**NBA DATABASE REPORT**

1. **Application description:** For our project, we decided to implement a database that can represent the National Basketball Association (NBA). More specifically, we included players, teams, coaches, arenas, owners, and the two conferences. Obviously it would take a lot of time to add all of the players in the NBA as well as all of the coaches, so instead we just took the list of players who made the All Star game last year and the head coach for each of the 30 teams. However, we did use all 30 teams, all arenas, and we used one owner for each team. Teams with multiple owners were indicated with etc. at the end of the owner name. This database could be used for many different things such as finding players that meet certain statistical criteria or finding teams with a certain amount of championships. The possibilities for this database are endless and it could be used in a similar way to something like Basketball Reference (a home for all kinds of basketball information for anyone who is interested).
2. **Conceptual model/E-R diagram:**

Diagram

Description automatically generated

1. **Initial Database schema:**

Table

Description automatically generated

1. **Final Database schema:**

Owner table:

Table

Description automatically generated

Player table:

Table

Description automatically generated

Team table:

Table

Description automatically generated

Arena table:

Table

Description automatically generated

Coach table:

Table

Description automatically generated

Conference table:

Table

Description automatically generated

This database schema holds under all four properties of good database design because all attributes are preserved, all dependencies are preserved, all relations are in 3rd normal form, and all joins are lossless. This is because all of the attributes and dependencies are from the real world application we are modeling (the NBA). Additionally, there is not a transitive dependency for non-prime attributes, and it is in 2nd normal form. Finally, decomposing and rejoining relations would yield the original relation. So, all four properties of good database design hold. No normalization took place.

1. **Database instance:**

Player table:

Table

Description automatically generated with medium confidenceTable

Description automatically generated

Coach table:

Table

Description automatically generatedTable

Description automatically generated

Team table:

Table

Description automatically generatedTable

Description automatically generated

Owners table:

Table

Description automatically generatedTable

Description automatically generated

Arena table:

Table

Description automatically generatedTable

Description automatically generated

Conference table:

Graphical user interface, text, table

Description automatically generated with medium confidence

1. **Data manipulation:**

Query 1: Retrieve team names with multiple owners

Graphical user interface, text, application, email

Description automatically generated

Query 2: Retrieve team name, number of championships, and coach name of teams who have had the same coach for 10 or more years

Graphical user interface, text, application, email

Description automatically generated

Query 3: Retrieve team name of teams with no 2022 All Stars (no players in the database)

Graphical user interface, text, application, email

Description automatically generated

Query 4: Retrieve player information from all players who play for each of the last conference champions

Text

Description automatically generated

Query 5: Retrieve arena name, arena location, team name, team location from teams that do not have the same exact team and arena location

Text

Description automatically generated

Query 6: Retrieve player names and stats of players who score more than 25 points per game and get more than 5 assists and rebounds per game

Text

Description automatically generated

Query 7: Retrieve the player name, team, and points per game of players that score more than 30 points per game and sort it by division

Text

Description automatically generated with medium confidence

Query 8: Retrieve the name, team, and years owned of owners of teams that have no championships

A picture containing graphical user interface

Description automatically generated

Query 9: Retrieve the team name of teams with more than one 2022 All Star (player in the database) as well as the count

Graphical user interface, text, application, Word

Description automatically generated

Query 10: Retrieve the name, position, and height of players between 6’5 and 6’10

A picture containing graphical user interface

Description automatically generated

1. **Observations:**

Some observations we had were that it was pretty interesting to be able to take something fun and entertaining for a lot of people (such as sports) and turn it into a database that could have legitimate uses. We found that the toughest part of the project was just coming up with an idea that fit all of the requirements and then turning this idea into an actual database schema that can be implemented. In terms of actually getting the tables into Oracle, the only issue was that we eventually realized the order of the tables had to be switched around for it to work so that foreign keys could be properly defined.  Other than this, creating and importing the schema file was easier than expected. One thing that we found to be particularly annoying was the process of just finding all of the data for each table. Additionally, we ran into issues importing the data file in Oracle but running individual insert commands worked so we had to take extra time to do all of the insertions individually. Overall we found it generally enjoyable to take a real world organization like the NBA and turn it into something that we have been studying all semester.

NOTE: slides and schema and data SQL files will be included in submission as separate files.