

Suggested Solutions for Assignment 1

Consider the following relational model for a basketball league:

- Player (PlayerID, PName, Position, TeamID)
- Team (TeamID, TName, Venue)
- Game (GameNo, Date, Time, HomeTeamID, AwayTeamID)
- Record (GameNo, PlayerID, Points)

In this basketball league, each team has a unique name and each player plays for only one team. One team has at least 10 players. Two teams (home team versus away team) participate in each game at home team's venue. A team meets each of other teams twice (i.e., double round-robin tournament), one time as home team and the other time as away team. For each game, the league records the number of points scored by each player. If a player played for a game but did not score any point, there is a record for the player with zero point. If a player did not play for a game, there is no record for this player for that game (i.e., no record in the Record relation).

Please answer all the following questions based on the provided relational model and information for the basketball league.

Question 1. Draw an ER-diagram model for the basketball league. (The relationship between two entities should be 1-to-1, 1-to-many, many-to-1 or many-to-many. Do NOT use (min, max) notation for cardinality constraints. You should upload an image for your ER-diagram.) [20 marks]

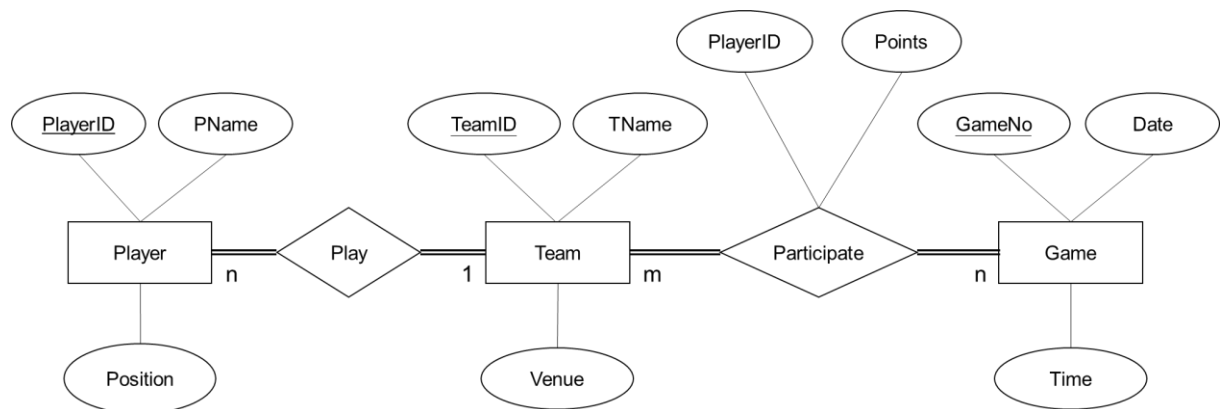
Suggested Solution:

This is just one suggested solution. The answer is correct if it is reasonable. However, the ER-diagram must show the following properties:

1. All relationships are total participations.
2. PlayerID, TeamID and GameNo are unique attributes.
3. The relationship between Player and Team is 1-to-many relationship.
4. The relationship between Team and Game is many-to-many relationship.
5. All the attributes should be involved in the ER-diagram.

Note: A correct answer may have additional entities.

It is also correct to have HomeTeamID and AwayTeamID as Game's attributes.



Question 2. Write a relational algebra that returns the PlayerID and PName of players who did not play for any game. [20 marks]

You should type notations of relational algebra as depicted in the following table.

Notations	Typing
σ	[S]
π	[P]
*	*
\bowtie	[J]
\leftarrow	\leq
ρ	[R]

Suggested Solution 1:

$\text{Tmp} \leftarrow [P]_{\text{PlayerID}} (\text{Player}) - [P]_{\text{PlayerID}} (\text{Record})$
 $\text{Result} \leftarrow [P]_{\text{PlayerID, PName}} (\text{Tmp} * \text{Player})$

(“Result \leftarrow ” is not necessary)

Suggested Solution 2:

$[P]_{\text{PlayerID, PName}} (\text{Player} * ([P]_{\text{PlayerID}} (\text{Player}) - [P]_{\text{PlayerID}} (\text{Record})))$

(There could be some other correct solutions.)

Question 3. Write a relational algebra that returns the GameNo, Venue, team name of the home team (renamed it as HomeTeamName) and team name of the away team (renamed it as AwayTeamName). [20 marks]

You should type notations of relational algebra as depicted in the following table.

Notations	Typing
σ	[S]
π	[P]
*	*
\bowtie	[J]
\leftarrow	\leq
ρ	[R]

Suggested Solution:

```

Tmp  $\leq$  [P] GameNo, Venue, TName, AwayTeamID (Game [J] HomeTeamID=TeamID Team)
Tmp2  $\leq$  [R] (GameNo, Venue, HomeTeamName, AwayTeamID)(Tmp)
Tmp3  $\leq$  [P] GameNo, Venue, HomeTeamName, TName (Tmp2 [J] AwayTeamID=TeamID Team)
Result  $\leq$  [R] (GameNo, Venue, HomeTeamName, AwayTeamName)(Tmp3)

```

("Result \leq " is not necessary)

(There could be some other correct solutions.)

Question 4. Write a SQL to retrieve the TeamID, TeamName and the number of players for each team. [20 marks]

Suggested Solution 1:

```
SELECT Team.TeamID, TName, COUNT(*)  
FROM Team, Player  
WHERE Team.TeamID=Player.TeamID  
GROUP BY Team.TeamID, TName;
```

Suggested Solution 2:

```
SELECT Player.TeamID, TName, COUNT(*)  
FROM Team, Player  
WHERE Team.TeamID=Player.TeamID  
GROUP BY Player.TeamID, TName;
```

(There could be some other correct solutions.)

Question 5. Write a SQL to retrieve the PlayerID and PName of players who played for their team in two or more games. [20 marks]

Suggested Solution 1:

```
SELECT Player.PlayerID, PName
FROM Player, Record
WHERE Player.PlayerID=Record.PlayerID
GROUP BY Player.PlayerID, PName
HAVING COUNT(*) >= 2;
```

Suggested Solution 2:

```
SELECT PlayerID, PName
FROM Player
WHERE PlayerID IN (
SELECT PlayerID
FROM Record
GROUP BY PlayerID
HAVING COUNT(*)>=2);
```

(There could be some other correct solutions.)