time	Message	Duration / Fetch
select CalculateTotalWicketsByTeamID(2) LIMIT 0, 1000	173 23:45:59	1 row(s) returned	0.000 sec / 0.000 sec
select CalculatePlayerAgeByID (2) LIMIT 0, 1000	174 23:46:39	1 row(s) returned	0.016 sec / 0.000 sec
CALL IncreaseTeamPoints()	175 23:47:34	1 row(s) returned	0.015 sec / 0.000 sec
select * from awards LIMIT 0, 1000	176 23:48:18	1 row(s) returned	0.000 sec / 0.000 sec



Founder: Prof. Dr. S. B. Mujumdar, M.Sc.,Ph.D. (Awarded Padma Bhushan and

Padma Shri by President of India)

Trigger:

A trigger is also a set of SQL statements in the database which automatically execute whenever any special event occurs in the database, like insert, delete, update, etc.

Triggers:

1. After update trigger on Tournaments table:

```

CREATE TRIGGER UpdateTournamentWinner20
AFTER UPDATE ON Matches
FOR EACH ROW
BEGIN
    IF NEW.MatchID = 60 AND NEW.Result IS NOT NULL THEN
        UPDATE Tournaments
        SET Winner = NEW.Result
        WHERE TournamentID = 1; -- Adjust the condition as needed
    END IF;
END$$
DELIMITER ;
UPDATE Tournaments

```

TournamentID	TournamentName	StartDate	EndDate	Winner	WinnerID
1	IPL 2020	2020-09-19	2020-11-30	ME	1

Output:

Action	Time	Action	Message	Duration / Fetch
174	23:46:39	select CalculatePlayerAgeByID (2) LIMIT 0, 1000	1 row(s) returned	0.016 sec / 0.000 sec
175	23:47:34	CALL IncreaseTeamPoints();	1 row(s) returned	0.015 sec / 0.000 sec
176	23:48:18	select * from awards LIMIT 0, 1000	7 rows(s) returned	0.000 sec / 0.000 sec
177	23:49:33	select * from Tournaments LIMIT 0, 1000		0.000 sec / 0.000 sec

References:

Tools: <https://app.diagrams.net/>

Research: <https://www.iplt20.com/>

<https://www.google.com/>