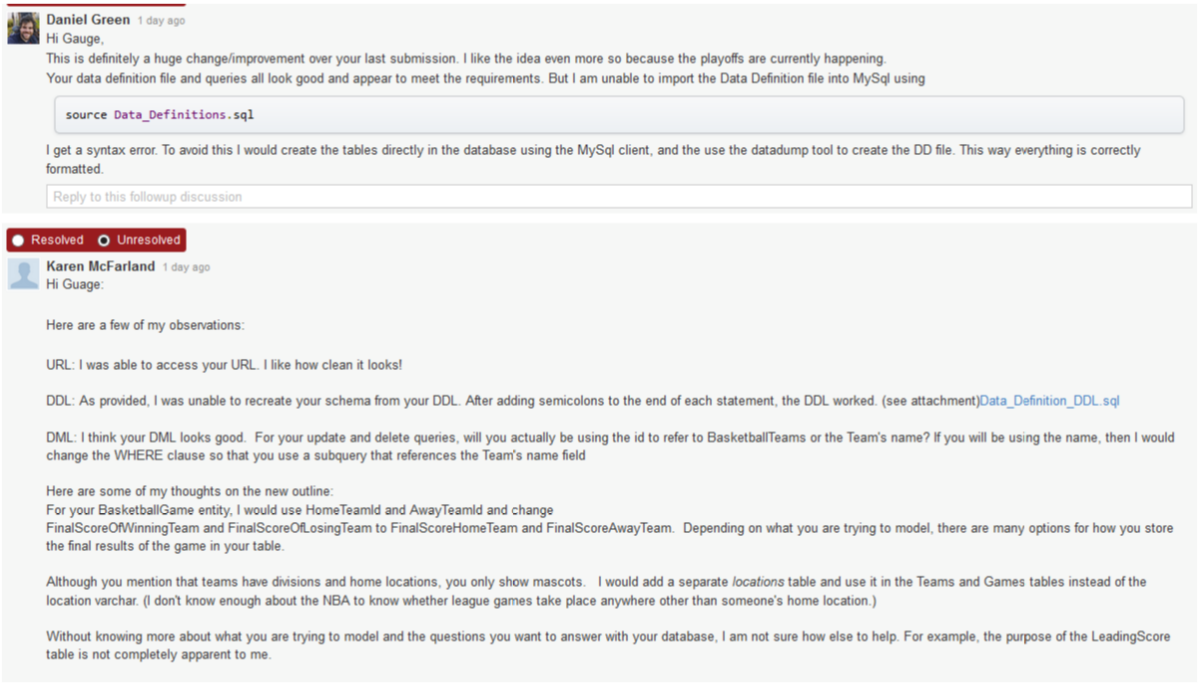
Gauge Hartwell
Allen Amusin
CS340 -
section 401

Project Step 4 Draft Version: CREATE & READ Operations

https://web.engr.oregonstate.edu/~hartwelg/

Feedback by the reviewers:

Include verbatim the peer review that your Draft submission received. If you did not receive any peer review, mention that.



Daniel Green 1 day ago
Hi Gauge,
This is definitely a huge change/improvement over your last submission. I like the idea even more so because the playoffs are currently happening.
Your data definition file and queries all look good and appear to meet the requirements. But I am unable to import the Data Definition file into MySql using

```
source Data_Definitions.sql
```

I get a syntax error. To avoid this I would create the tables directly in the database using the MySql client, and the use the datadump tool to create the DD file. This way everything is correctly formatted.

Reply to this followup discussion

Resolved Unresolved

Karen McFarland 1 day ago
Hi Guage:

Here are a few of my observations:

URL: I was able to access your URL. I like how clean it looks!

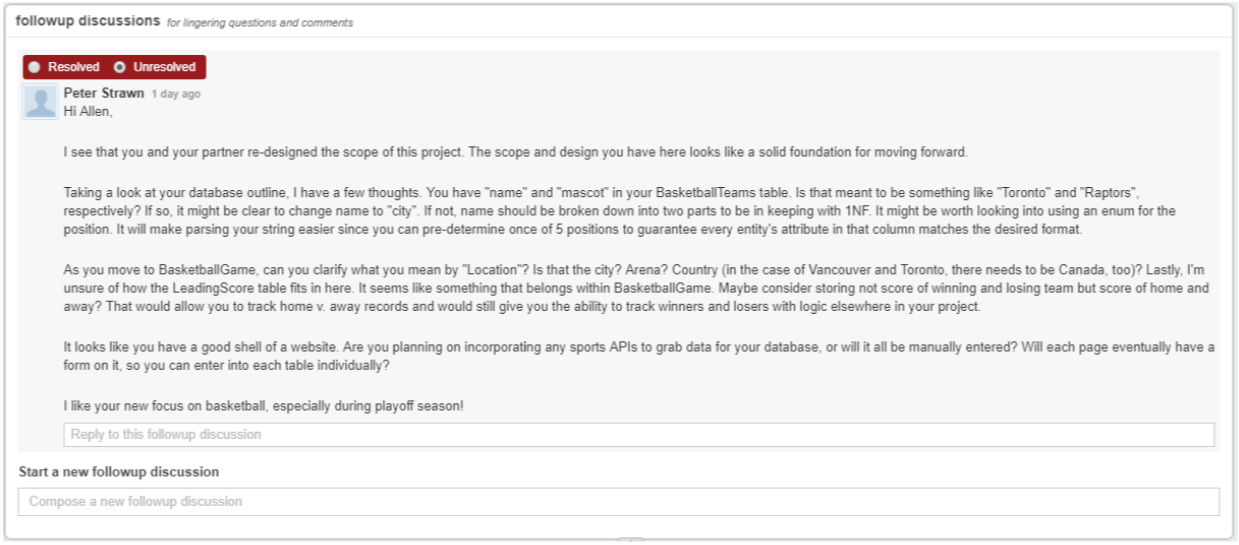
DDL: As provided, I was unable to recreate your schema from your DDL. After adding semicolons to the end of each statement, the DDL worked. (see attachment)Data_Definition_DDL.sql

DML: I think your DML looks good. For your update and delete queries, will you actually be using the id to refer to BasketballTeams or the Team's name? If you will be using the name, then I would change the WHERE clause so that you use a subquery that references the Team's name field

Here are some of my thoughts on the new outline:
For your BasketballGame entity, I would use HomeTeamId and AwayTeamId and change
FinalScoreOfWinningTeam and FinalScoreOfLosingTeam to FinalScoreHomeTeam and FinalScoreAwayTeam. Depending on what you are trying to model, there are many options for how you store the final results of the game in your table.

Although you mention that teams have divisions and home locations, you only show mascots. I would add a separate *locations* table and use it in the Teams and Games tables instead of the location varchar. (I don't know enough about the NBA to know whether league games take place anywhere other than someone's home location.)

Without knowing more about what you are trying to model and the questions you want to answer with your database, I am not sure how else to help. For example, the purpose of the LeadingScore table is not completely apparent to me.



Actions based on the feedback:
List briefly the actions that you chose to take based on the above feedback. If you decided not to act on a specific suggestion, you need to describe your reasoning in detail.

We updated our Data_Definition based on the feedback.
We could not find an API so we manually added data.

Upgrades to the Draft version: If you are making any changes to the files/static website based on your own changed design decisions, they should be listed under this section.

We updated our Data_Definition based on the feedback.

a) Project Outline and Database Outline – Updated Version:

Project Outline

We will be creating a database representing the 2018 National Basketball Association (NBA). The NBA is a professional sports league consisting of 30 teams, each with a mascots and players. Teams are further broken down into 6 divisions each, with 5 teams in each division. Statistics are kept for each player to compare and track their abilities. To model this, we will use 4 entities: BasketballTeams, BasketballTeamPlayer, BasketballGame, and LeadingScore.

Database Outline

Entity: BasketballTeams		
Attribute	Data Type	Description
id	Integer, not null	Auto-incrementing integer automatically assigned when a row is created in this entity.
name	Varchar, not null	name of the team. Varchar value. No defaults, no empty strings
mascot	Varchar, not null	string containing the mascot name. Varchar value of up to 30 characters. No empty strings, no defaults.
NumWins	Integer, not null	number of wins in the current season. Int value. No empty values, no defaults.
NumLosses	Integer, not null	number of losses in the current season. Int value. No empty values, no defaults.
BasketballTeamId	Integer, not null	Id of the basketball team. Automatically assigned int value, automatically incrementing.

Entity: BasketballTeamPlayer		
Attribute	Data Type	Description
Id	Integer, not null	Id of the player on the basketball team. Int value, automatically incrementing, automatically assigned, no empty values.
BasketballTeamId	Integer, not	id of the team the basketball player belongs

	null	to. Int value, connects to BasketballTeamId in other tables.
FirstName	Varchar, not null	first name of the player. String value of up to 60 characters. No empty values, no default names.
LastName	Varchar, not null	last name of the player. String value of up to 60 characters. No empty values, no default names.
JerseyNumber	Integer, not null	number on the player's jersey. Int value, no empty strings, no default values.
Position	Varchar, not null	position played by the player. String value of up to 60 characters. No empty strings. Automatically assigned, defaults to either PG, SG, SF, PF, C.

Entity: BasketballGame		
Attribute	Data Type	Description
id	Integer, not null	id of the game. Int value. Automatically assigned, no empty strings, no defaults.
FinalScoreOfWinningTeam	Integer, not null	final score for the game for the winning team. Int value. No empty values, defaults to 0.
FinalScoreOfLosingTeam	Integer, not null	final score for the game for the losing team. Int value. No empty values,

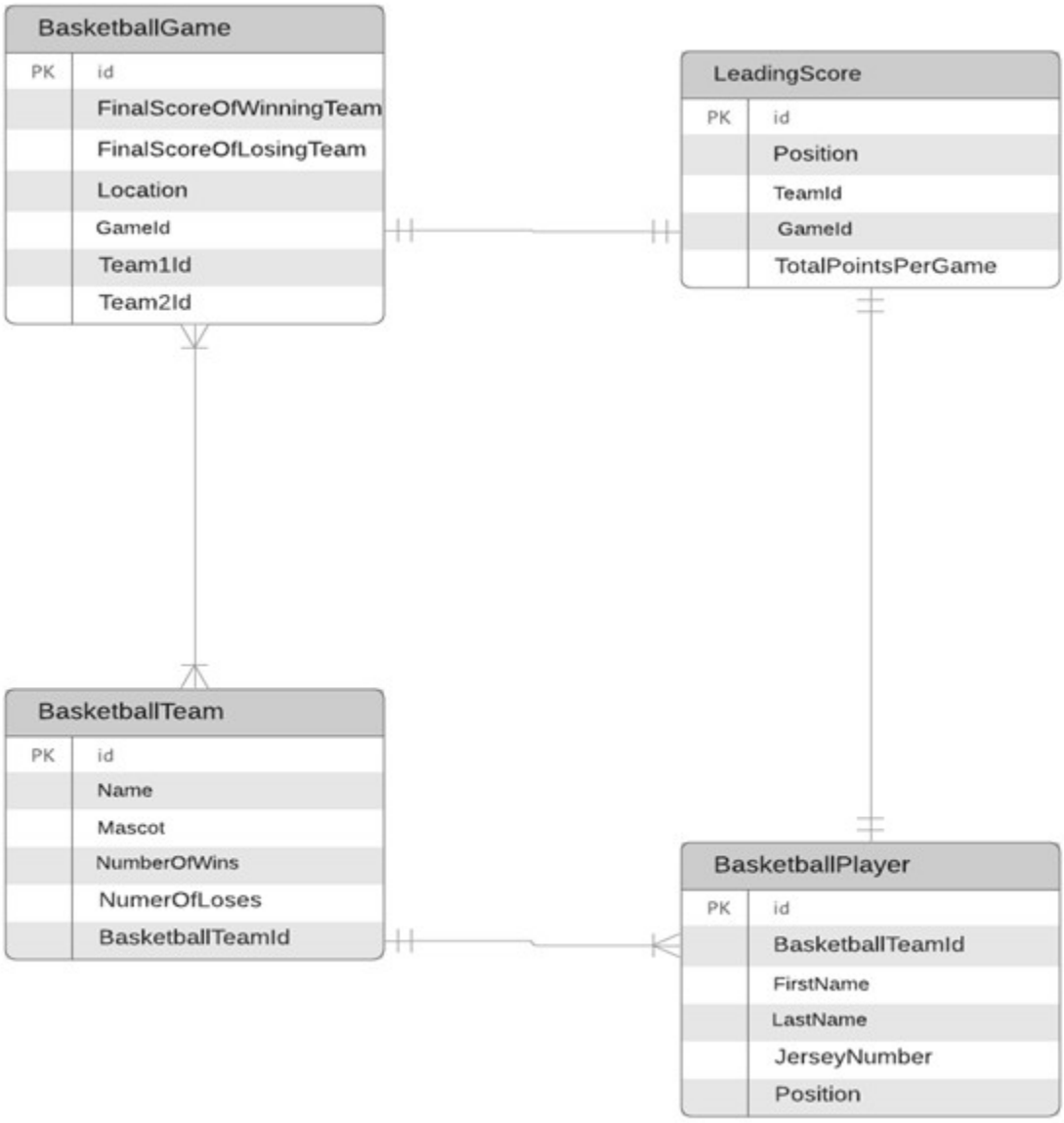
		defaults to 0.
Location	Varchar, not null	location for the basketball game. String value of up to 60 characters. No empty strings, no defaults.
GameId	Integer, not null	id of the basketball game. Int value, automatically assigned, automatically incrementing, no empty values.
Team1Id	Integer, not null	id of team 1. Int value. Links to BasketballTeamId in other tables.
Team2Id	Integer, not null	id of team 2. Int value. Links to BasketballTeamId in other tables.

Entity: LeadingScore		
Attribute	Data Type	Description
id	Integer, not null	Id of the BasketballTeamPlayer with the leading score. Links to Id in the BasketballTeamPlayer table.
Position	Varchar, not null	Int of value 1 or 2, to indicate winning or losing team. No empty values.
TeamId	Integer, not null	Id of team with leading score. Int value, links to BasketballTeamId in other tables.
GameId	Integer, not null	Id of current game being played. Int value, links to GameId in other tables.
TotalPointsPerGame	Integer, not null	Total points of the BasketballTeam with the leading score. Int value, defaults to 0.

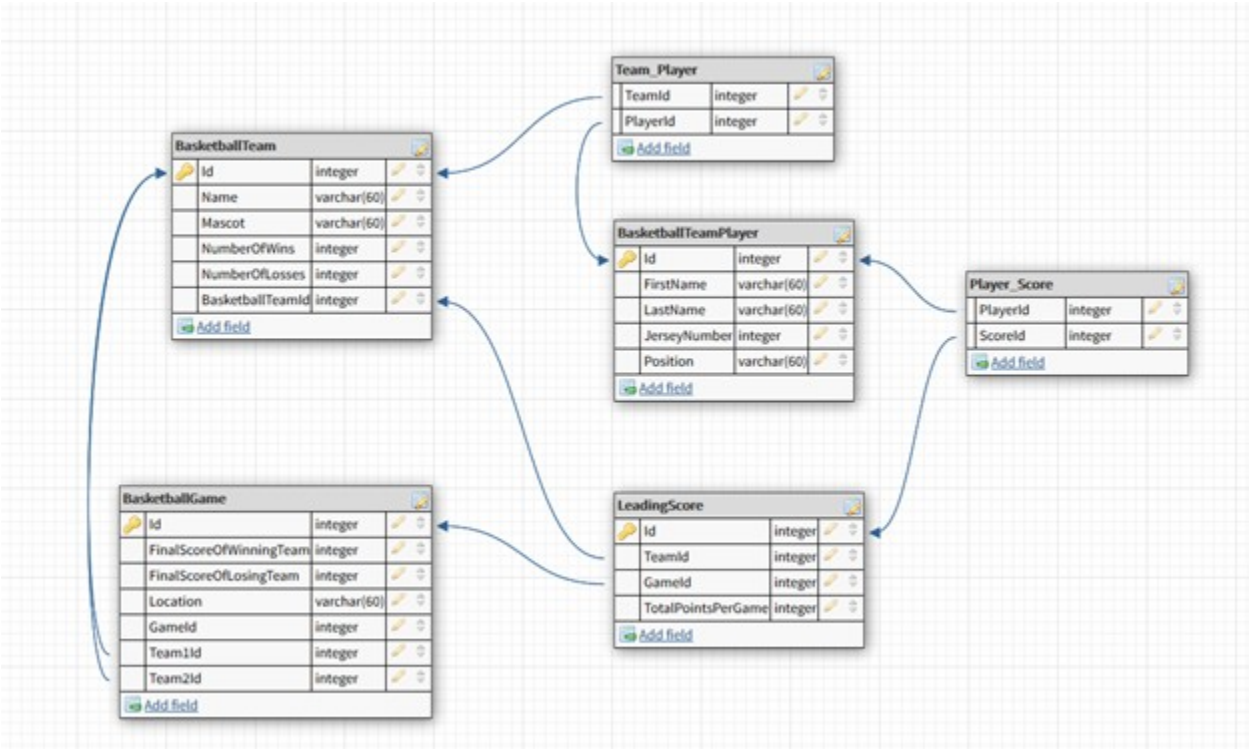
The relationships in our database are:

1. Many BasketballTeams can play many games, and BasketballGame can have many different teams play. Many to many relationship.
2. Each BasketballGame can have only one LeadingScore, and there is a LeadingScore in every BasketballGame. One to Many relationship.
3. Each BasketballTeamPlayer is part of a BasketballTeam, and a BasketballTeam can have many BasketballTeamPlayers. One to many relationship.
4. BasketballTeamPlayers can play in many BasketballGames, and each BasketballGame can have many players. Many to many relationship.

ERD


BasketballGame
PK
id
FinalScoreOfWinningTeam
FinalScoreOfLosingTeam
Location
GameId
Team1Id
Team2Id
LeadingScore
PK
id
Position
TeamId
GameId
TotalPointsPerGame
BasketballTeam
PK
id
Name
Mascot
NumberOfWins
NumerOfLoses
BasketballTeamId
BasketballPlayer
PK
id
BasketballTeamId
FirstName
LastName
JerseyNumber
Position

Schema



b) Fixes based on Feedback from Previous Steps:

TA Feedback:

We received a perfect score on our step 1, and we then got feedback on our original project casino database. We updated our project when turning in Project Step 2 final Draft Version: ERD & Schema and we did not get feedback on that part.

Peer Feedback:

we then got feedback on our original project casino database. We updated our project when turning in Project Step 2 final Draft Version: ERD & Schema and we did not get feedback on that part.

Here is the feedback we got for the casino database, but we can not use it since we changed the project.

Reviews

Database Outline

The database will record data found within a Casino. The entities will be:

- People
- Roles
- Games
- Winnings
- Time

The attributes of each entity are not correctly listed or explained as per the assignment requirements. You use the word Boss through your outline, which I believe is a typo. Why is the ID attribute of People being "assigned to each Boss"? This occurs throughout the outline and makes everything very confusing. Roles seems to be more of an attribute than an entity. In fact it seems that other than People, every other entity is being treated as if it were an attribute. There is a lot of work that still needs to be done.

ERD

The Entity-Relationship-Diagram is not designed correctly and makes absolutely no sense. How does Time have a many-to-many relationship with People?

Schema

The schema has only one entity displayed and therefore is not complete

---

Greetings Allen & Gauge....Ka-Ching! The house always wins!
Fun idea. Pretty sure I wouldn't have thought of this theme.
For People entity, I think "the boss" might find the age of the people useful. If it won't cause too much extra work, I think that would be useful information.
It wasn't clear in the schema, but the ID for people will be the primary key, right? I think it would be helpful if you guys added that information for all the entities.
I could be wrong here, but should Time be its own entity? Wouldn't it be easier to make it an attribute of ....oh wait. it wouldn't make sense for it to be an attribute of people. Maybe a relationship table with Person (player) ID and Time?
Also, why not make time TIME data type? Wouldn't that be easier to read 240 minutes vs 4 hours?
One last thing. What did you guys use for the ERD & Schema, it looks a bit blurry. If you haven't checked it out yet, I HIGHLY recommend Lucidchart.com - it's easy to use!

A casino database sounds fun!
Here are some of my thoughts:

- I think your entities should be People, Roles, Games and Visits.
- I think the Visits table should include winnings and time spent as fields. I think the visits should be in a M-M relationship with both Games and People
- Finally, I think that your ERD diagram should only include the blue boxes.
