

1. Name of Your Company

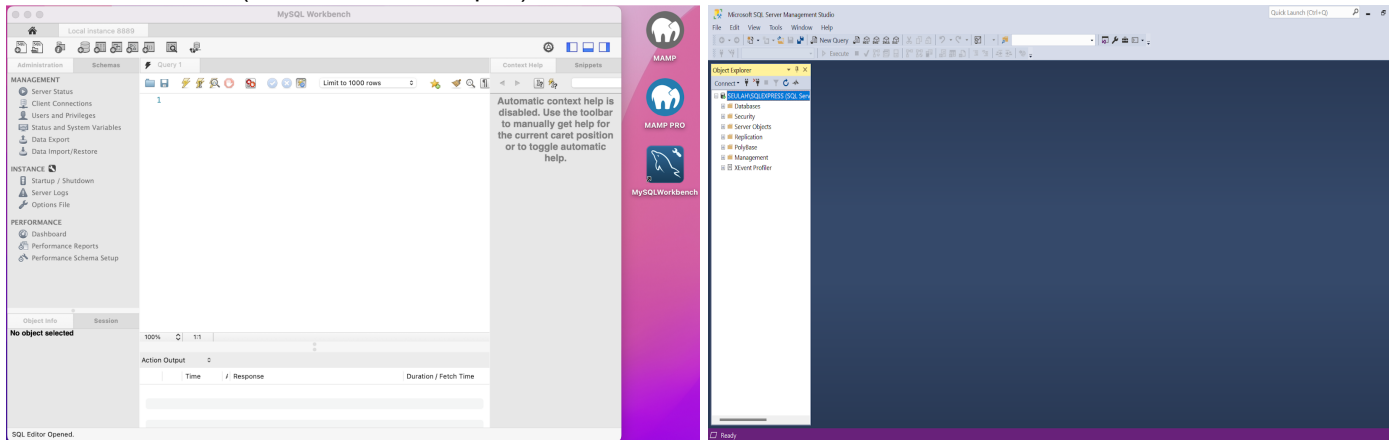
KickBase

2. Project Title

Soccer Matches Management System

3. Team

- Sharon Shen (MySQL Expert) - MAMP
- Seulah Lee (MS SQL Server Expert)



4. Weekly Meeting Hours

We will meet and work on the project every **Monday** from **10:00 AM** to **11:30 AM**.

5. Project Description

A database is created which records the information about many soccer teams, their members, the captain and coach of the team, the matches they participated in, the result of the matches, and the championships they received.

The data requirements for this database are summarized as follows:

- We store the name, date of birth, number of the player (marked on their shirt), picture of each team member, and player ID (surrogate key). Every team member has a role in each match that can be different from match to match. We store the number of goals scored by a team member in each match.
- Every team has a coach. We store the name, date of birth, coach ID (surrogate key) and start and end time of coaching for each.
- For each team, we store the team name, team ID (surrogate key), and home location.
- For each match we store the match ID (surrogate key), location, date, time, the two teams that played against each other, and the result of the match (e.g. 0-5, or 2-4,...).

- If a team won a championship, we store the date and the name of the championship and the medal that they won (gold, silver...or 1st place, 2nd place...)
- Teams may have sponsors that pay the team for a specific match to advertise for them. If there was a sponsor (a team may not have a sponsor for some of its matches), we store the name of the sponsor (the company), the match in which the sponsor supported the team, and the amount of money paid.

6. Assumptions about Cardinality and Participations

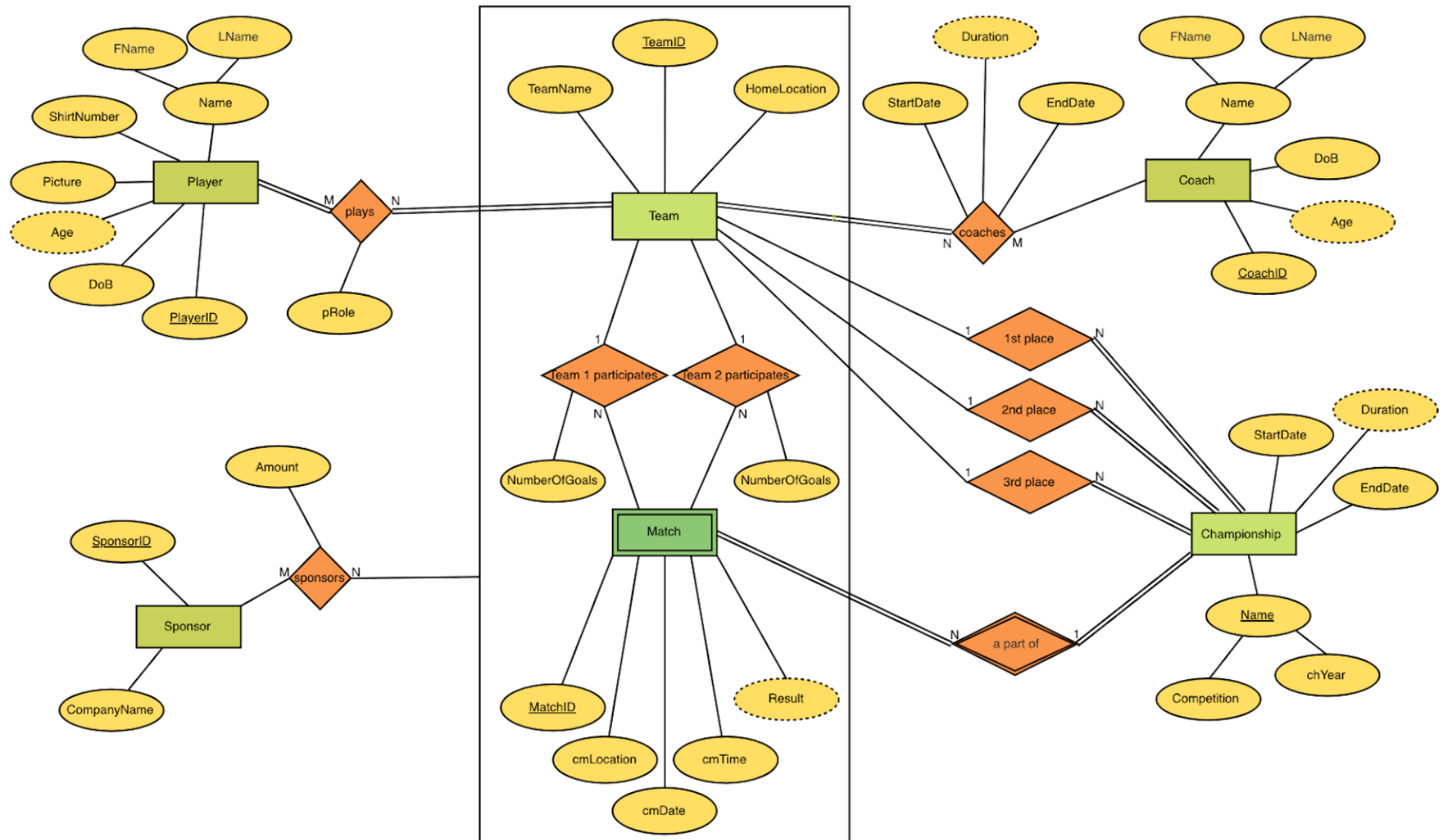
You can write all the assumptions about Cardinality and Participations (total/partial) here.

- Player - Team
 - o Many-to-many (M:N) : Each player can play on multiple teams in their lifetime, and each team can have multiple players.
 - o Total Participation : each team must have players.
 - o Total Participation : each player should be part of the team.
- Coach - Team
 - o Many-to-many (M:N) : Each coach can coach multiple teams in their lifetime, and each team can have multiple coaches.
 - o Total Participation : each team must be coached by a coach at least .
 - o Partial Participation : not all coaches coach a team.
- Team - Match
 - o There will be always two teams play against each other
 - o One-to-many (1:N) : each team participates in multiple matches, while each match can have two teams at a time.
 - o Partial Participation : Not all team play matches
- Tournament - Match
 - o One-to-many (1:N) : each tournament can have multiple matches, but each match cannot have multiple tournaments
 - o Total Participation : each match is identified by/is part of the Championship Tournament.
 - o Total Participation : each championship tournament must have at least one match.
- Tournament - Team
 - o One-to-many (1:N) : each team can win 1st, 2nd, 3rd place in many championships, however, each championship can have only one 1st, 2nd, and 3rd place
 - o Total Participation : Championship must have 1st, 2nd, and 3rd place.
 - o Partial Participation : Not all team places 1st or 2nd or 3rd place in the championship
- [Team - Match] - Sponsor
 - o Many-to-many (M:N) : Teams may have multiple sponsors that pay the team for a specific match to advertise for them. Sponsors may sponsors many teams that play a specific match.
 - o Partial Participation : not all sponsors sponsor teams for a specific match.
 - o Partial Participation : not all teams have to be sponsored by a sponsor.

*Total Participation – Each entity is involved in the relationship.

*Partial participation – Not all entities are involved in the relationship.

7. EER Modeling Diagram



8. ER-Model Mapping to Database Relational Schema

1. Map all regular/strong entities (including Specialization/Generalization)
2. Map weak entities
3. Map binary 1:1 relationship type - None
4. Map binary 1:M relationship type - skip
5. Map binary M:N relationship type
6. Map Recursive relationship type - none
7. Map N-ary relationship type - none
8. Map Multivalued Attributes
9. Map Aggregation

Player(PlayerID, FName, LName, ShirtNumber, Picture, DoB)

Team(TeamID, Name, HomeLocation)

Coach(CoachID, FName, LName, DoB)

Championship(Competition, chYear, StartDate, EndDate, **TeamID_1stPlace**, **TeamID_2ndPlace**, **TeamID_3rdPlace**)

Sponsor(SponsorID, CompanyName)

Championship_Match(**(Competition, chYear)**, MatchID, cmLocation, cmDate, cmTime, Team1_NumberOfGoals, Team2_NumberOfGoals, **Team1_ID**, **Team2_ID**)

Player_Plays_Team(**PlayerID**, **TeamID**, pRole)

Coach_Coaches_Team(**CoachID**, **TeamID**, StartDate, EndDate)

Sponsor_Sponsors_Team_Match(**TeamID**, **(Competition, chYear, MatchID)**, **SponsorID**, Amount)

9. Normalization

All relations must be normalized up to BCNF. You must explain why you believe every relation in your database is normalized.

1NF : remove composite attributes, multivalued attributes, nested tables.

There are composite attributes in Player, Coach, and Championship entities. "Name" composite attributes are broken down to FName and LName for Player and Coach entities. "Name" composite key is decomposed to Competition and chYear for the Championship entity. They are all atomic attributes on the relational schema above. It is in the first normal form.

2NF : remove partial functional dependencies. A non-prime attribute depends on only part of the key.

All relations are already in 2NF. Player, Team, Coach, Championship, and Sponsor entities have a single key. For the rest of relationship relations entities, all the non-prime attributes are fully functionally dependent on the primary key.

3NF : remove transitive functional dependencies only if Y is not a candidate key ($X \rightarrow Y \rightarrow Z$).

There are no transitive functional dependencies, so this relation is in third normal form.

BCNF : non-prime attribute cannot determine any part of the key.

10. Determining Data Types (Domain) and Constraints

You explain why you choose a certain data type for a field and why you apply certain constraints

Field	Data Type	Constraint
Player Table		
PlayerID	INT We want an integer number to identify each player.	PRIMARY KEY Each player should be uniquely identified. MySQL : AUTO_INCREMENT = 1000 MS SQL : IDENTITY(1000,1) We want a unique number to be generated automatically when a new record is inserted into a table (surrogate key) and be consistent. It starts with 1000 and increments by 1.
FName	VARCHAR(50) There are variant lengths of first names depending on countries.	NOT NULL Each player should have a valid first name.
LName	VARCHAR(50) There are variant lengths of last names depending on countries.	NOT NULL Each player should have a valid first name.
ShirtNumber	NUMERIC(2) Each player has integer number limited to two digits	
Picture	MySQL : BLOB MS SQL : VARBINARY We are storing an image in this column. Recently, Microsoft began suggesting using VARBINARY(MAX) instead of IMAGE for storing a large amount of data in a single column since IMAGE will be retired in a future version of MS SQL Server.	
DoB	DATE We want the date format to be stored for the Date of Birth.	
Team Table		
TeamID	INT We want an integer number to identify each team	PRIMARY KEY Each team should be uniquely identified. MySQL : AUTO_INCREMENT = 1000 MS SQL : IDENTITY(1000,1) We want a unique number to be generated automatically

		when a new record is inserted into a table (surrogate key) and be consistent. It starts with 1000 and increments by 1.
TeamName	VARCHAR(20) We expect the TeamName to be less than 20 characters.	
HomeLocation	VARCHAR(20) We expect the HomeLocation to be less than 20 characters.	
Coach Table		
Coach	INT We want an integer number to identify each coach.	PRIMARY KEY Each coach should be uniquely identified. MySQL : AUTO_INCREMENT = 1000 MS SQL : IDENTITY(1000,1) We want a unique number to be generated automatically when a new record is inserted into a table (surrogate key) and be consistent. It starts with 1000 and increments by 1.
FName	VARCHAR(50) There are variant lengths of first names depending on countries.	NOT NULL Each coach should have a valid first name.
LName	VARCHAR(50) There are variant lengths of last names depending on countries.	NOT NULL Each coach should have a valid first name.
DoB	DATE We want the date format to be stored for the Date of Birth.	
Championship		
Competition	VARCHAR(20) We expect the Competition Name to be less than 20 characters.	PRIMARY KEY Championship is identified by a combination of the name of competition and the year it is held (Composite Key)
chYear	INT The year held Championship Tournament is in integer format.	
StartDate	DATE We want the date format to be stored for the StartDate	
EndDate	DATE We want the date format to be stored for the EndDate.	

TeamID_1stPlace	INT We want an integer number to identify 1stPlace team	FOREIGN KEY We want to know which team placed first (1stPlace) for each championship.
TeamID_2ndPlace	INT We want an integer number to identify 2ndPlace team	FOREIGN KEY We want to know which team placed second (2ndPlace) for each championship.
TeamID_3rdPlace	INT We want an integer number to identify 3rdPlace team	FOREIGN KEY We want to know which team placed third (3rdPlace) for each championship.
Sponsor Table		
SponsorID	INT We want an integer number to identify each sponsor	PRIMARY KEY Each sponsor should be uniquely identified. MySQL : AUTO_INCREMENT = 1000 MS SQL : IDENTITY(1000,1) We want the primary key to be auto-generated (surrogate key) and consistent. It is starting with 1000 and increment by 1.
CompanyName	VARCHAR(20) We expect the Company Name to be less than 20 characters.	
Championship_Match Table		
Competition	VARCHAR(20) We expect the Competition Name to be less than 20 characters.	PRIMARY KEY Championship is identified by a combination of the name of competition and the year it is held (Composite Key). As Match is part of (Identifying relationship) Championship, it becomes a primary key to identify this relationship. FOREIGN KEY ON DELETE CASCADE If a value is deleted from Championship, the value from this relationship must be deleted too.
chYear	INT The year held Championship Match is in integer format.	
MatchID	INT	PRIMARY KEY The partial key of the weak entity becomes the primary key in the relationship.

cmLocation	VARCHAR(20)	
cmDate	DATE We want the date format to be stored for the Championship Match Date.	
cmTime	TIME We want the time format to be stored for the Championship Match Time.	
Team1_NumberOfGoals	INT We want an integer number to identify the number of goals scored for team1.	
Team2_NumberOfGoals	INT We want an integer number to identify the number of goals scored for team2.	
Team1ID	INT Team1 is identified with a unique integer number. *There are always two teams play against each other	FOREIGN KEY We want to know which team has participated as team1.
Team2ID	INT Team2 is identified with a unique integer number. *There are always two teams play against each other	FOREIGN KEY We want to know which Team has participated as team2.
Player_Plays_Team Table		
PlayerID	INT Each player is identified with a unique integer number.	PRIMARY KEY FOREIGN KEY We want to know which Team (TeamID) the Player (PlayerID) plays on.
TeamID	INT Each team is identified with a unique integer number.	PRIMARY KEY FOREIGN KEY We want to know which Team (TeamID) the Player (PlayerID) plays on.
pRole	VARCHAR(20) Each team member plays a specific role on the team. The role title is less than 20 characters. e.g., Forward, Defender, Goalkeeper, Midfielder	
Coach_Coaches_Team Table		

CoachID	INT Each coach is identified with a unique integer number.	PRIMARY KEY FOREIGN KEY We want to know which Team (TeamID) the Coach (CoachID) coaches.
TeamID	INT Each team is identified with a unique integer number.	PRIMARY KEY We want to know which Team (TeamID) the Coach (CoachID) coaches. FOREIGN KEY ON DELETE CASCADE If a value is deleted from Team, the values of this relationship must be deleted too.
StartDate	DATE We want the date format to be stored for the StartDate.	
EndDate	DATE We want the date format to be stored for the EndDate.	
Sponsor_Sponsors_Team_Match Table		
TeamID	INT Each team is identified with a unique integer number.	PRIMARY KEY FOREIGN KEY As part of a virtual entity, TeamID from Team entity is referenced to identify this relationship.
Competition	VARCHAR(20) Competition Name to be less than 20 characters.	PRIMARY KEY FOREIGN KEY As part of a virtual entity, Competition, chYear, MatchID, from Match weak entity is referenced to identify this relationship.
chYear	INT The year held Championship Match is in integer format.	
MatchID	INT Each Match is identified with a unique integer number.	
SponsorID	INT Each sponsor is identified with a unique integer number.	PRIMARY KEY FOREIGN KEY SponsorID from Sponsor entity is referenced to identify this relationship.
Amount	NUMERIC(10, 2) The amount of money paid is in dollars, therefore need two decimal points, and a large enough number of the digits.	

11. Creating Database and Tables - SQL DDL

You do not need to copy SQL commands here. Save your SQL commands in a script file and just mention the name of the file here. Make sure the script file is stored besides this document within the same folder.

- **MSSQLScripts** - CreateTables.sql & CreateTables.txt , DropTables.sql & DropTables.txt
- **MySQLScripts** - CreateTables.sql & CreateTables.txt , DropTables.sql & DropTables.txt

12. Inserting Values in Tables

You do not need to copy SQL commands here. Save your SQL commands in a script file and just mention the name of the file here. Make sure the script file is stored beside this document within the same folder.

- **MSSQLScripts** - InsertIntoValues.sql & InsertIntoValues.txt
- **MySQLScripts** - InsertIntoValues.sql & InsertIntoValues.txt

FIFA Women's World Cup refs:

<https://www.fifa.com/tournaments/womens/womensworldcup>

13. SQL Queries

You do not need to copy SQL commands here. Save your SQL commands in a script file and just mention the name of the file here. Make sure the script file is stored beside this document within the same folder.

- **MSSQLScript** - RetrivalData.sql & RetrivalData.txt
- **MySQLScript** - RetrivalData.sql & RetrivalData.txt

14. Views

You do not need to copy SQL commands here. Save your SQL commands in a script file and just mention the name of the file here. Make sure the script file is stored beside this document within the same folder.

- **MSSQLScript** - RetrievalData.sql & RetrievalData.txt
- **MySQLScript** - RetrievalData.sql & RetrievalData.txt