

## CPSC 304

### Cover Page for Project Milestone 3

**Date: Mar. 31 2022**

**Project Group Number on Canvas: 6**

#### Group Members:

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Dennis Zhu	45427168	v2d2b	denniszhu1998@gmail.com
Yilin Yang	24754350	e4l2b	Justin.jtyang@gmail.com
Oliver Meng	59212092	d8u2b	olivermeng6@gmail.com

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

1. All code used in the application:

- The webpage can be accessed at <https://www.students.cs.ubc.ca/~mshizhe/admin.php>
- All code are included in the repository below, there the script to create all tables is named "NBA.sql" [https://github.com/FreddieNeverLeft/CPSC\\_304\\_NBA](https://github.com/FreddieNeverLeft/CPSC_304_NBA)

2. How our final schema differed:

- "sid" in "stadiums" table was changed to "stid" for clarity.
- "Name" in "stadiums" table changed to "stname" for clarity.
- "date" was changed to "gdate" in "Regular" and "Playoff" table since "date" was a keyword.
- Added "ON DELETE CASCADE" for the foreign keys in the "Player\_Stats\_Only" table. We would want a player's information to be deleted when he is removed from the team, since the information stored is team specific.
- Added "ON DELETE CASCADE" for the foreign keys in the "Injury" table. We no longer want to record a player's injury information when that player is deleted/retired.
- Moderate split to moderate\_regular and moderate\_playoff so we can easily distinguish between which type of game is regulated by which referee.

The goal of our project is to capture and represent the events that happen throughout the NBA season. We created a database that focuses on recording the team/player statistics after each game and the corresponding effect it has on the other administration/operational entities that support the league.

As a result, moderators and users of our project can update the database after each game played throughout the season, and query the statistics (ie. player's game stat, favorite team's statistics, the playoff standings across conferences) that interest them. People who use our database will finally have a central hub where they have access to all the data related to the NBA, and discover new relations that otherwise wouldn't have been obvious.

When implementing our design, we made some improvements to the schema so that the relationships and attributes are easier to understand. For starters we changed the "sid", "name" attribute names in the "Stadium" table into "stid" and "sname" for clarity, and changed the "date" attribute name in the "Regular"/"Playoff" tables into "gdate" since "date" was already a key word.

Furthermore, we realized some of our foreign key relationships did not fully explain the dependency between our entities. We added "ON DELETE CASCADE" to the foreign key relation (with "Player\_Team\_Name") in the "Player\_Stats\_Only" table because we want a player's information to be deleted when he is removed from the team (ie. lost his job, out of the league, career over). We also added the same command to the foreign key relation (with "Player\_Stats\_Only") in the "Injury" table because we no longer want to record a player's injury information when that player is deleted/retired.

Lastly we split the "Moderate" table into "moderate\_regular" and "moderate\_playoff" so we can more easily distinguish which type of game is regulated by which referee.

## Insert Operation :

Request:

### (Insert Operation) Insert Values into Coach

Name:

Salary: \$

Result:

Before Query:

Retrieved data from table Coach:

Name	Salary
Tom Thibodeau	\$4375000
Frank Vogel	\$4000000
Doc Rivers	\$12365000
Billy Donovan	\$5000000
Monty Williams	\$10000000

After Query:

Retrieved data from table Coach:

Name	Salary
Tom Thibodeau	\$4375000
Frank Vogel	\$4000000
Doc Rivers	\$12365000
Billy Donovan	\$5000000
Monty Williams	\$10000000
Oliver	\$100

### Update Operation:

Now that we added Oliver, we will update it to Justin.

Request:

## (Update Operation) Update Name in Coach

The values are case sensitive.

Old Name:

New Name:

Result:

Before Query:

Retrieved data from table Coach:

Name	Salary
Tom Thibodeau	\$4375000
Frank Vogel	\$4000000
Doc Rivers	\$12365000
Billy Donovan	\$5000000
Monty Williams	\$10000000
Oliver	\$100

After Query:

Retrieved data from table Coach:

Name	Salary
Tom Thibodeau	\$4375000
Frank Vogel	\$4000000
Doc Rivers	\$12365000
Billy Donovan	\$5000000
Monty Williams	\$10000000
Justin	\$100

### Delete Operation:

Delete Player with ID = 3 from Player\_stats\_only, cascade removes their injury as well.

Before Query:

Player ID	Team Name
-----------	-----------

1	Lakers
2	Suns
3	76ers
4	Bulls
5	Knicks

Player ID	Injury Name
-----------	-------------

1	Knee Soreness
1	Back Spasms
3	Ankle Sprain
2	Back Soreness
5	Neck Soreness

After Query:

Player ID	Team Name
-----------	-----------

1	Lakers
2	Suns
4	Bulls
5	Knicks

Player ID	Injury Name
-----------	-------------

1	Knee Soreness
1	Back Spasms
2	Back Soreness
5	Neck Soreness

### Count Operation:

Counts the number of coaches

Retrieved data from table Coach:

Name	Salary
Tom Thibodeau	\$4375000
Frank Vogel	\$4000000
Doc Rivers	\$12365000
Billy Donovan	\$5000000
Monty Williams	\$10000000

The number of tuples in coach: 5

### Selection Operation:

Select the score for the games where the Lakers won at home. For simplicity, we only display the two teams and their corresponding scores when showing all regular games.

All Regular Games:

Home Team Name	Home Points	Away Points	Away Team Name
Knicks	111	98	Lakers
Lakers	123	88	Bulls
Bulls	108	97	76ers
76ers	122	125	Suns
Suns	98	125	Knicks

Score for the regular home games where the Lakers won:

Team Name	Points
Lakers	123

### Project Operation:

Project and find the scores of regular games:

Home Team Name	Home Points	Away Points	Away Team Name
Knicks	111	98	Lakers
Lakers	123	88	Bulls
Bulls	108	97	76ers
76ers	122	125	Suns
Suns	98	125	Knicks

### Join Operation:

Find the name & number of players who are American by joining the player\_team\_name/player\_stat\_only table(The tables are joined by player number and team name since they are the primary key):

Before Query:

Player name, Player number, Team Name from Player\_Team\_Name:

Player Name	Player Number	Team
Demar Derozan	11	Bulls
Joel Embiid	21	76ers
Lebron James	6	Lakers
Chris Paul	3	Suns
RJ Barrett	9	Knicks

Player number, Team name, and Nationality from Player\_Stats\_Only:

Player Number	Team	Nationality
6	Lakers	American
3	Suns	American
21	76ers	American
11	Bulls	American
9	Knicks	Canadian

After Query:

All American players:

Player Name	Player Number	Nationality
Demar Derozan	11	American
Joel Embiid	21	American
Lebron James	6	American
Chris Paul	3	American



**Aggregation Operation:**

Find the what the average amount of points each team scored at home:

<b>Team Name</b>	<b>Average Home Points</b>
Lakers	123
76ers	122
Knicks	111
Suns	98
Bulls	108

**Nested Aggregation:**

Find the team that has players with the lowest shooting percentage(average)

Each team's players average shooting percentage:

<b>Team Name</b>	<b>Average Shooting Percentage</b>
Lakers	52
76ers	49
Knicks	42
Suns	49
Bulls	52

The team with the lowest players average shooting percentage:

<b>Team Name</b>	<b>Average Shooting Percentage</b>
Knicks	42

**Performing the above query using views:**

We obtain the same result and we can verify the above is correct.

<b>Team Name</b>	<b>Average Shooting Percentage</b>
Knicks	42

**Division:**

Find the regular games that were officiated by all referees with more than 20 years of experience. From the Referee table, we can see that we need to find games that are regulated by both Referee 3 and 9. Game 3 is indeed the only game that matches the query.

Regular Games table and Referee Table:

Game ID	Referee ID
---------	------------

1	1
2	2
3	3
3	9
4	4
5	5

Referee ID	Years of Experience
------------	---------------------

1	5
2	5
3	26
4	13
5	14
6	7
7	10
8	11
9	28

After Query:

Game ID
---------

3
---