INFO20003 Semester 2, 2022

Assignment 2: SQL
Due: 6:00pm Friday 16th September
Weighting: 10% of your total assessment

Melbourne Touch Rugby Database

Description

The Melbourne Touch Rugby competition is a 6-player-a-side competition for men's, women's and mixed teams (a mixed team has a minimum of 3 women and maximum of 3 men on the field at any time). Players must be registered with a club to play in the competition and players can only be registered for one club at a time.

Currently there are 8 rugby clubs participating in the competition, which began in 2020. Each club has three teams – a men's team, a women's team and a mixed team. The men's competition is known as the Dewar Shield; the women's competition is known as the Williams Plate; and the mixed teams competition is known as the Bingham Trophy.

A *season* is the set of games played in a competition in a calendar year. Each season consists of *rounds* in which every team plays a game.

Even though each game is played by 12 players (6 from each team), a team is not a fixed group of 6 players. The team officials choose 6 players from their club, or possibly even from another club, to play for that team in a game. As such, the composition of a team may vary from game to game.

If for any reason a team is unable to organise enough players to play a game, that team will forfeit the game and their opponents will score a "walkover". A walkover awards 28 points to the team who scores a walkover and no score is recorded for the team who forfeited. If a game is cancelled (e.g. due to extreme heat, unsuitable playing pitch), no score is recorded against either team. For the purposes of considering which games a player has *'played'* in, forfeit and cancelled games are still counted unless otherwise specified. A walkover counts as a '*win*' and '*loss*' for the corresponding teams, whereas cancelled games are considered a *'draw'* (neither team won nor lost) for the purposes of analysis.

The Data Model
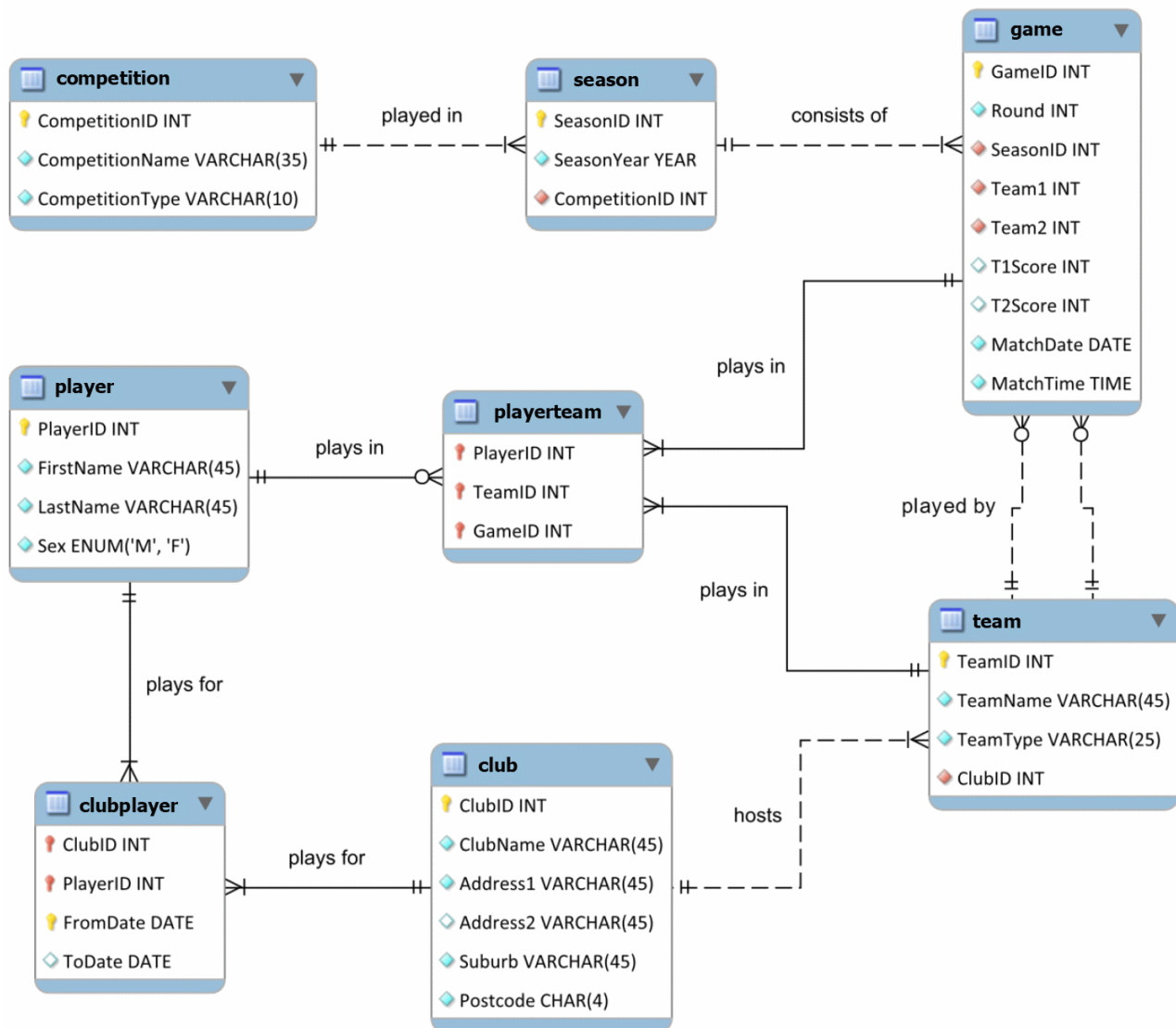
The physical ER model of Melbourne Touch Rugby database.

Implementation Notes

Each season consists of multiple rounds, the 'round' that each game is played in is stored as an attribute of Game.

A forfeit game (walkover) results in the winning team having a score of 28, and the forfeit team having a score of 'NULL'. Cancelled games result in both teams having 'NULL' scores.

Assignment 2 Setup

A dataset is provided which you may use when developing your solutions. To set up the dataset, download the file **rugby_2022.sql** from the Assignment link on Canvas and run it in Workbench. This script creates the database tables and populates them with data. Note that this dataset is provided for you to experiment with, it is not the same dataset as what your queries will be tested against (the schema will stay the same, but the data itself may be different). *This means when designing your queries you must consider edge cases even if they are not represented in this particular data set.* We encourage you to modify the dataset and add your own data to test edge cases.

The script is designed to run against your account on the Engineering IT server (info20003db.eng.unimelb.edu.au). If you want to install the schema on your own MySQL Server installation, *uncomment the lines at the beginning of the script.*

***Note: Do NOT disable only_full_group_by mode when completing this assignment*.** This mode is the default, and is turned on in all default installs of MySQL workbench. You can check whether it is turned on by running the following in workbench:

```
SELECT @@sql_mode;
```

The command should return a string containing "ONLY_FULL_GROUP_BY" or "ANSI". When marking, our test server WILL have this mode turned on, and if your query fails due to this, you will lose marks. Run the below command to add the only_full_group_by mode:

```
SET sql_mode=(SELECT CONCAT(@@sql_mode,',ONLY_FULL_GROUP_BY'))
```

The SQL tasks

In this section are listed 10 questions for you to answer. Write one (single) SQL statement per question. Subqueries and nesting are allowed. You may be penalized for *overly* complicated SQL statements (e.g. multiple times longer than required, used vague/poorly named variables, formatting makes it difficult to read, etc). Unless specified in the question, you do NOT need to sort the results of your query. **DO NOT USE VIEWS (or 'WITH' statements/common table expressions) to answer questions.**

1. Find the names of all players that are currently in the 'Melbourne Tigers' club. Your query should return data in the form (firstName, lastName) **(1 mark)**

2. Find the team which has given the most walkovers (forfeited the most number of games). Assume there are no ties. Your query should return one row of the form (teamName). **(1 mark)**

3. Find the player that has played in the most clubs over time. You may assume there are no ties. Don't count the same club multiple times if a player has played for them multiple times, e.g. if a player joined and quit the 'Melbourne Tigers' several times over the years, all these memberships only count as having played in one club. Return as (firstName, lastName). **(1 mark)**

4. Find players that have played more games in 2020 than in 2021. Return as (firstName, lastName, numGames2020, numGames2021) **(2 marks)**

5. Find the team(s) that ended with the most points scored in the Williams Plate in the 2021 season. If multiple teams tied for total points, you must display all teams that tied for first. Return as (teamName, sumOfPoints) **(2 marks)**

6. Find the names of all women players who have played in the Bingham Trophy, but never the Williams Plate. Return as (firstName, lastName). **(2 marks)**

7. For all players that have ever played a game in the league, list the first *and* most recent club they ever joined. If these two clubs happen to be the same, return the same name for both. Return as (firstName, lastName, firstClubName, mostRecentClubName) **(2 marks)**

8. Find players that have played for every club which has the word 'Melbourne' in its name, and have never joined any of those clubs more than once (i.e. they've never left and rejoined any club with the word 'Melbourne' in its name). Return as (firstName, lastName) **(3 marks)**

9. A player is said to have 'the magic touch' when they win (or draw) every game they play in during a competition season. Find all players who had 'the magic touch' in some season, *but one or more of the teams they played for during that season still lost a game during that season (when they weren't playing!)*. Note a 'competition season' is specific to a competition, so e.g. the '2021 Dewar Shield' season is distinct from the '2021 Bingham Trophy Season'. Return as (firstName, lastName, competitionName, competitionYear). **(3 marks)**

10. Find the club that has the largest difference between the win-ratio (ratio of wins/losses; where a draw is neither a win nor loss) of their best and worst teams in 2021 (across all competitions). Assume there are no ties. Teams that did not play a game in 2021 should not be considered. Return as (clubName) **(3 marks)**

SQL Response Formatting Requirements

To help us mark your assignment queries as quickly/accurately as possible, please ensure that:

- Your query returns the projected attributes in the same order as given in the question, and does **not** include additional columns. E.g., if the question asks 'return as (userId, name)', please write **"SELECT userId, name …"** instead of **"SELECT name, userId…"** (*you can name the columns using `As` however you'd like, only the **order** matters*).

- Please do NOT use "databaseName.tableName" format. E.g., please write **"SELECT userId FROM users…"** instead of **"SELECT userId FROM coltonc.users …"**.

- Ensure that you are using single quotes(') for strings (e.g. **…WHERE name = 'bob'…**) and double quotes (") only for table names (e.g. **SELECT name FROM "some table name with spaces"…**). Do **NOT** use double quotes for strings **"…WHERE name = "bob"…".**

Submission Instructions

Your submission will be in the form of an SQL script. There is a template file on the LMS, into which you will paste your solutions and fill in your student details (more information below).

This .sql file should be submitted on Canvas by **6pm** on the due date of **Friday, September 16th**. Name your submission as 987654.sql, where 987654 corresponds to YOUR student id.

Filling in the template file:

The template file on the LMS has spaces for you to fill in your student details and your answers to the questions. There is also an example prefilled script available on the LMS as well. Below are screenshots from those two documents explaining the steps you need to take to submit your solutions:

Step	Example

1. At the top of the template, you'll need to replace "XXXXXXXX" with your student number and name	*Template*
	
	`-- Your Name: XXXXXXX`
	`-- Your Student Number: XXXXXXX`
	
	Example Filled in
	
	`-- Your Name: Colton Carner`
	`-- Your Student Number: 693281`

2. For each question 1-10, place your SQL solution in between the "BEGIN QX" and "END QX" markers. **Ensure each query is terminated with a semicolon ";"**	*Template*
	
	`--`
	`-- BEGIN Q1`
	
	`-- END Q1`
	`--`
	
	Example Filled in
	
	`--`
	`-- BEGIN Q1`
	
	`-- **This is an example of a comment which will not be executed`
	`-- **(comments are not NEEDED, but if your query is complex you can leave some)`
	
	`-- **Below is an example of how you would enter your answer:`
	`SELECT *`
	`FROM delivery`
	`NATURAL JOIN deliveryitem`
	`WHERE supplierid = 101;`
	
	`-- **It's OK to add more space in between the 'BEGIN QX' and 'END QX' markers`
	`-- **for each question, just don't DELETE the markers!`
	
	`-- **Make sure you fill out your name + student num at the top of the document`
	
	`-- **After reading / understanding these comments in the Q1 section`
	`-- **(comments starting with '**'), feel free to delete them`
	`-- **(don't delete lines without **)`
	
	`-- END Q1`
	`--`

3. Test that your script is valid SQL by running it from MySQL Workbench. Run the entire script by copy-pasting this entire file into a new workbench tab, placing your cursor at the start of the file (without selecting anything), and pressing the lightning bolt to run the entire file.	*All queries ran sequentially and were successful.*
All queries should run successfully one after another. If not, check to make sure you added semicolons ';' after each query.	

All queries ran sequentially and were successful.

	#	Time	Action
✓	9	13:15:53	select id, t...
✓	10	13:15:53	select foru...
✓	11	13:15:53	select foll...
✓	12	13:15:53	select ad...
✓	13	13:15:53	select out...
✓	14	13:15:53	select pos...
✓	15	13:15:53	select par...
✓	16	13:15:53	select id fr...
✓	17	13:15:53	select out...
✓	18	13:15:53	SELECT ...

Late submission

Unless you have an approved extension (see below), you will be penalised -10% of the total number of marks in the assignment per day that your submission is late. For instance, if you received a 78% raw score, but submitted 2 days late, you'd receive a 58% score for the assignment.

Requesting a Submission Deadline Extension

If you need an extension due to a valid (medical) reason, you will need to provide evidence to support your request by **6pm on the 14th of September**. Extensions received after this time may be rejected. Medical certificates need to be at least two days in length.

To request an extension:

- Email the Head Tutor Colton (coltonc@unimelb.edu.au) from your university email address, supplying your student ID, the extension request and supporting evidence. Please add INFO20003 in the subject title.
- If your submission deadline extension is granted you will receive an email reply granting the new submission date. Do not lose this email!

Reminder: INFO20003 Hurdle Requirements

To pass INFO20003, you must pass two hurdles:

- **Hurdle 1:** Obtain at least 50% (15/30) for the three assignments (each worth 10%)
- **Hurdle 2:** Obtain at least 50% (35/70) for the combination of the quizzes and final exam

Therefore, it is our recommendation that you attempt every assignment and question in the exam.

GOOD LUCK!