Relational Schema

Stadium (sName, location, capacity)

SeatInStadium(sName, section, row_num)
sName foreign key referencing Stadium
Section, row_num foreign key referencing Seat

Seat (section, row_num, sName)

Buyer (email, confirmationNum, bName, totalAmount)

Ticket (ticketNum, availability, tType, section, row_num, matchId, price, confirmationNum)
Note: "availability" if the seat is sold already or available for purchase.

section foreign key referencing Seat
row_num foreign key referencing Seat
section, row_num NOT NULL

confirmationNum foreign key referring to Buyer

matchId foreign key referring Match
matchId NOT NULL

Match (matchId, mRound, mLength, mDate, mTime, sName, fCountry, fGoal, sCountry, sGoal)
Note: fCountry, sCountry refers to "first" and "second country".
sName foreign key referencing Stadium
sName NOT NULL

fCountry foreign key referencing Team
sCountry foreign key referencing Team
fCountry NOT NULL
sCountry NOT NULL

Referee (yearsOfExp, country, rName, rId)
Not captured: each referee refs at least one game

RefMatch (role, rId, matchId)
rId referencing Referee
matchId referencing Match

Team (country, NA, tGroup, url)
Note: NA = National Association

TeamMember(tName, DOB, tId, country)
Note: DOB = Date of birth
country foreign key referencing Team
country NOT NULL
Player (tId, shirtNum, position)

tId foreign key referencing TeamMember

Coach (tId, role)
tId foreign key referencing TeamMember

Goal(matchId, occurrence, tId, minute, duringPen)
Note: "duringPen" if the goal is scored during penalty kicks .
matchId references Match
tId foreign key referencing Player NOT NULL
matchId NOT NULL

PlaysMatch (timeIn, timeOut, positionPlayed, YellowCards, receiveRed, matchId, tId)
tId foreign key referencing Player NOT NULL
matchId foreign key referencing Match NOT NULL

Pending constraints

- Can't restrict the team group to be between A - H; could store group "X" in database
- Can't restrict number of matches played in total; can't limit to just 64 matches played
- Can't restrict using enums for example, which player the position played, we can put some random string that isn't limited to forward, midfielder, defender, goalkeeper

SQL Queries:

A)

```
CREATE VIEW TEMP (sName, mDate) as
(SELECT sName, mDate
FROM Match
WHERE matchId in
(
SELECT DISTINCT matchId
FROM Goal
WHERE tId in
(SELECT tId
FROM TeamMember
WHERE tName = 'Christine Sinclair')));

SELECT Stadium.sName, location, mDate
from Stadium, TEMP
where Stadium.sName = TEMP.sName;

drop view TEMP;
```

```
SNAME
                                               LOCATION
                                                                                               MDATE
------------------------------------------------------------------------------------------------------------------------
---------------------------------------------- ----------------------------------------------------------------------
---------------------------------------------------------------------------------------------- ----------
Rose Bowl
                                               USA
                                                                                               07/20/2023
Estadio Monumental
                                               Peru
                                                                                               07/22/2023

  2 record(s) selected.
```

B)

```
Create view notAll as (select tm.tId FROM teammember tm JOIN player p ON tm.tId = p.tId where not exists
(select pm.matchid from playsmatch pm where pm.tId = p.tId));

SELECT t.tName, p.shirtNum, t.country from teammember t join player p on t.tId = p.tId where t.tId not in
(select tId from notall);
```

C)

```
create view tmp as (select t.country, matchid from match m join team t on t.country = m.fCountry) union (select
t.country, m.matchid from match m join team t on t.country = m.sCountry);

Create view tmp2 as (select t.country, t.matchid, g.duringpen from tmp t join goal g on t.matchid = g.matchid);

Create view matchesPlayed as (select country, count(*) matches_played from tmp group by country);
```

Create view nonpengoals as (select country, count(*) goals_scored from tmp2 where duringpen = 0 group by country);

select m.country, m.matches_played, g.goals_scored from matchesplayed m join nonpengoals g on m.country = g.country;

```
db2 => select m.country, m.matches_played, g.goals_scored from matchesplayed m join nonpengoals g on m.country = g.country;

COUNTRY
                                                                            MATCHES_PLAYED GOALS_SCORED
-----------------------------------------------------------------------------------------------------------------------------
---------------------------------------------------------------------------- -------------- ------------
AUS
                                                                                         4            7
CAN
                                                                                         3            6
IRL
                                                                                         2            4
NGA
                                                                                         2            3
NOR
                                                                                         2            3
NZL
                                                                                         3            6
PHI
                                                                                         2            4
SUI
                                                                                         4            7

  8 record(s) selected.
```

D)
select distinct sName
from Seat
where section in
(select section
from Ticket);

Explanation: Select Stadiums in which the seats have been purchased

```
SNAME

------------------------------
------------------------------
Croke Park

Rose Bowl

Stade de France


  3 record(s) selected.
```

E)
SELECT matchId, mDate
FROM Match
WHERE mTime < '19:00';

Explanation: Select match identification and match date in which the match starts before 7 pm

```
MATCHID     MDATE
----------- ----------
          3 07/21/2023
          4 07/21/2023
          6 07/22/2023
          9 07/24/2023

  4 record(s) selected.
```

Player information

A)
CREATE VIEW playerinfo AS
SELECT tm.tName, p.shirtNum, tm.DOB, tm.country, t.NA, t.tGroup
FROM teammember tm JOIN player p

```
        ON tm.tId = p.tId
    JOIN team t
        ON tm.country = t.country;
```

B)

```
db2 => CREATE VIEW playerinfo AS SELECT tm.tName, p.shirtNum, tm.DOB, tm.country, t.NA, t.tGroup FROM teammember tm JOIN
player p ON tm.tId = p.tId JOIN team t ON tm.country = t.country;
DB20000I  The SQL command completed successfully.
db2 =>
db2 =>
```

C)

```
db2 =>
db2 => SELECT * FROM playerinfo LIMIT 5;
```

TNAME	SHIRTNUM	DOB	COUNTRY	NA	TGROUP
Lexi	2	05/12/2002	NZL	NZL	C
Nell	5	05/11/2003	AUS	AUS	B
Tomos	4	01/01/2002	PHI	PHI	C
Cara	1	05/01/2002	NGA	NGA	B
Christine Sinclair	3	05/10/2002	NOR	NOR	C

```
  5 record(s) selected.
```

D)

```
db2 =>
db2 => SELECT * FROM playerinfo WHERE tGroup = 'A' LIMIT 5;
```

TNAME	SHIRTNUM	DOB	COUNTRY	NA	TGROUP
Harry	25	05/10/2000	SZL	SZL	A
Noah	26	01/01/2002	SZL	SZL	A
Ivan	27	05/11/2003	SZL	SZL	A
Hayden	28	08/06/1999	SWD	SWD	A
Mika	29	09/01/1998	SWD	SWD	A

```
  5 record(s) selected.
```

E)

```
db2 => INSERT INTO playerinfo VALUES('George', 18, '03/10/2000','SZL', 'SZL','A');
DB21034E  The command was processed as an SQL statement because it was not a
valid Command Line Processor command.  During SQL processing it returned:
SQL0150N  The target fullselect, view, typed table, materialized query table,
range-clustered table, or staging table in the INSERT, DELETE, UPDATE, MERGE,
or TRUNCATE statement is a target for which the requested operation is not
permitted.  SQLSTATE=42807
db2 =>
```

We get this error because the view table is joining **3** table data; we can only insert into a single-table view.

Check Constraints:

The check constraint was added to the Match table to ensure that there are at most 64 matches. Since the matchId increases for each match, we can simply use ‘CHECK (MatchId<=64)’. Note that matchId is the first value of the Match table.

What displays when creating the Match table:

```
db2 => db2 => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.)
=> db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => db2 (cont.) => CREATE TABLE Match ( ma
tchId INTEGER NOT NULL, mRound INTEGER, mLength INTEGER, mDate DATE, mTime TIME, sName VARCHAR(250) NOT NULL, fCountry VARCHAR(250) NOT NULL, fGoa
l INTEGER, sCountry VARCHAR(250) NOT NULL, sGoal INTEGER, CHECK (matchId<=64), PRIMARY KEY (matchId), FOREIGN KEY (sName) REFERENCES Stadium, FORE
IGN KEY (sCountry) REFERENCES Team, FOREIGN KEY (fCountry) REFERENCES Team )
DB20000I  The SQL command completed successfully.
```

What displays when the value is illegal:

```
db2 => INSERT INTO Match VALUES(65, 4, 45, '07/25/2023', '19:00:00', 'Rose Bowl', 'AUS', 1, 'SUI', 1)
DB21034E  The command was processed as an SQL statement because it was not a
valid Command Line Processor command.  During SQL processing it returned:
SQL0545N  The requested operation is not allowed because a row does not
satisfy the check constraint "CS421G133.MATCH.SQL230227015209730".
SQLSTATE=23513
```