--

CREATE TABLE Board (

BoardID INT PRIMARY KEY,

Name VARCHAR(100),

Address VARCHAR(255),

Contact\_No VARCHAR(15)

);

--

CREATE TABLE Team (

TeamID INT PRIMARY KEY,

TName VARCHAR(100),

Coach VARCHAR(100),

Captain VARCHAR(100),

BoardID INT,

FOREIGN KEY (BoardID) REFERENCES Board(BoardID)

);

--

CREATE TABLE Player (

PlayerID INT PRIMARY KEY,

PFName VARCHAR(100),

PLName VARCHAR(100),

Age INT,

PDateofBirth DATE,

PlayingRole VARCHAR(50),

email VARCHAR(100),

contact\_no VARCHAR(15),

Batting VARCHAR(50),

Bowling VARCHAR(50),

TeamID INT,

FOREIGN KEY (TeamID) REFERENCES Team(TeamID)

);

--

CREATE TABLE Ground (

GroundID INT PRIMARY KEY,

GName VARCHAR(100),

Location VARCHAR(255),

Capacity INT

);

--

CREATE TABLE Umpire (

UmpireID INT PRIMARY KEY,

UFName VARCHAR(100),

ULName VARCHAR(100),

UAge INT,

UDateofBirth DATE,

Country VARCHAR(100),

Uemail VARCHAR(100),

Ucontact\_no VARCHAR(15)

);

--

CREATE TABLE MatchDetails (

MatchID INT PRIMARY KEY,

Match\_Date DATE,

Time1 TIME,

Result VARCHAR(100),

GroundID INT,

UmpireID INT,

FOREIGN KEY (GroundID) REFERENCES Ground(GroundID),

FOREIGN KEY (UmpireID) REFERENCES Umpire(UmpireID)

);

--

CREATE TABLE Player\_Match (

PlayerID INT,

MatchID INT,

PRIMARY KEY (PlayerID, MatchID),

FOREIGN KEY (PlayerID) REFERENCES Player(PlayerID),

FOREIGN KEY (MatchID) REFERENCES MatchDetails(MatchID)

);

Insert Sample Data:

INSERT INTO Board VALUES (1, 'Indian Cricket Board', 'Mumbai, India', '9876543210');

INSERT INTO Team VALUES (101, 'Chennai Super Kings', 'Stephen Fleming', 'MS Dhoni', 1);

INSERT INTO Player VALUES

(1001, 'Virat', 'Kohli', 35, '1990-11-05', 'Batsman', 'virat@gmail.com', '9998887776', 'Right-hand', 'Off-spin', 101);

INSERT INTO Ground VALUES (201, 'Eden Gardens', 'Kolkata', 68000);

INSERT INTO Umpire VALUES (301, 'Richard', 'Kettleborough', 52, '1973-03-15', 'UK', 'richardk@icc.com', '44123456789');

INSERT INTO MatchDetails VALUES (401, '2025-10-09', '18:30:00', 'Team A Won', 201, 301);

INSERT INTO Player\_Match VALUES (1001, 401);

--

SELECT P.PFName, P.PLName, T.TName, B.Name AS BoardName

FROM Player P

JOIN Team T ON P.TeamID = T.TeamID

JOIN Board B ON T.BoardID = B.BoardID;

--

SELECT M.MatchID, M.Match\_Date, G.GName, U.UFName, U.ULName

FROM MatchDetails M

JOIN Ground G ON M.GroundID = G.GroundID

JOIN Umpire U ON M.UmpireID = U.UmpireID;

-- List all players who played in a given match

FROM Player P

JOIN Player\_Match PM ON P.PlayerID = PM.PlayerID

WHERE PM.MatchID = 401;

CREATE TABLE FunctionalDependencies (

Determinant VARCHAR(50),

Dependent VARCHAR(200)

);

INSERT INTO FunctionalDependencies (Determinant, Dependent) VALUES

('BoardID', 'Name, Address, Contact\_No'),

('TeamID', 'TName, Coach, Captain, BoardID'),

('PlayerID', 'PFName, PLName, Age, PDateofBirth, PlayingRole, email, contact\_no, Batting, Bowling, TeamID'),

('MatchID', 'Match\_Date, Time1, Result, GroundID'),

('GroundID', 'GName, Location, Capacity'),

('UmpireID', 'UFName, ULName, UAge, UDateofBirth, Country, Uemail, Ucontact\_no');

Step 2: Verify FD⁺ using Attribute Closure (α⁺)

views (α⁺ sets) for each primary determinant showing what attributes are functionally determined.

--

CREATE VIEW AlphaPlus\_Board AS

SELECT 'BoardID' AS Determinant, 'Name, Address, Contact\_No' AS Closure;

--

CREATE VIEW AlphaPlus\_Team AS

SELECT 'TeamID' AS Determinant, 'TName, Coach, Captain, BoardID' AS Closure;

--

CREATE VIEW AlphaPlus\_Player AS

SELECT 'PlayerID' AS Determinant, 'PFName, PLName, Age, PDateofBirth, PlayingRole, email, contact\_no, Batting, Bowling, TeamID' AS Closure;

--

CREATE VIEW AlphaPlus\_Match AS

SELECT 'MatchID' AS Determinant, 'Match\_Date, Time1, Result, GroundID' AS Closure;

--

CREATE VIEW AlphaPlus\_Ground AS

SELECT 'GroundID' AS Determinant, 'GName, Location, Capacity' AS Closure;

--

CREATE VIEW AlphaPlus\_Umpire AS

SELECT 'UmpireID' AS Determinant, 'UFName, ULName, UAge, UDateofBirth, Country, Uemail, Ucontact\_no' AS Closure;

SELECT \* FROM AlphaPlus\_Board

UNION ALL SELECT \* FROM AlphaPlus\_Team

UNION ALL SELECT \* FROM AlphaPlus\_Player

UNION ALL SELECT \* FROM AlphaPlus\_Match

UNION ALL SELECT \* FROM AlphaPlus\_Ground

UNION ALL SELECT \* FROM AlphaPlus\_Umpire;

Canonical cover (also known as minimal cover) involves:

1. Making each FD have a single attribute on the RHS.

2. Removing redundant attributes on LHS.

3. Removing redundant FDs that can be derived from others.

CREATE TABLE CanonicalCover (

Determinant VARCHAR(50),

Dependent VARCHAR(50)

);

INSERT INTO CanonicalCover (Determinant, Dependent) VALUES

('BoardID', 'Name'),

('BoardID', 'Address'),

('BoardID', 'Contact\_No'),

('TeamID', 'TName'),

('TeamID', 'Coach'),

('TeamID', 'Captain'),

('TeamID', 'BoardID'),

('PlayerID', 'PFName'),

('PlayerID', 'PLName'),

('PlayerID', 'Age'),

('PlayerID', 'PDateofBirth'),

('PlayerID', 'PlayingRole'),

('PlayerID', 'email'),

('PlayerID', 'contact\_no'),

('PlayerID', 'Batting'),

('PlayerID', 'Bowling'),

('PlayerID', 'TeamID'),

('MatchID', 'Match\_Date'),

('MatchID', 'Time1'),

('MatchID', 'Result'),

('MatchID', 'GroundID'),

('GroundID', 'GName'),

('GroundID', 'Location'),

('GroundID', 'Capacity'),

('UmpireID', 'UFName'),

('UmpireID', 'ULName'),

('UmpireID', 'UAge'),

('UmpireID', 'UDateofBirth'),

('UmpireID', 'Country'),

('UmpireID', 'Uemail'),

('UmpireID', 'Ucontact\_no');

SELECT \* FROM CanonicalCover ORDER BY Determinant;

SELECT c.Determinant, c.Dependent

FROM CanonicalCover c

LEFT JOIN FunctionalDependencies f

ON f.Determinant = c.Determinant

WHERE f.Dependent LIKE CONCAT('%', c.Dependent, '%');

Remove partial dependencies by separating entities.

-- Cricket Board

CREATE TABLE CricketBoard\_1NF (

BoardID INT PRIMARY KEY,

Name VARCHAR(100),

Address VARCHAR(150),

Contact\_No VARCHAR(15)

);

-- Team (depends on Board)

CREATE TABLE CricketTeam (

TeamID INT PRIMARY KEY,

TName VARCHAR(100),

Coach VARCHAR(100),

Captain VARCHAR(100),

BoardID INT,

FOREIGN KEY (BoardID) REFERENCES CricketBoard\_1NF(BoardID)

);

-- Player (depends on Team)

CREATE TABLE CricketPlayer (

PlayerID INT PRIMARY KEY,

PFName VARCHAR(100),

PLName VARCHAR(100),

Age INT,

PDateofBirth DATE,

PlayingRole VARCHAR(50),

email VARCHAR(100),

contact\_no VARCHAR(15),

Batting VARCHAR(50),

Bowling VARCHAR(50),

TeamID INT,

FOREIGN KEY (TeamID) REFERENCES CricketTeam(TeamID)

);

-- Ground

CREATE TABLE CricketGround (

GroundID INT PRIMARY KEY,

GName VARCHAR(100),

Location VARCHAR(100),

Capacity INT

);

-- Match (depends on Ground)

CREATE TABLE CricketMatch (

MatchID INT PRIMARY KEY,

Match\_Date DATE,

Time1 TIME,

Result VARCHAR(50),

GroundID INT,

FOREIGN KEY (GroundID) REFERENCES CricketGround(GroundID)

);

-- Umpire

CREATE TABLE CricketUmpire (

UmpireID INT PRIMARY KEY,

UFName VARCHAR(100),

ULName VARCHAR(100),

UAge INT,

UDateofBirth DATE,

Country VARCHAR(50),

Uemail VARCHAR(100),

Ucontact\_no VARCHAR(15)

);

Remove transitive dependencies (e.g., TeamID → BoardID → Board attributes)

Already separated → 3NF achieved.

Add associative (relationship) tables for many-to-many relationships.

-- Match–Umpire (many-to-many)

CREATE TABLE MatchUmpire (

MatchID INT,

UmpireID INT,

PRIMARY KEY (MatchID, UmpireID),

FOREIGN KEY (MatchID) REFERENCES CricketMatch(MatchID),

FOREIGN KEY (UmpireID) REFERENCES CricketUmpire(UmpireID)

);

-- Player–Match (many-to-many)

CREATE TABLE PlayerMatch (

PlayerID INT,

MatchID INT,

PRIMARY KEY (PlayerID, MatchID),

FOREIGN KEY (PlayerID) REFERENCES CricketPlayer(PlayerID),

FOREIGN KEY (MatchID) REFERENCES CricketMatch(MatchID)

);

--

SELECT b.BoardID, b.Name, t.TeamID, t.TName

FROM CricketBoard\_1NF b

JOIN CricketTeam t ON b.BoardID = t.BoardID;

--

SELECT p.PlayerID, p.PFName, t.TeamID, t.TName

FROM CricketPlayer p

JOIN CricketTeam t ON p.TeamID = t.TeamID;

--

SELECT m.MatchID, m.Match\_Date, g.GName, u.UFName

FROM CricketMatch m

JOIN CricketGround g ON m.GroundID = g.GroundID

JOIN MatchUmpire mu ON m.MatchID = mu.MatchID

JOIN CricketUmpire u ON mu.UmpireID = u.UmpireID;