DATABASE MANAGEMENT SYSTEM PROJECT

TOPIC: IPL DATABASE MANAGEMENT SYSTEM

Submitted to: Mr. Anand Gupta

Submitted By:

Group 1

Ishani – 2022UIC3591

Ankit Kumar Sharma – 2022UIC3592

Mohammad Asad – 2022UIC3593

Aditya Gupta – 2022UIC3594

CONTENTS

PROBLEM STATEMENT

Design a relational database schema for managing information related to cricket matches, teams, head coaches, and umpires. The database should facilitate tracking details of matches played, team compositions, umpire assignments, and coaching staff.

The database should allow the following functionalities:
- Store information about head coaches including their contact details and team assignments.
- Maintain details of cricket teams such as team name, captain, coach, etc.
- Establish relationships between matches and umpires indicating which umpires officiated each match.
- Store details of matches played including the date, venue, and match outcome.

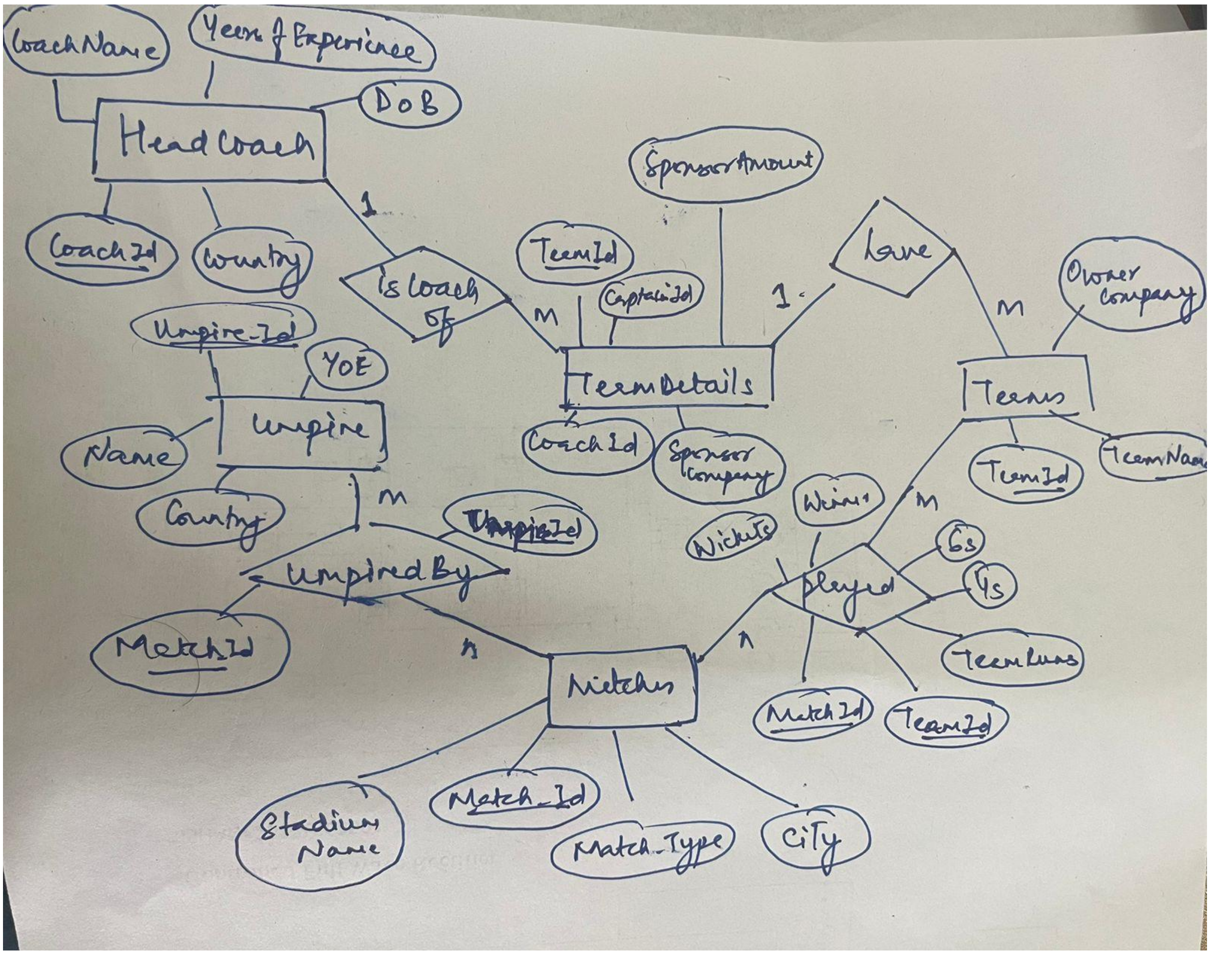
Design a relational database schema that efficiently organizes and manages these entities to support various queries and data retrieval operations effectively.

ER MODEL

An entity-relationship (ER) model is a conceptual data model used in database management systems (DBMS) to describe the relationships between entities and their attributes. The ER model uses graphical notations to represent entities, relationships, and their attributes, which helps in visualizing the structure of the database.

- In an ER model, an entity is a real-world object or concept, such as a player, match, or team. Each entity has attributes that describe its characteristics or properties.
- Entities are connected through relationships, which represent the associations between entities. Relationships can be one-to-one, one-to-many, or many-to-many, depending on the number of entities involved.
- The ER model also includes cardinality and participation constraints, which specify the minimum and maximum number of entities that can be involved in a relationship.
- Overall, the ER model provides a clear and concise way to represent the structure of a database and its relationships, which helps in the design and development of the database management system.

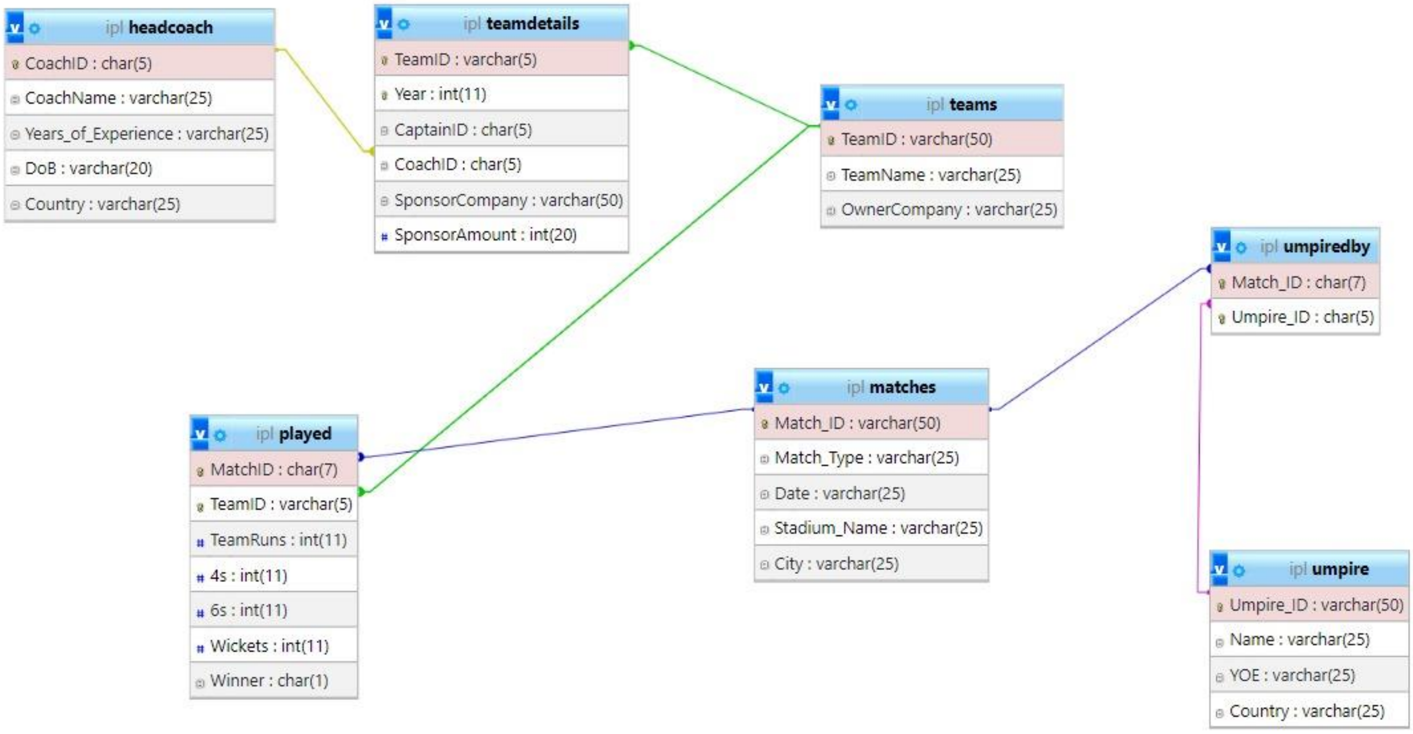
ER DIAGRAM


CoachName
Year of Experince
DoB
Head Coach
1
CoachId
Country
Is Coach Of
M
Umpire-Id
YOE
Umpire
Name
Country
m
UmpireId
Umpired By
MatchId
n
SponsorAmount
TeamId
CaptainId
1
have
M
Owner Company
TeamDetails
CoachId
Sponsor Company
Teams
TeamId
TeamName
M
Wins
Wickets
6s
4s
Played
n
TeamRuns
MatchId
TeamId
Matches
Stadium Name
Match_Id
Match_Type
City

RELATIONAL MODEL

Relations:

1. TeamDetails (TeamID, CaptainID, CoachID, SponsorCompany, SponsorAmount)
2. Played (MatchID, TeamID, Team Runs, 4s, 6s, Wickets, Winner)
3. Umpire (Umpire_ID, Name, YOE, Country)
4. UmpiredBY (Match_ID, Umpire_ID)
5. Teams (TeamID, TeamName, OwnerCompany)
6. Matches (Match_ID, Match_Type, Date, Stadium_Name, City, TeamID)
7. HeadCoach (CoachID, CoachName, Years_of_Experience, DoB, Country)



FUNCTIONAL DEPENDENCIES

1. Table { Matches)

MatchID → {MatchType, Date, Stadium_Name, City, Team_ID}

2. Table (Played)

MatchID ,TeamID →{TeamRuns, 4s, 6s, Wickets, Winner}

MatchID → Winner

TeamID → {TeamRuns, 4s, 6s, Wickets}

3. Table (HeadCoach)

Coach ID → {CoachName, Years-of-Experience, DOB, Country)

~~Table~~

4. Table (TeamDetails)

TeamID, Year → (CaptainID, CoachID, SponsorCompany, SponsorAmount}

CoachID → {CoachName, Years-of-Experience, DOB, Country}

TeamID → {TeamName, HomeGround, Owner}

5. Table (Teams)

TeamID → {TeamName, OwnerCompany}

~~Fa~~

6. Table (Umpire)

UmpireID → (Name, YOE, Country)

7. Table (UmpiredBy)

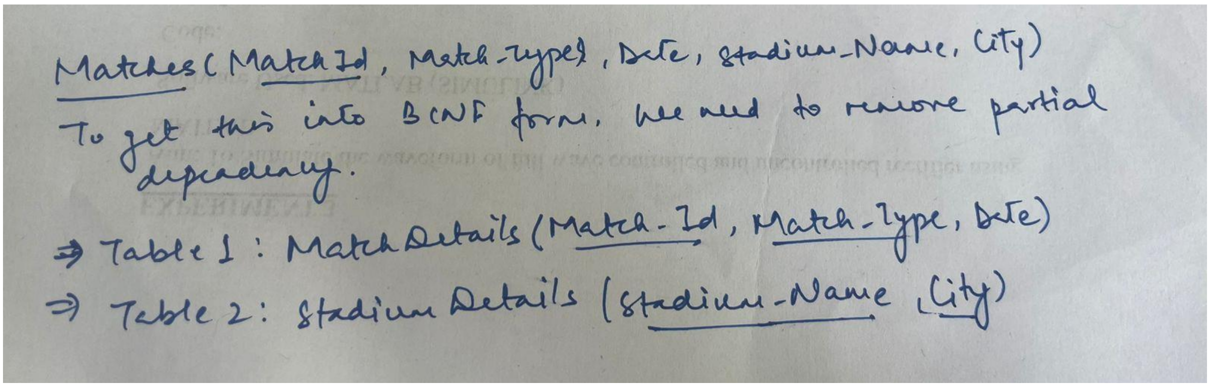
MatchID → Umpire_ID

Umpire_ID → Match_ID

NORMALIZATION IN BCNF

To be in BCNF, a relation (table) must meet two conditions:

1. Every determinant (attributes determining other attributes) is a candidate key.
2. Every non-trivial functional dependency (where the determined attributes are not part of the candidate key) is based on a superkey.

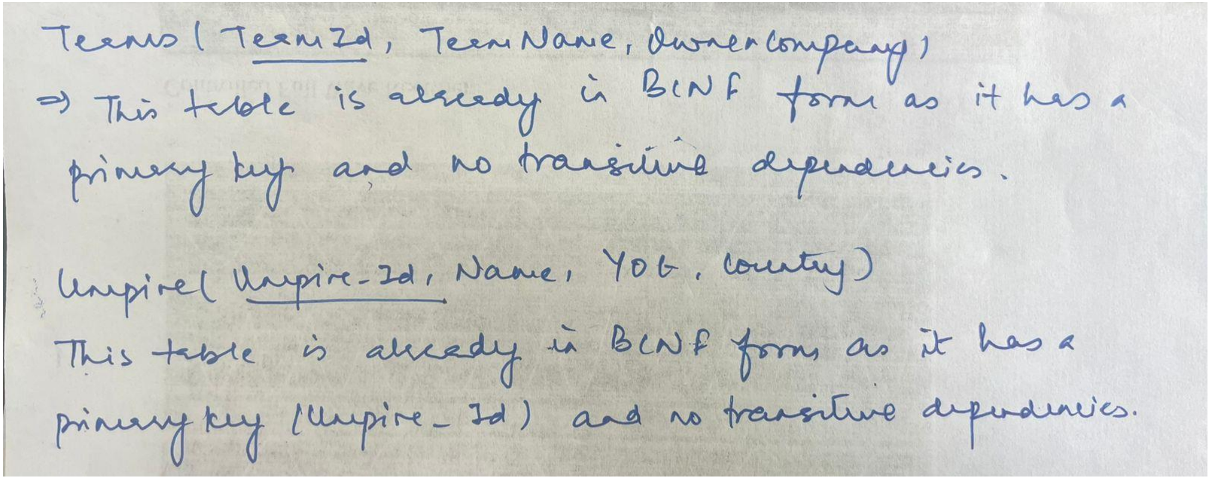


Matches(MatchId, Match-Type, Date, Stadium_Name, City)

To get this into BCNF form, we need to remove partial dependency.

⇒ Table 1 : MatchDetails (Match-Id, Match-Type, Date)

⇒ Table 2: Stadium Details (Stadium-Name, City)

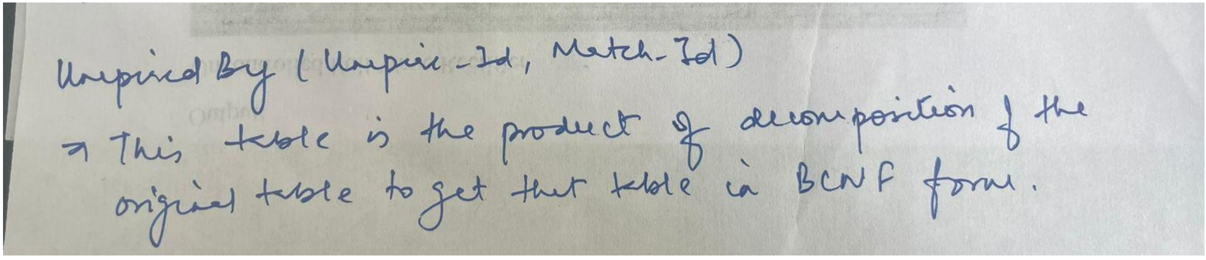


Teams (TeamId, TeamName, OwnerCompany)

⇒ This table is already in BCNF form as it has a primary key and no transitive dependencies.

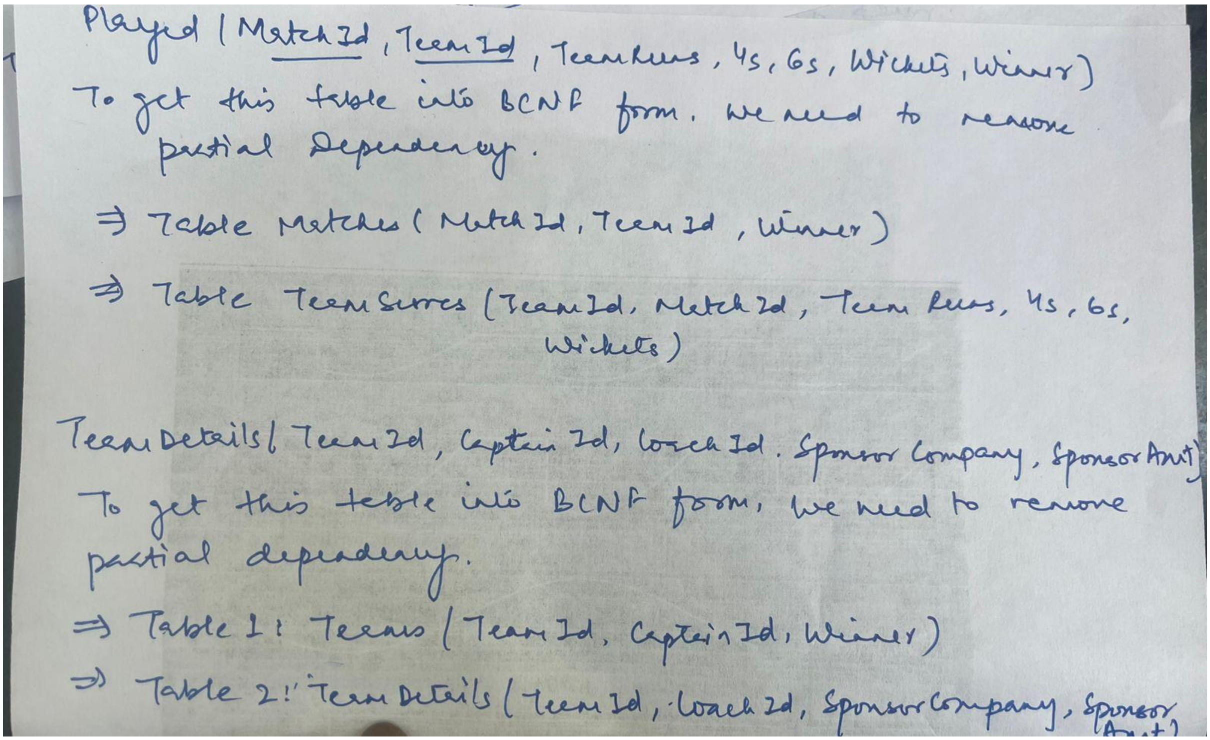
Umpire(Umpire-Id, Name, YOB, Country)

This table is already in BCNF form as it has a primary key (Umpire_Id) and no transitive dependencies.



Umpired By (Umpire-Id, Match-Id)

→ This table is the product of decomposition of the original table to get that table in BCNF form.



Played (Match Id , Team Id , Team Runs , 4s , 6s , Wickets , Winner)

To get this table into BCNF form. We need to remove partial Dependency.

⇒ Table Matches (Match Id , Team Id , Winner)

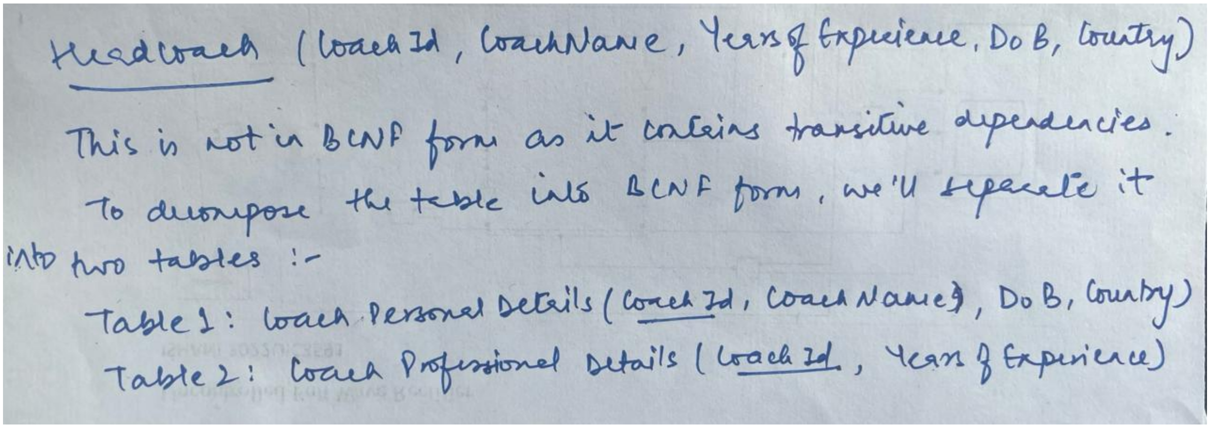
⇒ Table Team Scores (Team Id, Match Id , Team Runs, 4s , 6s, Wickets)

Team Details (Team Id, Captain Id, Coach Id. Sponsor Company, Sponsor Amt)

To get this table into BCNF form, we need to remove partial dependency.

⇒ Table 1: Teams (Team Id, Captain Id, Winner)

⇒ Table 2: Team Details (Team Id, Coach Id, Sponsor Company, Sponsor Amt)



HeadCoach (Coach Id, CoachName, Years of Experience, DoB, Country)

This is not in BCNF form as it contains transitive dependencies.

To decompose the table into BCNF form, we'll separate it into two tables :-

Table 1: Coach Personal Details (Coach Id, Coach Name), DoB, Country)

Table 2: Coach Professional Details (Coach Id, Years of Experience)

XAMPP IMPLEMENTATION

1. CONNECTION OF XAMPP MYSQL AND PYTHON

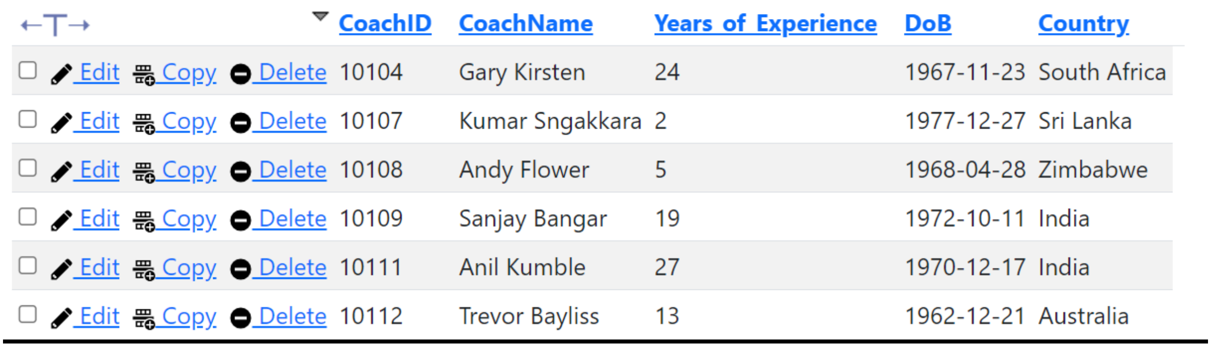
```
import mysql.connector
mydb = mysql.connector.connect(host='localhost', user='root',
password='', database='IPL')
mycursor = mydb.cursor()
```

2. TABLE CREATION AND DATA INSERTION

Table: HeadCoach

Code:

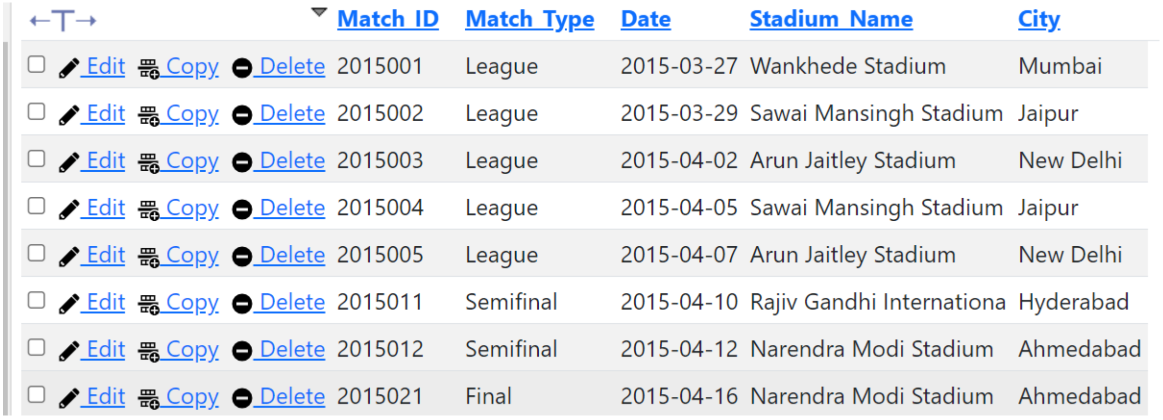
```
import mysql.connector

mydb = mysql.connector.connect(host='localhost', user='root', password='',
database='practice')
mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE HeadCoach ( CoachID CHAR(5) PRIMARY KEY,
CoachName VARCHAR(25), "
                 "Years_of_Experience VARCHAR(25), DoB VARCHAR(20), Country
VARCHAR(25))")

sql = "INSERT INTO HeadCoach (CoachID, CoachName, Years_of_Experience, DoB,
Country)\
    VALUES (%s, %s, %s, %s, %s)"

val = [("10104","Gary Kirsten",24,"1977-12-27","South Africa"),
       ("10107","Kumar Sngakkara",2,"1977-12-27","Sri Lanka"),
       ("10108","Andy Flower",5,"1968-04-28","Zimbabwe"),
       ("10109","Sanjay Bangar",19,"1972-10-11","India"),
       ("10111","Anil Kumble",27,"1970-12-17","India"),
       ("10112","Trevor Bayliss",13,"1962-12-21","Australia")]
mycursor.executemany(sql, val)
mydb.commit()
mydb.close()
```



←T→	CoachID	CoachName	Years_of_Experience	DoB	Country
☐ ✎ Edit Copy ⊖ Delete	10104	Gary Kirsten	24	1967-11-23	South Africa
☐ ✎ Edit Copy ⊖ Delete	10107	Kumar Sngakkara	2	1977-12-27	Sri Lanka
☐ ✎ Edit Copy ⊖ Delete	10108	Andy Flower	5	1968-04-28	Zimbabwe
☐ ✎ Edit Copy ⊖ Delete	10109	Sanjay Bangar	19	1972-10-11	India
☐ ✎ Edit Copy ⊖ Delete	10111	Anil Kumble	27	1970-12-17	India
☐ ✎ Edit Copy ⊖ Delete	10112	Trevor Bayliss	13	1962-12-21	Australia

Table: Matches

```
import mysql.connector

mydb = mysql.connector.connect(host='localhost', user='root', password='',
database='practice')
mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE Matches ( Match_ID VARCHAR(50) PRIMARY KEY,
"
                 "Match_Type VARCHAR(25), Date VARCHAR(25),\
                 Stadium_Name VARCHAR(25),City VARCHAR(25),Team_ID
VARCHAR(25))")
sql = ("INSERT INTO Matches
(Match_ID,Match_Type,Date,Stadium_Name,City,Team_ID)\
    VALUES (%s, %s, %s, %s, %s, %s)")

val = [("2015001","League","2015-03-27","Wankhede
Stadium","Mumbai","00012"),
("2015002","League","2015-03-29","Sawai Mansingh
Stadium","Jaipur","00057"),
("2015003","League","2015-04-02","Arun Jaitley Stadium","New
Delhi","00042"),
("2015004","League","2015-04-05","Sawai Mansingh
Stadium","Jaipur","00020"),
("2015005","League","2015-04-07","Arun Jaitley Stadium","New
Delhi","00029"),
("2015011","Semifinal","2015-04-10","Rajiv Gandhi International Cricket
Stadium","Hyderabad","00040"),
("2015012","Semifinal","2015-04-12","Narendra Modi
Stadium","Ahmedabad","00012"),
("2015021","Final","2015-04-16","Narendra Modi
Stadium","Ahmedabad","00038")]
mycursor.executemany(sql, val)
mydb.commit()
mydb.close()
```

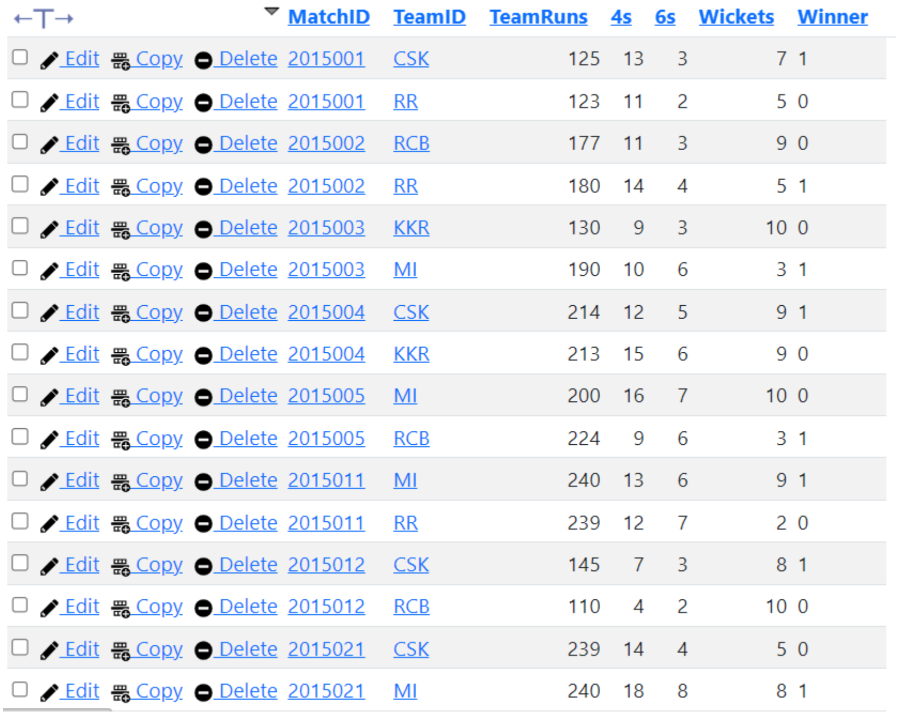


←T→	▼	Match ID	Match Type	Date	Stadium Name	City
□	Edit Copy Delete	2015001	League	2015-03-27	Wankhede Stadium	Mumbai
□	Edit Copy Delete	2015002	League	2015-03-29	Sawai Mansingh Stadium	Jaipur
□	Edit Copy Delete	2015003	League	2015-04-02	Arun Jaitley Stadium	New Delhi
□	Edit Copy Delete	2015004	League	2015-04-05	Sawai Mansingh Stadium	Jaipur
□	Edit Copy Delete	2015005	League	2015-04-07	Arun Jaitley Stadium	New Delhi
□	Edit Copy Delete	2015011	Semifinal	2015-04-10	Rajiv Gandhi Internationa	Hyderabad
□	Edit Copy Delete	2015012	Semifinal	2015-04-12	Narendra Modi Stadium	Ahmedabad
□	Edit Copy Delete	2015021	Final	2015-04-16	Narendra Modi Stadium	Ahmedabad

Table: Played

Code:

```
import mysql.connector
mydb = mysql.connector.connect(host='localhost', user='root', password='',
database='practice')
cursor = mydb.cursor()
cursor.execute('''
CREATE TABLE Played(
    MatchID CHAR(7),TeamID VARCHAR(5),TeamRuns INT NOT NULL,4s INT NOT
NULL,
    6s INT NOT NULL,Wickets INT NOT NULL,Winner CHAR(1) NOT NULL,
    PRIMARY KEY(MatchID,TeamID))''')
cursor.execute('''
ALTER TABLE Played ADD  FOREIGN KEY (TeamID)
REFERENCES Teams (TeamID) ON DELETE CASCADE ON UPDATE CASCADE''')
cursor.execute('''
ALTER TABLE Played ADD FOREIGN KEY (MatchID)
REFERENCES Matches (Match_ID) ON DELETE CASCADE ON UPDATE CASCADE''')
sql = """ INSERT INTO Played (MatchID,TeamID,TeamRuns,4s,6s,Wickets,Winner)
VALUES (%s,%s,%s,%s,%s,%s,%s)"""

values = [('2015001','RR',123,11,2,5,'0'),
('2015001','CSK',125,13,3,7,'1'),
('2015002','RR',180,14,4,5,'1'),
('2015002','RCB',177,11,3,9,'0'),
('2015003','KKR',130,9,3,10,'0'),
('2015003','MI',190,10,6,3,'1'),
('2015004','KKR',213,15,6,9,'0'),
('2015004','CSK',214,12,5,9,'1'),
('2015005','MI',200,16,7,10,'0'),
('2015005','RCB',224,9,6,3,'1'),
('2015011','MI',240,13,6,9,'1'),
('2015011','RR',239,12,7,2,'0'),
('2015012','CSK',145,7,3,8,'1'),
('2015012','RCB',110,4,2,10,'0'),
('2015021','MI',240,18,8,8,'1'),
('2015021','CSK',239,14,4,5,'0')
]
cursor.executemany(sql, values)
mydb.commit()
cursor.close()
```

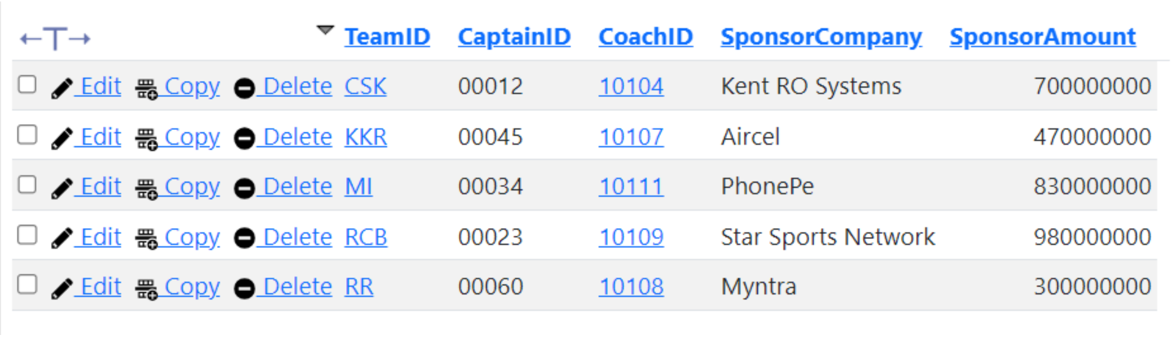


←T→				MatchID	TeamID	TeamRuns	4s	6s	Wickets	Winner
□	Edit	Copy	Delete	2015001	CSK	125	13	3	7	1
□	Edit	Copy	Delete	2015001	RR	123	11	2	5	0
□	Edit	Copy	Delete	2015002	RCB	177	11	3	9	0
□	Edit	Copy	Delete	2015002	RR	180	14	4	5	1
□	Edit	Copy	Delete	2015003	KKR	130	9	3	10	0
□	Edit	Copy	Delete	2015003	MI	190	10	6	3	1
□	Edit	Copy	Delete	2015004	CSK	214	12	5	9	1
□	Edit	Copy	Delete	2015004	KKR	213	15	6	9	0
□	Edit	Copy	Delete	2015005	MI	200	16	7	10	0
□	Edit	Copy	Delete	2015005	RCB	224	9	6	3	1
□	Edit	Copy	Delete	2015011	MI	240	13	6	9	1
□	Edit	Copy	Delete	2015011	RR	239	12	7	2	0
□	Edit	Copy	Delete	2015012	CSK	145	7	3	8	1
□	Edit	Copy	Delete	2015012	RCB	110	4	2	10	0
□	Edit	Copy	Delete	2015021	CSK	239	14	4	5	0
□	Edit	Copy	Delete	2015021	MI	240	18	8	8	1

Table: TeamDetails

Code:

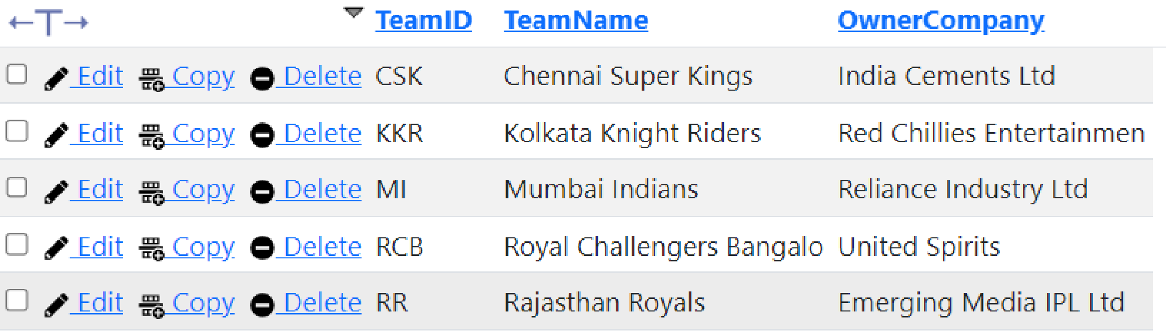
```
import mysql.connector as c
mydb = c.connect(
 host="localhost",
 user="root",
 password="",
 database="practice"
)
cursor = mydb.cursor()

cursor.execute('''
CREATE TABLE TeamDetails(
TeamID VARCHAR(5),CaptainID CHAR(5) NOT NULL,
CoachID CHAR(5) NOT NULL, SponsorCompany VARCHAR(50) NOT NULL,SponsorAmount
INT(20) NOT NULL,   PRIMARY KEY (TeamID))''')
cursor.execute('''
ALTER TABLE TeamDetails ADD FOREIGN KEY (CoachID) REFERENCES HeadCoach
(coachID) ON DELETE CASCADE ON UPDATE CASCADE''')
cursor.execute('''
ALTER TABLE TeamDetails ADD FOREIGN KEY (TeamID) REFERENCES Teams (TeamID)
ON DELETE CASCADE ON UPDATE CASCADE''')
sql = """ INSERT INTO TeamDetails (TeamID,CaptainID, CoachID,
SponsorCompany, SponsorAmount)
VALUES (%s,%s, %s, %s, %s)"""

value = [
('RCB','00023','10109','Star Sports Network',980000000),
('CSK','00012','10104','Kent RO Systems',700000000),
('RR','00060','10108','Myntra ',300000000),
('MI','00034','10111','PhonePe',830000000),
('KKR','00045','10107','Aircel',470000000)
]

cursor.executemany(sql, value)
mydb.commit()
cursor.close()
```



←T→		TeamID	CaptainID	CoachID	SponsorCompany	SponsorAmount
□	Edit Copy Delete	CSK	00012	10104	Kent RO Systems	700000000
□	Edit Copy Delete	KKR	00045	10107	Aircel	470000000
□	Edit Copy Delete	MI	00034	10111	PhonePe	830000000
□	Edit Copy Delete	RCB	00023	10109	Star Sports Network	980000000
□	Edit Copy Delete	RR	00060	10108	Myntra	300000000

Table: Teams

```
import mysql.connector

mydb = mysql.connector.connect(host='localhost', user='root', password='',
database='practice')
mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE Teams ( TeamID VARCHAR(50) PRIMARY KEY,
TeamName VARCHAR(25), OwnerCompany VARCHAR(25))")
sql = "INSERT INTO Teams (TeamID, TeamName, OwnerCompany)\
    VALUES (%s, %s, %s)"

val = [("RCB","Royal Challengers Bangalore","United Spirits"),
("MI","Mumbai Indians","Reliance Industry Ltd"),
("CSK","Chennai Super Kings","India Cements Ltd"),
("RR","Rajasthan Royals","Emerging Media IPL Ltd"),
("KKR","Kolkata Knight Riders","Red Chillies Entertainment")]
mycursor.executemany(sql, val)
mydb.commit()
mydb.close()
```

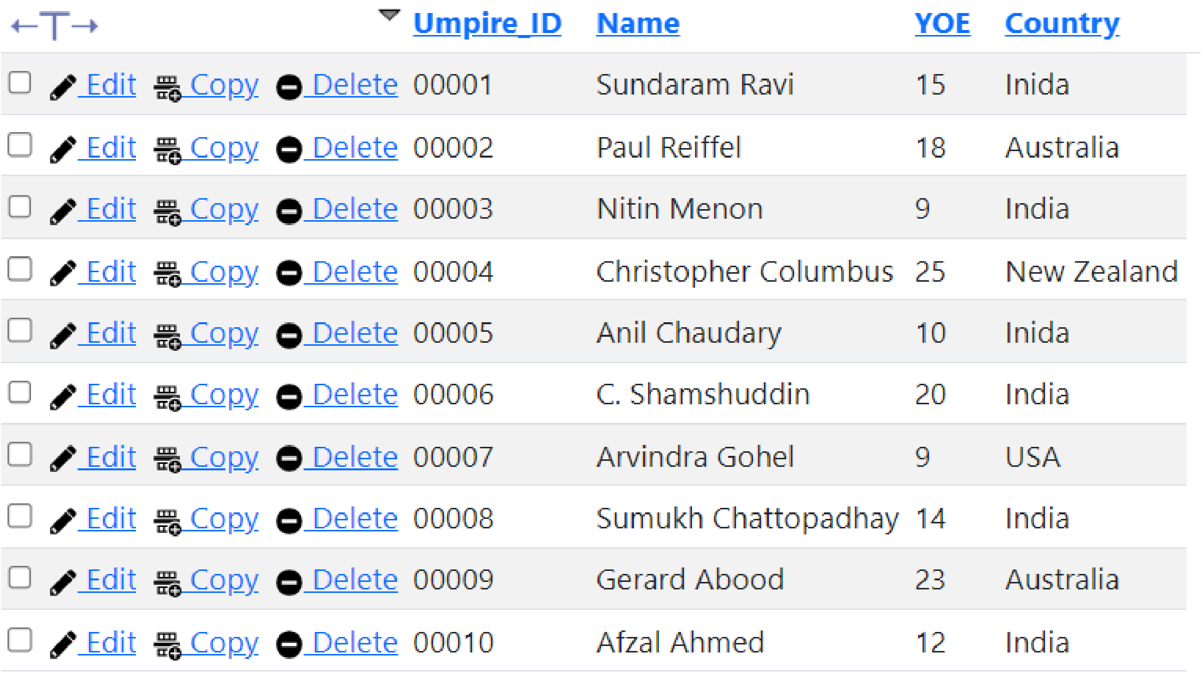


←T→	TeamID	TeamName	OwnerCompany
□ Edit Copy Delete	CSK	Chennai Super Kings	India Cements Ltd
□ Edit Copy Delete	KKR	Kolkata Knight Riders	Red Chillies Entertainmen
□ Edit Copy Delete	MI	Mumbai Indians	Reliance Industry Ltd
□ Edit Copy Delete	RCB	Royal Challengers Bangalo	United Spirits
□ Edit Copy Delete	RR	Rajasthan Royals	Emerging Media IPL Ltd

Table: Umpire

Code:

```
import mysql.connector

mydb = mysql.connector.connect(host='localhost', user='root', password='',
database='practice')
mycursor = mydb.cursor()

mycursor.execute("CREATE TABLE Umpire ( Umpire_ID VARCHAR(50) PRIMARY KEY,
Name VARCHAR(25), YOE VARCHAR(25),\
                 Country VARCHAR(25))")
sql = "INSERT INTO Umpire (Umpire_ID,Name,YOE,Country)\
    VALUES (%s, %s, %s, %s)"


val = [("00001","Sundaram Ravi","15","Inida"),
("00002","Paul Reiffel","18","Australia"),
("00003","Nitin Menon","9","India"),
("00004","Christopher Columbus","25","New Zealand"),
("00005","Anil Chaudary","10","Inida"),
("00006","C. Shamshuddin","20","India"),
("00007","Arvindra Gohel","9","USA"),
("00008","Sumukh Chattopadhay","14","India"),
("00009","Gerard Abood","23","Australia"),
("00010","Afzal Ahmed","12","India")]
mycursor.executemany(sql, val)
mydb.commit()
mydb.close()
```

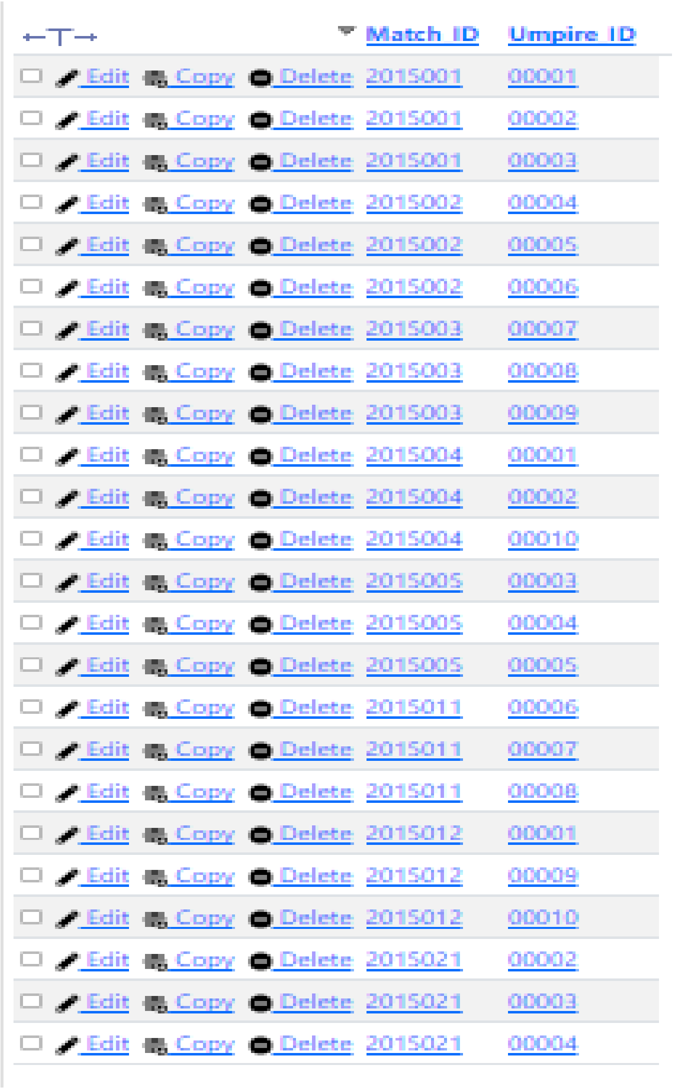


←T→				▾ Umpire_ID	Name	YOE	Country
□	Edit	Copy	⊖ Delete	00001	Sundaram Ravi	15	Inida
□	Edit	Copy	⊖ Delete	00002	Paul Reiffel	18	Australia
□	Edit	Copy	⊖ Delete	00003	Nitin Menon	9	India
□	Edit	Copy	⊖ Delete	00004	Christopher Columbus	25	New Zealand
□	Edit	Copy	⊖ Delete	00005	Anil Chaudary	10	Inida
□	Edit	Copy	⊖ Delete	00006	C. Shamshuddin	20	India
□	Edit	Copy	⊖ Delete	00007	Arvindra Gohel	9	USA
□	Edit	Copy	⊖ Delete	00008	Sumukh Chattopadhay	14	India
□	Edit	Copy	⊖ Delete	00009	Gerard Abood	23	Australia
□	Edit	Copy	⊖ Delete	00010	Afzal Ahmed	12	India

Table: UmpiredBy

Code:

```
import mysql.connector as c

mydb = c.connect(
    host="localhost",
    user="root",
    password="",
    database="practice"
)
cursor = mydb.cursor()

cursor.execute('''
CREATE TABLE UmpiredBy(
    Match_ID CHAR(7),
    Umpire_ID CHAR(5),
    PRIMARY KEY (Match_ID, Umpire_ID))''')

cursor.execute('''
ALTER TABLE UmpiredBy ADD FOREIGN KEY (Match_ID) REFERENCES
Matches(Match_ID) ON DELETE CASCADE ON UPDATE CASCADE''')
cursor.execute('''
ALTER TABLE UmpiredBy ADD FOREIGN KEY (Umpire_ID) REFERENCES
Umpire(Umpire_ID) ON DELETE CASCADE ON UPDATE CASCADE''')

sql = """ INSERT INTO UmpiredBy (Match_ID, Umpire_ID)
VALUES (%s,%s)"""

values = [('2015001','00001'),('2015001','00002'),('2015001','00003'),
('2015002','00004'),('2015002','00005'),('2015002','00006'),
('2015003','00007'),('2015003','00008'),('2015003','00009'),
('2015004','00010')('2015004','00001'),('2015004','00002'),
('2015005','00003'),('2015005','00004'),
('2015005','00005'),('2015011','00006'),('2015011','00007'),
('2015011','00008'),('2015012','00009'),
('2015012','00010'),('2015012','00001'),
('2015021','00002'),('2015021','00003'),
('2015021','00004')]

cursor.executemany(sql, values)
mydb.commit()
cursor.close()
```



←T→				Match ID	Umpire ID
□	Edit	Copy	Delete	2015001	00001
□	Edit	Copy	Delete	2015001	00002
□	Edit	Copy	Delete	2015001	00003
□	Edit	Copy	Delete	2015002	00004
□	Edit	Copy	Delete	2015002	00005
□	Edit	Copy	Delete	2015002	00006
□	Edit	Copy	Delete	2015003	00007
□	Edit	Copy	Delete	2015003	00008
□	Edit	Copy	Delete	2015003	00009
□	Edit	Copy	Delete	2015004	00001
□	Edit	Copy	Delete	2015004	00002
□	Edit	Copy	Delete	2015004	00010
□	Edit	Copy	Delete	2015005	00003
□	Edit	Copy	Delete	2015005	00004
□	Edit	Copy	Delete	2015005	00005
□	Edit	Copy	Delete	2015011	00006
□	Edit	Copy	Delete	2015011	00007
□	Edit	Copy	Delete	2015011	00008
□	Edit	Copy	Delete	2015012	00001
□	Edit	Copy	Delete	2015012	00009
□	Edit	Copy	Delete	2015012	00010
□	Edit	Copy	Delete	2015021	00002
□	Edit	Copy	Delete	2015021	00003
□	Edit	Copy	Delete	2015021	00004

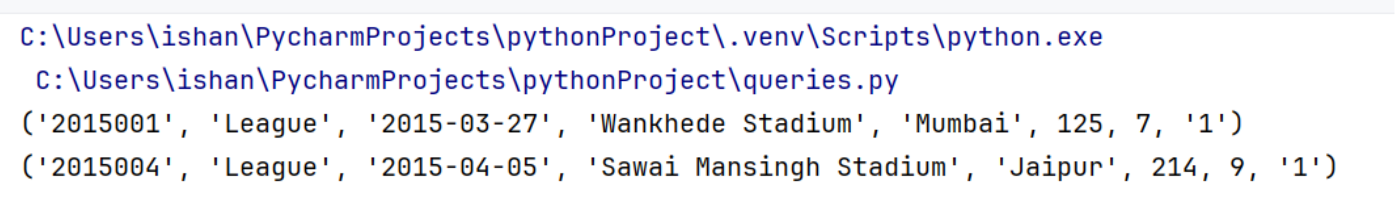
3.Queries in XAMPP

1. In which League Match did CSK win?

Code:

```
cursor.execute('''
SELECT m.Match_ID, m.Match_Type, m.Date, m.Stadium_Name, m.City, p.TeamRuns, p.Wickets,
p.Winner FROM Matches m JOIN Played p ON m.Match_ID = p.MatchID
WHERE m.Match_Type = 'League' AND p.TeamID = 'CSK' AND p.Winner = '1'''')
results = cursor.fetchall()
for row in results:
    type_fixed_row = tuple([el.decode('utf-8') if type(el) is bytearray else el for el
in row])
    print(type_fixed_row)
```

Output:

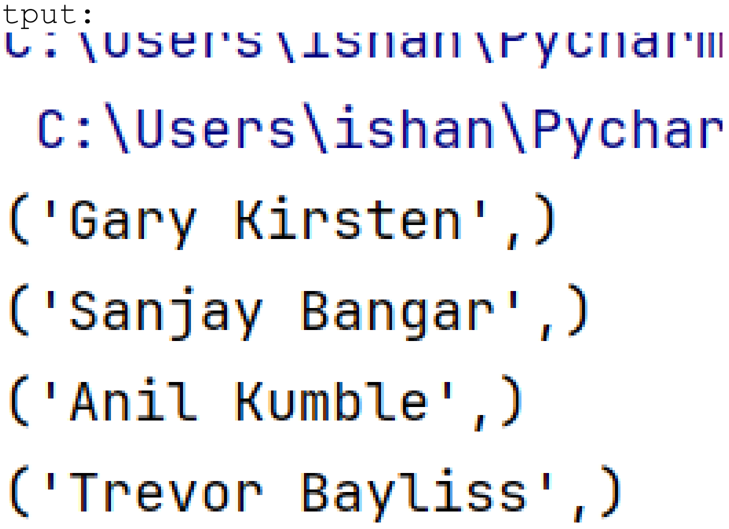


```
C:\Users\ishan\PycharmProjects\pythonProject\.venv\Scripts\python.exe
 C:\Users\ishan\PycharmProjects\pythonProject\queries.py
('2015001', 'League', '2015-03-27', 'Wankhede Stadium', 'Mumbai', 125, 7, '1')
('2015004', 'League', '2015-04-05', 'Sawai Mansingh Stadium', 'Jaipur', 214, 9, '1')
```

2 Find the names of all coaches who have more than 10 years of experience

```
cursor.execute(''' SELECT CoachName
FROM HeadCoach
WHERE Years_of_Experience > 10’’’)
```

Output:



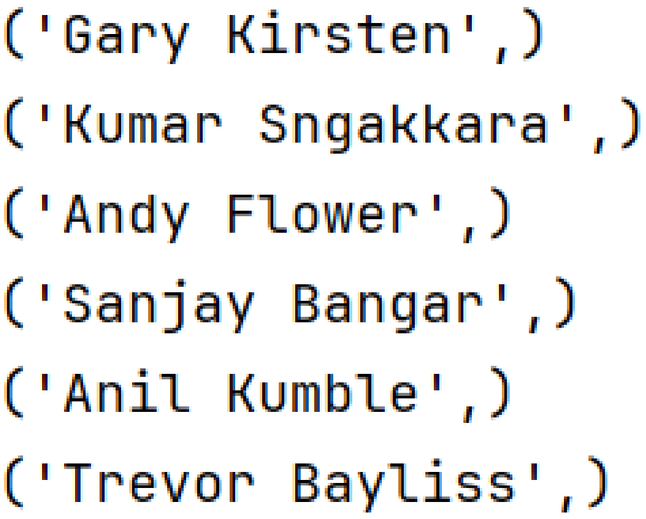
```
C:\Users\ishan\Pycharm
 C:\Users\ishan\Pychar
('Gary Kirsten',)
('Sanjay Bangar',)
('Anil Kumble',)
('Trevor Bayliss',)
```

3. Find the names of all coaches who have coached teams that played in the final match

Code:

```
cursor.execute('''
SELECT DISTINCT CoachName
FROM HeadCoach hc
JOIN Teams t
JOIN Matches m
WHERE m.Match_Type = 'Final'
’’’)
```

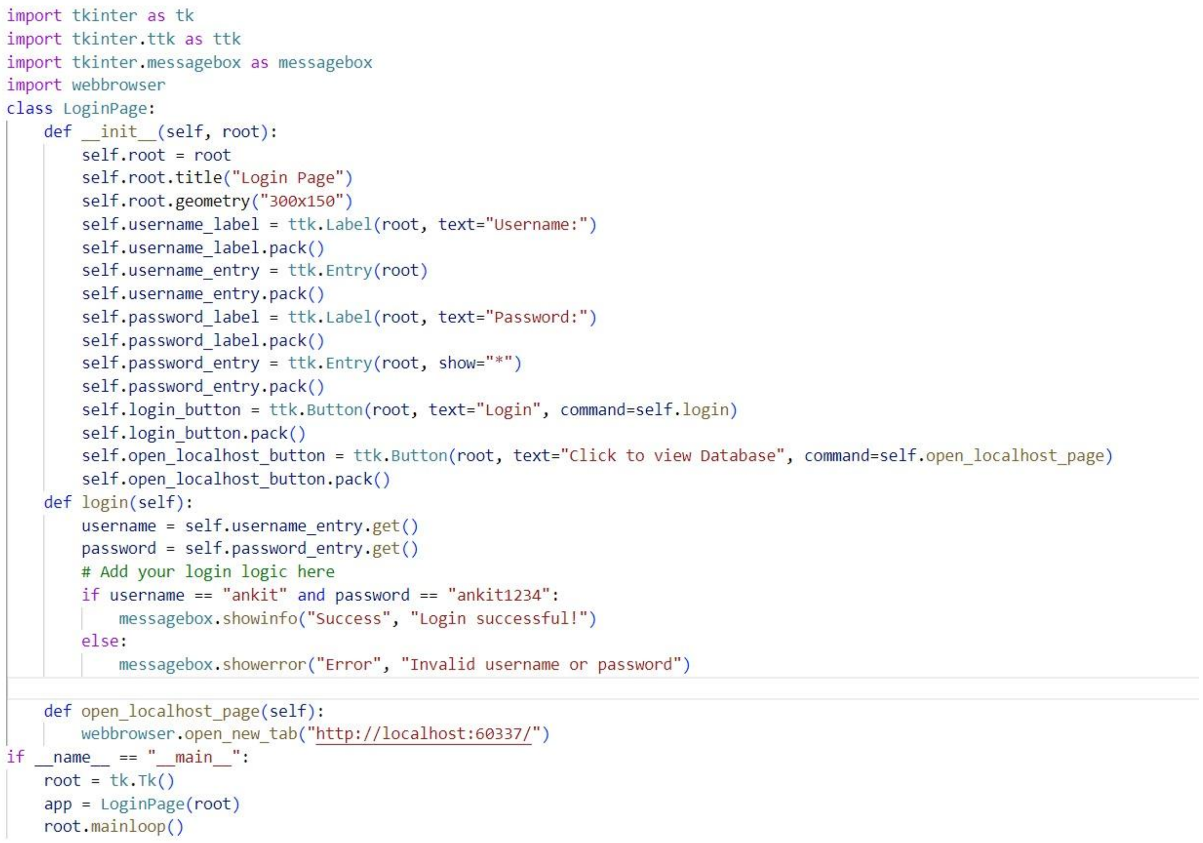
Output:



```
('Gary Kirsten',)
('Kumar Sngakkara',)
('Andy Flower',)
('Sanjay Bangar',)
('Anil Kumble',)
('Trevor Bayliss',)
```

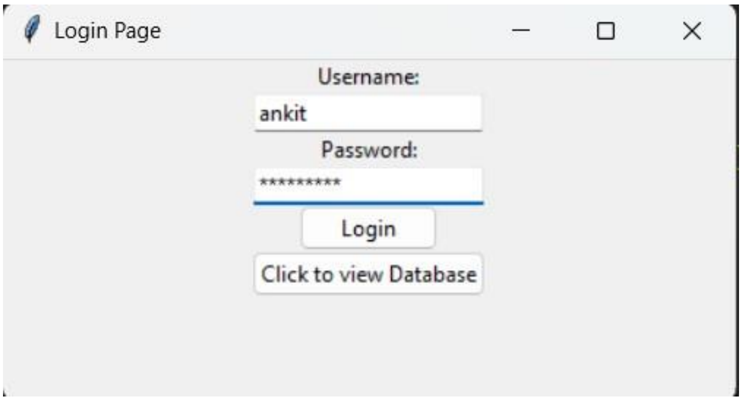
FRONT-END DEVELOPMENT

1. User Login



```
import tkinter as tk
import tkinter.ttk as ttk
import tkinter.messagebox as messagebox
import webbrowser
class LoginPage:
    def __init__(self, root):
        self.root = root
        self.root.title("Login Page")
        self.root.geometry("300x150")
        self.username_label = ttk.Label(root, text="Username:")
        self.username_label.pack()
        self.username_entry = ttk.Entry(root)
        self.username_entry.pack()
        self.password_label = ttk.Label(root, text="Password:")
        self.password_label.pack()
        self.password_entry = ttk.Entry(root, show="*")
        self.password_entry.pack()
        self.login_button = ttk.Button(root, text="Login", command=self.login)
        self.login_button.pack()
        self.open_localhost_button = ttk.Button(root, text="Click to view Database", command=self.open_localhost_page)
        self.open_localhost_button.pack()
    def login(self):
        username = self.username_entry.get()
        password = self.password_entry.get()
        # Add your login logic here
        if username == "ankit" and password == "ankit1234":
            messagebox.showinfo("Success", "Login successful!")
        else:
            messagebox.showerror("Error", "Invalid username or password")

    def open_localhost_page(self):
        webbrowser.open_new_tab("http://localhost:60337/")
if __name__ == "__main__":
    root = tk.Tk()
    app = LoginPage(root)
    root.mainloop()
```

Output:



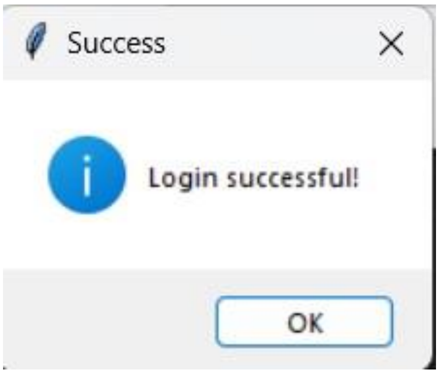
Login Page

Username:

ankit

Password:

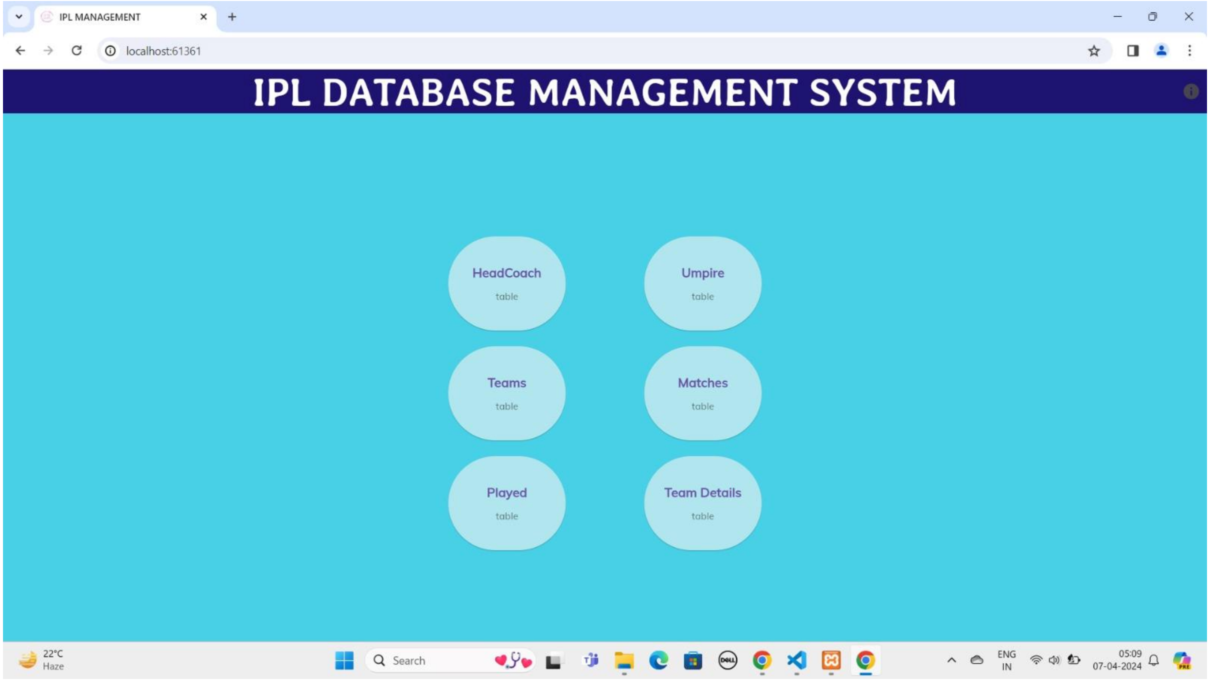
****



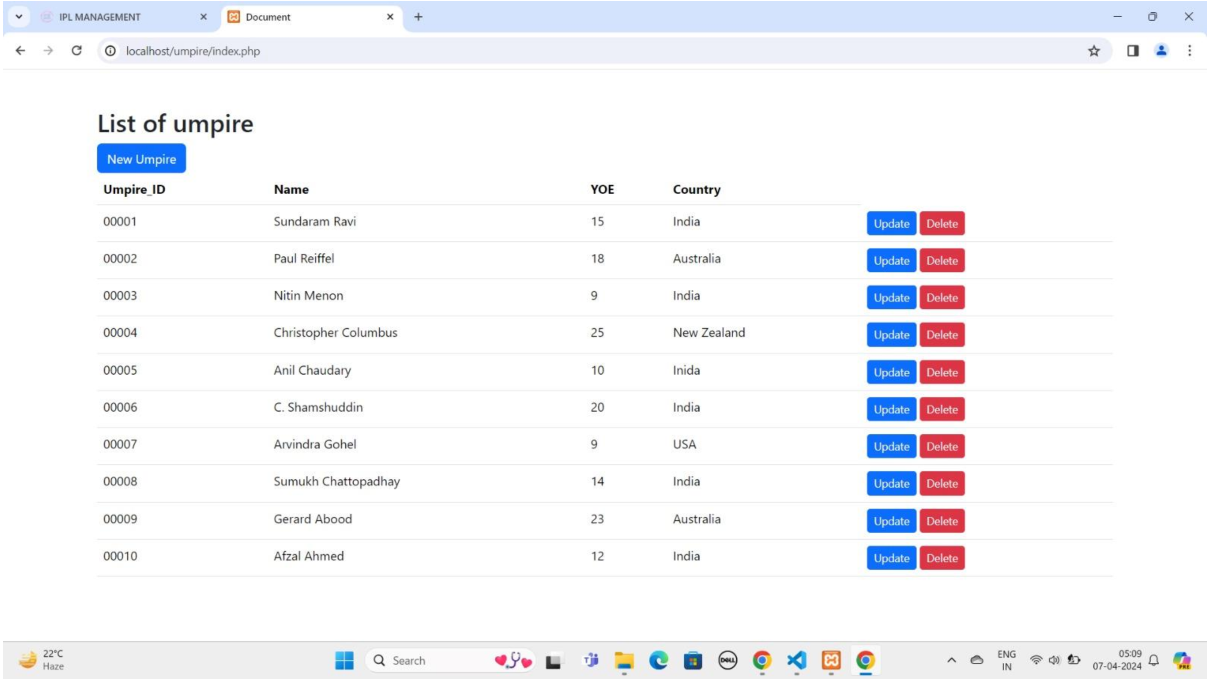
Success

Login successful!

2. Home Page



3. Details of Umpire after clicking on the “Umpire Table” Button

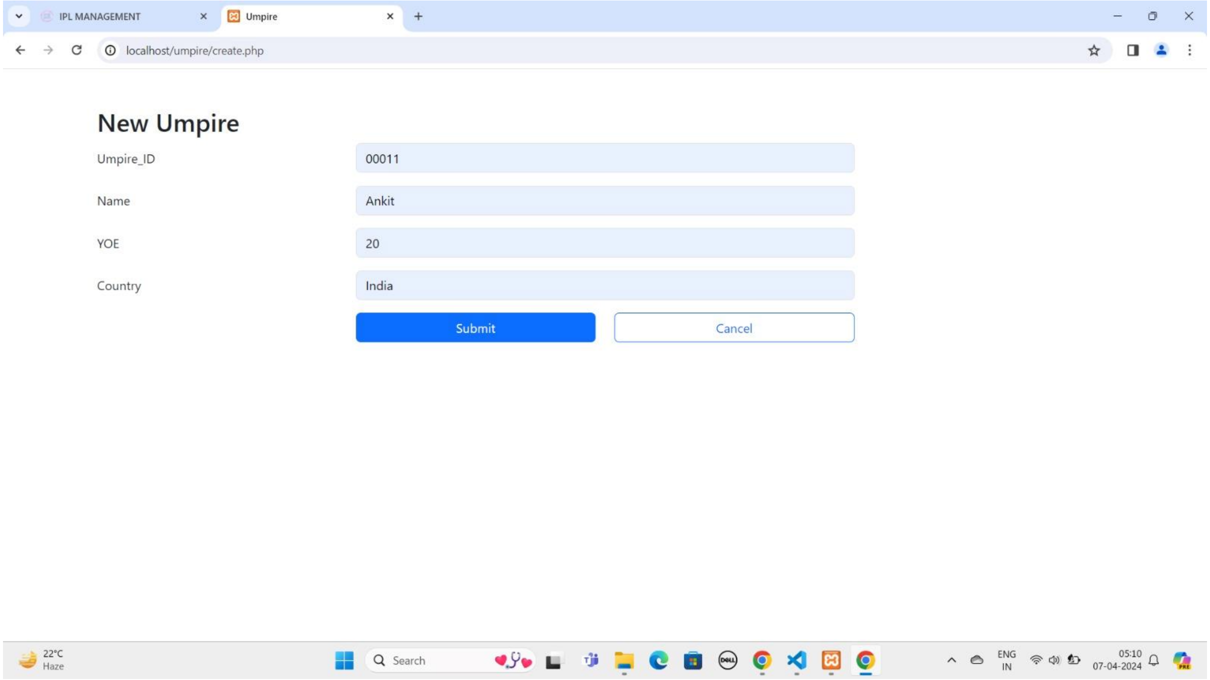


List of umpire

New Umpire

Umpire_ID	Name	YOE	Country	
00001	Sundaram Ravi	15	India	Update Delete
00002	Paul Reiffel	18	Australia	Update Delete
00003	Nitin Menon	9	India	Update Delete
00004	Christopher Columbus	25	New Zealand	Update Delete
00005	Anil Chaudary	10	Inida	Update Delete
00006	C. Shamshuddin	20	India	Update Delete
00007	Arvindra Gohel	9	USA	Update Delete
00008	Sumukh Chattopadhay	14	India	Update Delete
00009	Gerard Abood	23	Australia	Update Delete
00010	Afzal Ahmed	12	India	Update Delete

4. To update the Umpire Table



New Umpire

Umpire_ID	00011
Name	Ankit
YOE	20
Country	India