CPSC 304 Final Project Report

Jacob Lee Norman Sue Daniel Tsang Marvin Cadano

Project hosted at http://54.86.9.29/

Contents

1 Project Description **7**

Platforms . . . 7

System functionality . . . 7

players.php (Jacob) . . . 7

profiles.php (Jacob) . . . 7

teams.php (Marvin) . . . 8

rosters.php (Jacob) . . . 8

games.php (Daniel) . . . 8

stats.php (Jacob) . . . 8

update.php (Norman) . . . 8

venues.php (Norman) . . . 8

2 Entity-Relationship Diagram **9**

3 SQL Table Schemas **11**

Legend . . . 11

Division . . . 11

Venue . . . 11

NBATeam_BelongsTo . . . 12

NBAPlayer_PlaysFor . . . 12

NBAStaff_WorksFor . . . 12

Sponsor_Endorses . . . 13

NBAGame_Plays_PlayedAt . . . 13

NBAReferee . . . 14

Referees . . . 14

4 SQL Queries **15**

Selection and Projection Queries . . . 15
Join Queries . . . 16
Division Query . . . 16
Aggregation Query . . . 18
Nested Aggregation with Group-By Query . . . 18
Delete Operation . . . 18
Update Operation . . . 19
Other Queries and Extra Features . . . 21

5 Functional Dependencies **25**

Division . . . 25
Venue . . . 25
NBATeam_BelongsTo . . . 25
NBAPlayer_PlaysFor . . . 25
NBAStaff_WorksFor . . . 26
Sponsor_Endorses . . . 26
NBAGame_Plays_PlayedAt . . . 26
NBAReferee . . . 26
Referees . . . 26

6 SQL Table Contents **27**

Division . . . 27
Venue . . . 27
NBATeam_BelongsTo . . . 28
NBAPlayer_PlaysFor . . . 28
NBAStaff_WorksFor . . . 28
Sponsor_Endorses . . . 29
NBAGame_Plays_PlayedAt . . . 29
NBAReferee . . . 29
Referees . . . 30

7 Demo Script **31**

Before the demo . . . 31
Setup (2 minutes) . . . 31
Selection and Projection Query (2 minutes) . . . 31
Join Query (2 minutes) . . . 32
Division Query (2 minutes) . . . 32

Aggregation Query (2 minutes) 32
Nested Aggregation with Group By (2 minutes) 32
Delete Operation (4 minutes) 33
Update Operation (2 minutes) 33
Extra Features (2 minutes) 34

Chapter 1

Project Description

Our project can be found online at http://54.86.9.29/

We modelled the 2013-2014 NBA league using real data for the most part, and randomly generated data for certain data that we could not easily find.

We modelled NBA players that play for Teams, along with NBA Staff that work for Teams. Each Team belongs to a Division. Each Team plays a game refereed by an NBA Referee against another Team at a Venue on a particular date. Each Team's Sponsor is also modeled.

User classes include NBA Staff and normal users. Only NBA Staff can modify database information. All other users can only view database information. We did not actually implement any user access control, but included functionality for both user classes in our UI.

Platforms

- Amazon EC2 virtual server running Ubuntu 12.04.4 LTS
- Apache HTTP Server 2.4
- MySQL 5.5.36
- PHP 5.4.26 (with the PDO extension)

System functionality

Below is a listing of the specific functionalities we implemented in each of our PHP pages.

players.php (Jacob)

- Displays a table of all players including information about each player
- Includes an HTML form for choosing attributes (projection), filtering results (selection), and ordering results (order by)
- Includes an option to delete a player

profiles.php (Jacob)

- Lists all information about a specific player

- Includes a dropdown to select from a list of all players
- Retrieves and displays first Google image result of that player
- Retrieves and dipslays relevant news articles for that player

teams.php (Marvin)

- Displays a table of all teams including information about each team and their division
- Includes an option to delete a team (deletion should be rejected if team has at least one player)

rosters.php (Jacob)

- Displays a table of players on a team including information about each player
- Displays a table of staff on a team including information about each staff member
- Displays a table of sponsors for a team
- Includes a dropdown menu to select from a list of all teams

games.php (Daniel)

- Shows a table of games along with the venue the game was played at and who refereed the game
- Displays an HTML form to only show games between two dates

stats.php (Jacob)

- Displays an HTML form to pick max/min/average of height, weight, number, draft year stats for ALL players
- Displays an HTML form to pick max/min/average of height, weight, number, draft year stats for players of EACH team where that team is sponsored by a selected company

update.php (Norman)

- Displays an HTML form to update information about a specific player

venues.php (Norman)

- Displays a table of all venues
- Displays a table of teams that have played at all venues

Chapter 2

Entity-Relationship Diagram

The only change in our ER diagram since the second project submission was relaxing a full participation constraint on NBAGame and the Referees relationship so that we could demonstrate both a situation where a cascading delete works and one where it doesn't (for the purposes of meeting project demo requirements).

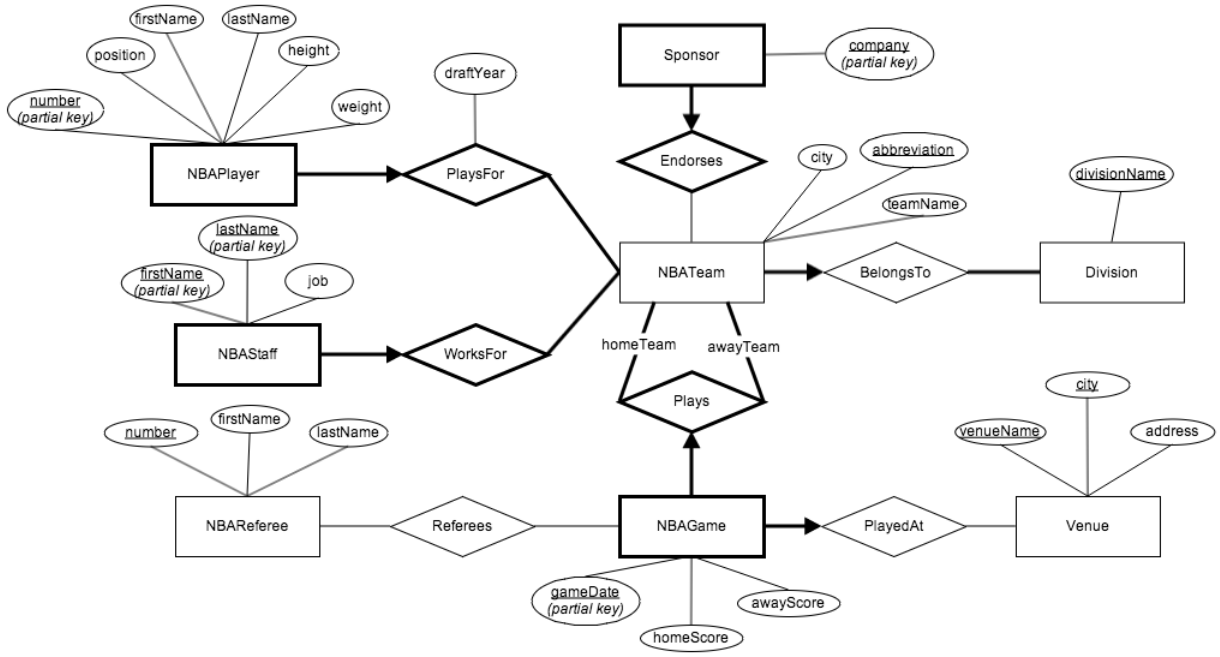

Figure 2.1: Entity-relationship diagram

Chapter 3

SQL Table Schemas

No changes were made to the schemas of our SQL tables, which are repeated below for convenience. Only additional `CHECK` constraints were added to the `CREATE TABLE` commands in the included `create_db.sql` script.

Our tables capture all aspects of our ER diagram except for the full participation constraints on NBATeam for the PlaysFor and WorksFor relationships.

Legend

- `PRIMARY KEY` = **Bold**
- `FOREIGN KEY` = *Italics*
- `PRIMARY KEY` and `FOREIGN KEY` = ***Bold Italics***

Division

Division(**divisionName**: VARCHAR(10))

```
CREATE TABLE `Division` (
    `divisionName` VARCHAR(10),
    PRIMARY KEY (`divisionName`)
);
```

Venue

Venue(**venueName**: VARCHAR(30), **city**: VARCHAR(30), address: VARCHAR(50))

```
CREATE TABLE `Venue` (
    `venueName` VARCHAR(30),
    `city` VARCHAR(30),
    `address` VARCHAR(50),
    PRIMARY KEY (`venueName`, `city`)
);
```

NBATeam_BelongsTo

NBATeam_BelongsTo(**abbreviation**: CHAR(3), city: VARCHAR(30), teamName: VARCHAR(30), *divisionName*: VARCHAR(10))

- *divisionName* `NOT NULL`
- *divisionName* `REFERENCES` Division

```
CREATE TABLE `NBATeam_BelongsTo` (
    `abbreviation` CHAR(3),
    `city` VARCHAR(30),
    `teamName` VARCHAR(30),
    `divisionName` VARCHAR(10) NOT NULL,
    PRIMARY KEY (`abbreviation`),
    FOREIGN KEY (`divisionName`) REFERENCES `Division` (`divisionName`)
);
```

NBAPlayer_PlaysFor

NBAPlayer_PlaysFor(**number**: INT, position: CHAR(2), firstName: VARCHAR(30), lastName: VARCHAR(30), height: INT, weight: INT, draftYear: YEAR(4), ***team***: CHAR(3))

- ***team*** `NOT NULL`
- ***team*** `REFERENCES` NBATeam_BelongsTo(**abbreviation**) `ON DELETE CASCADE`
- height in inches
- weight in pounds

```
CREATE TABLE `NBAPlayer_PlaysFor` (
    `number` INT,
    `position` CHAR(2),
    `firstName` VARCHAR(30),
    `lastName` VARCHAR(30),
    `height` INT,
    `weight` INT,
    `draftYear` YEAR(4),
    `team` CHAR(3) NOT NULL,
    PRIMARY KEY (`number`, `team`),
    FOREIGN KEY (`team`) REFERENCES `NBATeam_BelongsTo` (`abbreviation`)
        ON DELETE CASCADE,
    CHECK (`draftYear` >= 1946 AND `draftYear` <= YEAR(CURDATE()) AND
        `number` >= 0 AND `number` <= 99 AND
        `height` >= 0 AND `weight` >= 0)
);
```

NBAStaff_WorksFor

NBAStaff_WorksFor(**firstName**: VARCHAR(30), **lastName**: VARCHAR(30), job: VARCHAR(30), ***team***: CHAR(3))

- ***team*** NOT NULL
- ***team*** REFERENCES NBATeam_BelongsTo(**abbreviation**) ON DELETE CASCADE

```
CREATE TABLE `NBAStaff_WorksFor` (
    `firstName` VARCHAR(30),
    `lastName` VARCHAR(30),
    `job` VARCHAR(30),
    `team` CHAR(3) NOT NULL,
    PRIMARY KEY (`firstname`, `lastname`, `team`),
    FOREIGN KEY (`team`) REFERENCES `NBATeam_BelongsTo` (`abbreviation`)
        ON DELETE CASCADE
);
```

Sponsor_Endorses

Sponsor_Endorses(**company**: VARCHAR(30), ***team***: CHAR(3))

- ***team*** NOT NULL
- ***team*** REFERENCES NBATeam_BelongsTo(**abbreviation**) ON DELETE CASCADE

```
CREATE TABLE `Sponsor_Endorses` (
    `company` VARCHAR(30),
    `team` CHAR(3) NOT NULL,
    PRIMARY KEY (`company`, `team`),
    FOREIGN KEY (`team`) REFERENCES `NBATeam_BelongsTo` (`abbreviation`)
        ON DELETE CASCADE
);
```

NBAGame_Plays_PlayedAt

NBAGame_Plays_PlayedAt(**gameDate**: DATE, homeScore: INT, awayScore: INT, ***homeTeam***: CHAR(3), ***awayTeam***: CHAR(3), *venueName*: VARCHAR(30), *city*: VARCHAR(30))

- ***homeTeam*** NOT NULL
- ***awayTeam*** NOT NULL
- *venueName* NOT NULL
- *city* NOT NULL
- ***homeTeam*** REFERENCES NBATeam_BelongsTo(**abbreviation**) ON DELETE CASCADE
- ***awayTeam*** REFERENCES NBATeam_BelongsTo(**abbreviation**) ON DELETE CASCADE
- *venueName*, *city* REFERENCES Venue

```
CREATE TABLE `NBAGame_Plays_PlayedAt` (
    `gameDate` DATE,
    `homeScore` INT,
    `awayScore` INT,
    `homeTeam` CHAR(3) NOT NULL,
    `awayTeam` CHAR(3) NOT NULL,
    `venueName` VARCHAR(30) NOT NULL,
    `city` VARCHAR(30) NOT NULL,
```

```
    PRIMARY KEY (`gameDate`, `homeTeam`, `awayTeam`),
    FOREIGN KEY (`homeTeam`) REFERENCES `NBATeam_BelongsTo` (`abbreviation`)
        ON DELETE CASCADE,
    FOREIGN KEY (`awayTeam`) REFERENCES `NBATeam_BelongsTo` (`abbreviation`)
        ON DELETE CASCADE,
    FOREIGN KEY (`venueName`, `city`) REFERENCES `Venue` (`venueName`, `city`),
    CHECK (`gameDate` >= '1946-01-11' AND
        `homeScore` >= 0 AND `awayScore` >= 0)
);
```

NBAReferee

NBAReferee(**number**: INT, firstName: VARCHAR(30), lastName: VARCHAR(30))

```
CREATE TABLE `NBAReferee` (
    `number` INT,
    `firstName` VARCHAR(30),
    `lastName` VARCHAR(30),
    PRIMARY KEY (`number`),
    CHECK (`number` >= 0 AND `number` <= 99)
);
```

Referees

Referees(***refNumber***: INT, ***gameDate***: DATE, ***homeTeam***: CHAR(3), ***awayTeam***: CHAR(3))

- ***refNumber*** `REFERENCES` NBAReferee(**number**)
- ***gameDate***, ***homeTeam***, ***awayTeam*** `REFERENCES` NBAGame_Plays_PlayedAt(**gameDate**, ***homeTeam***, ***awayTeam***)

```
CREATE TABLE `Referees` (
    `refNumber` INT,
    `gameDate` DATE,
    `homeTeam` CHAR(3),
    `awayTeam` CHAR(3),
    PRIMARY KEY (`refNumber`, `gameDate`, `homeTeam`, `awayTeam`),
    FOREIGN KEY (`refNumber`) REFERENCES `NBAReferee` (`number`),
    FOREIGN KEY (`gameDate`, `homeTeam`, `awayTeam`)
        REFERENCES `NBAGame_Plays_PlayedAt` (`gameDate`, `homeTeam`, `awayTeam`)
);
```

Chapter 4

SQL Queries

Note that these are only example queries. The attributes in the SELECT clauses and the conditions in the WHERE clauses may vary depending on the user's input.

Selection and Projection Queries

File location: `players.php`

```
SELECT lastName, firstName, position, number, height, weight, draftYear, team
FROM NBAPlayer_PlaysFor
WHERE height > 80
ORDER BY lastName;
```

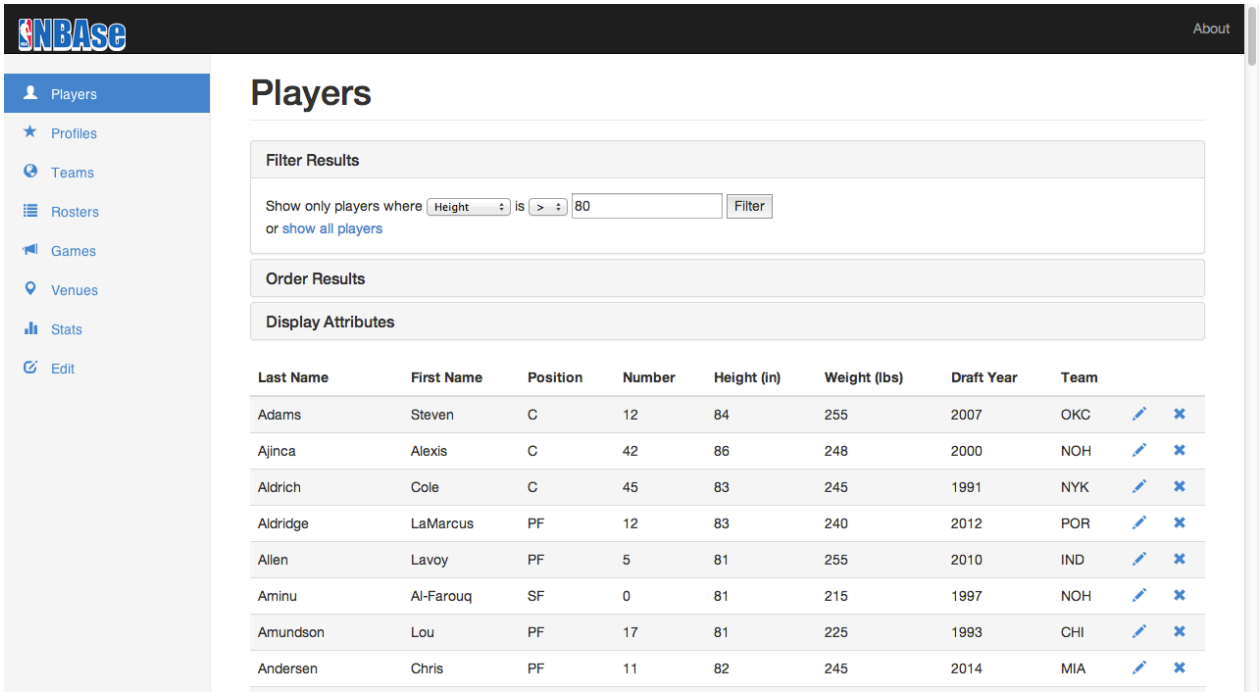

Figure 4.1: Selection query in players.php

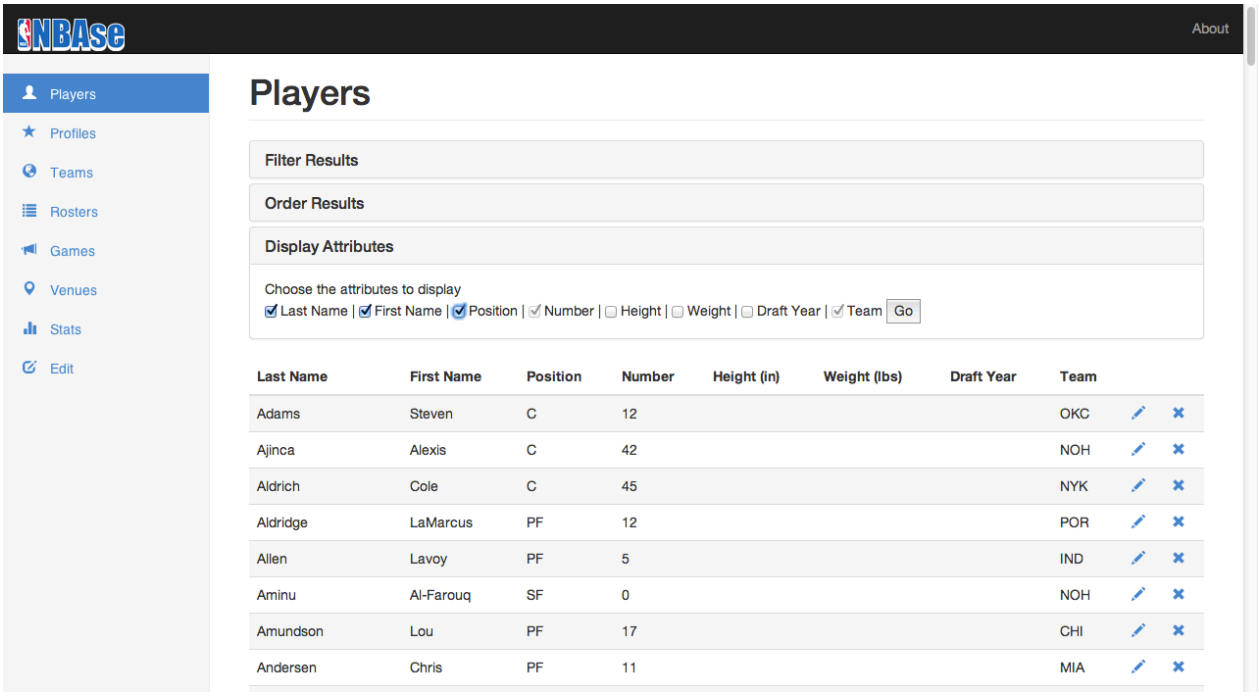

Figure 4.2: Projection query in players.php

Join Queries

File location: `games.php`

```
SELECT *
FROM NBAGame_Plays_PlayedAt NPP, NBAReferee NR, Referees R
WHERE NR.number = R.refNumber AND
      R.gameDate = NPP.gameDate AND
      R.homeTeam = NPP.homeTeam AND
      R.awayTeam = NPP.awayTeam
ORDER BY npp.gameDate
```

Division Query

File location: `venues.php`

```
SELECT *
FROM NBATeam_BelongsTo T
WHERE NOT EXISTS
    (SELECT V.venueName
    FROM Venue V
    WHERE NOT EXISTS
        (SELECT DISTINCT G.venueName
        FROM NBAGame_Plays_PlayedAt G
        WHERE V.venueName = G.venueName AND
                (G.homeTeam = T.abbreviation OR G.awayTeam = T.abbreviation)))
```

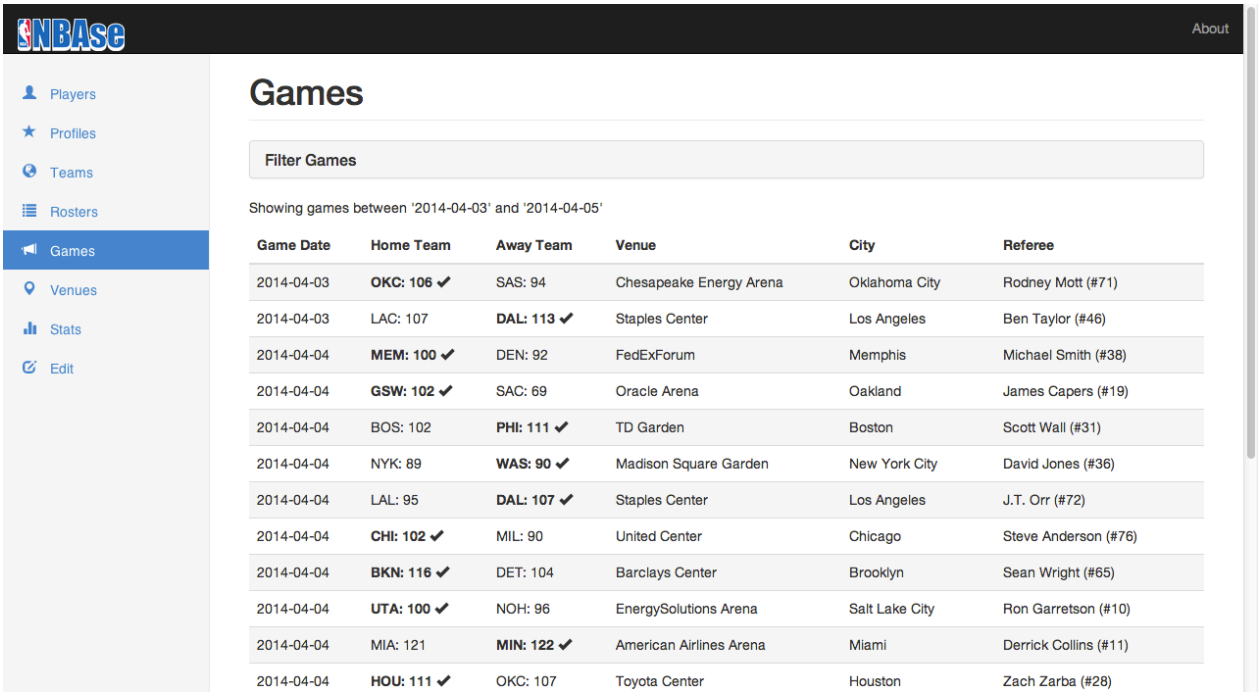

Figure 4.3: Join query in games.php

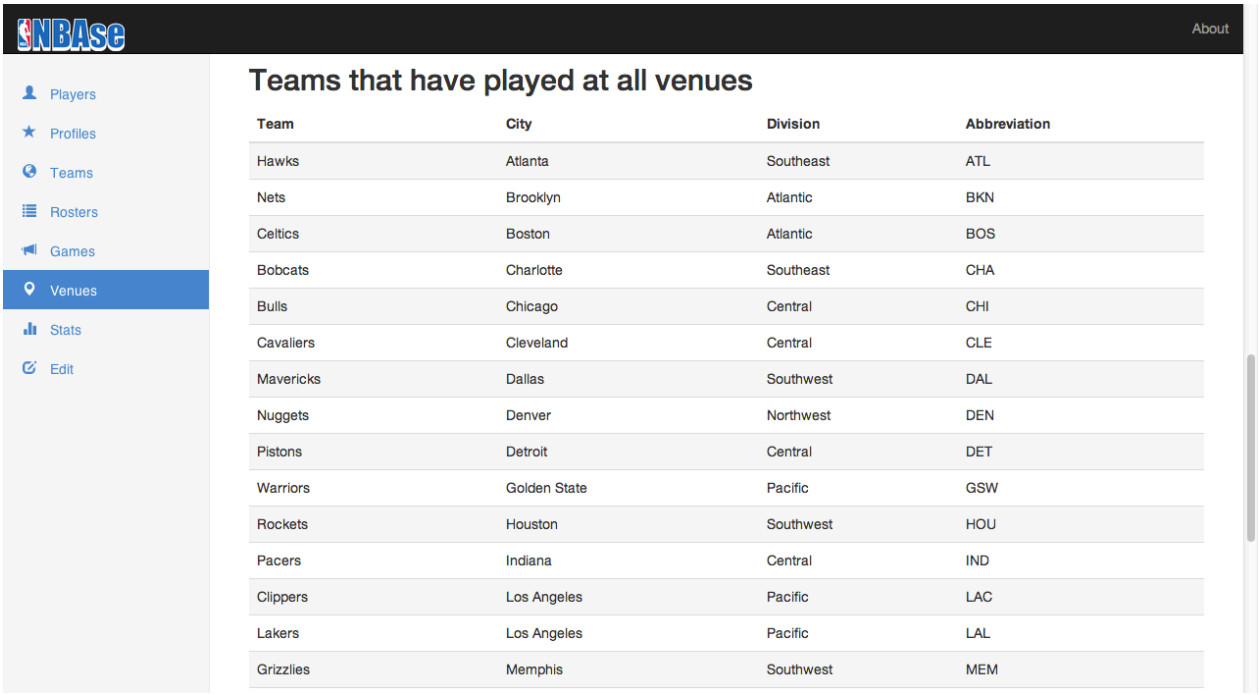

Figure 4.4: Division query in venue.php

Aggregation Query

File location: `stats.php`

```
SELECT MAX(height) as result
FROM NBAPlayer_PlaysFor
```

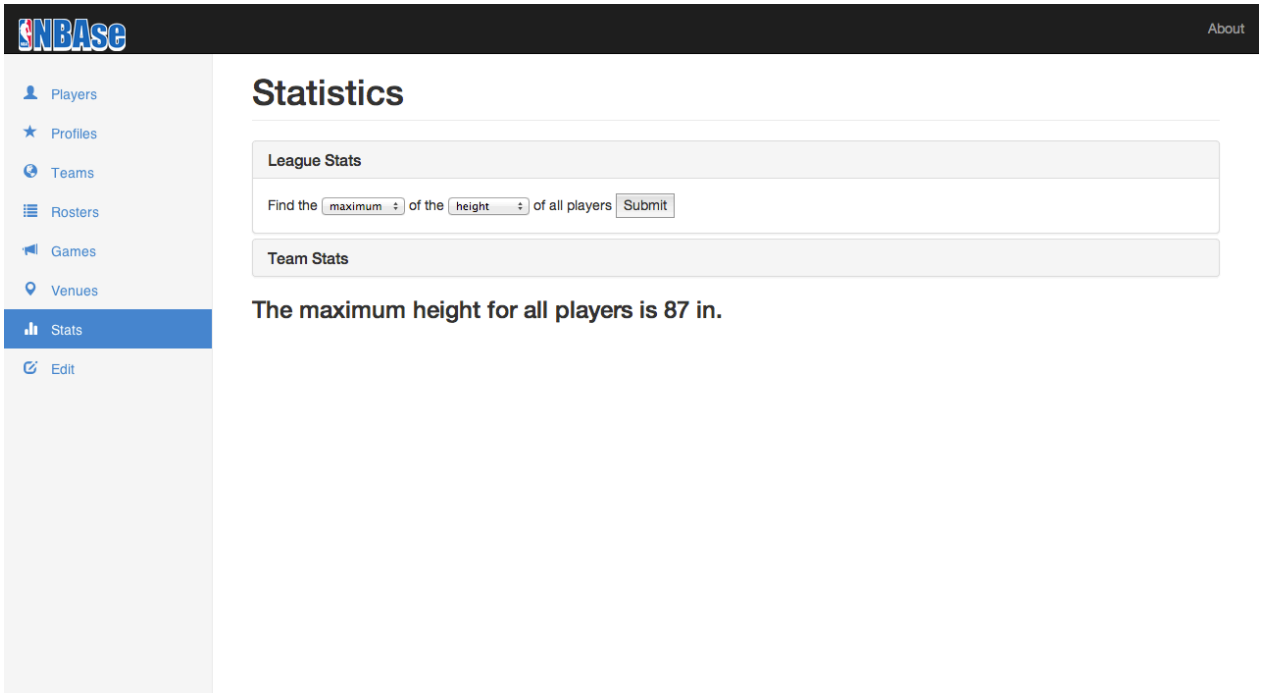

Figure 4.5: Aggregation query in stats.php

Nested Aggregation with Group-By Query

File location: `stats.php`

```
SELECT team, AVG(draftYear) as result
FROM NBAPlayer_PlaysFor
WHERE team IN
    (SELECT abbreviation
    FROM NBATeam_BelongsTo N, Sponsor_Endorses S
    WHERE N.abbreviation = S.team AND S.company = 'Mercedes Benz')
GROUP BY team
```

Delete Operation

File location: `delete_team.php`

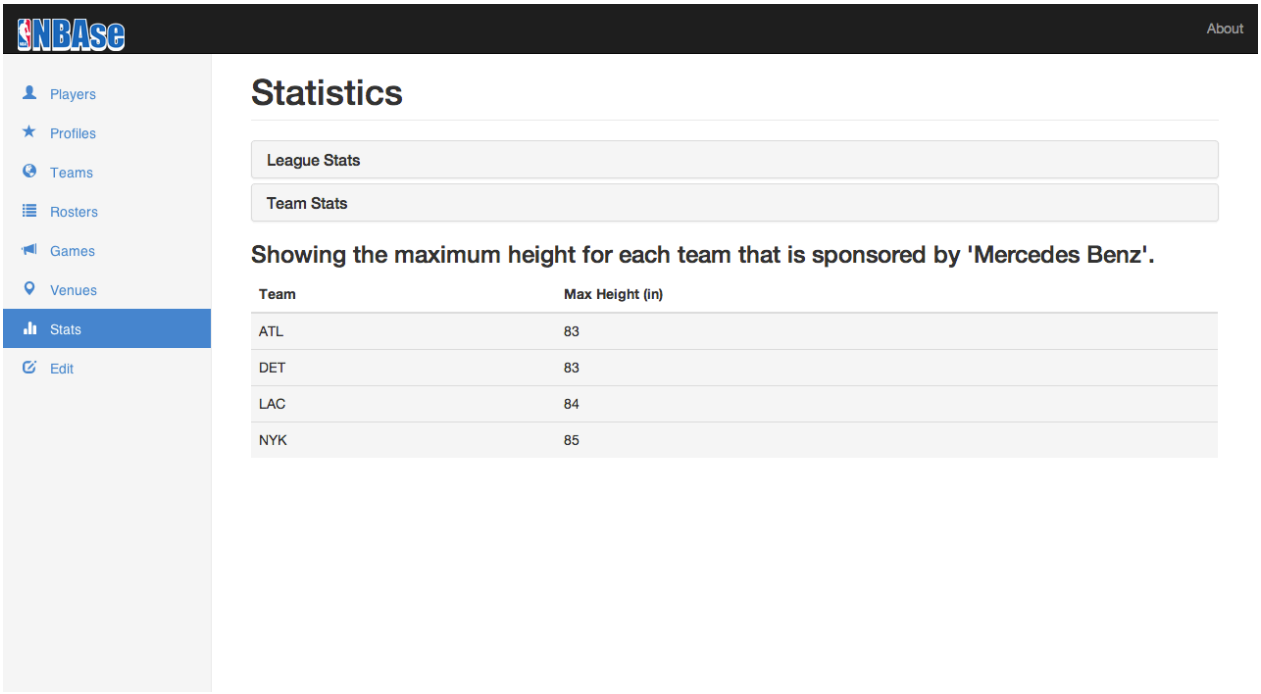

Figure 4.6: Nested aggregation with group-by query in stats.php

```
DELETE FROM NBATeam_BelongsTo
WHERE abbreviation = 'TOR'
```

File location: `delete_player.php`

```
DELETE FROM NBAPlayer_PlaysFor
WHERE team = 'DAL' AND number = 41
```

Update Operation

File location: `update.php`

```
UPDATE NBAPlayer_PlaysFor
SET firstName = 'Aaron',
    lastName = 'Brooks',
    team = 'DEN',
    number = 0,
    position = PG,
    weight = 161,
    height = 84,
    draftYear = 2012
WHERE team = 'DEN' AND number = 0;
```

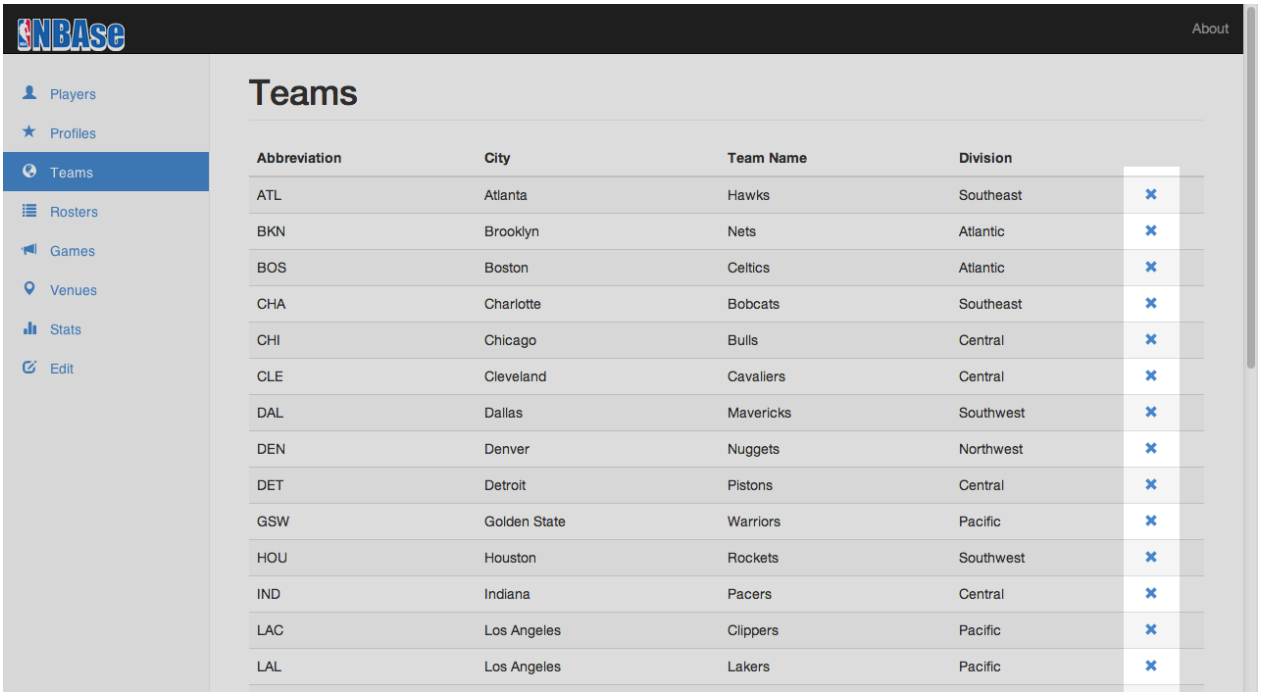

Figure 4.7: Delete operation in delete_team.php

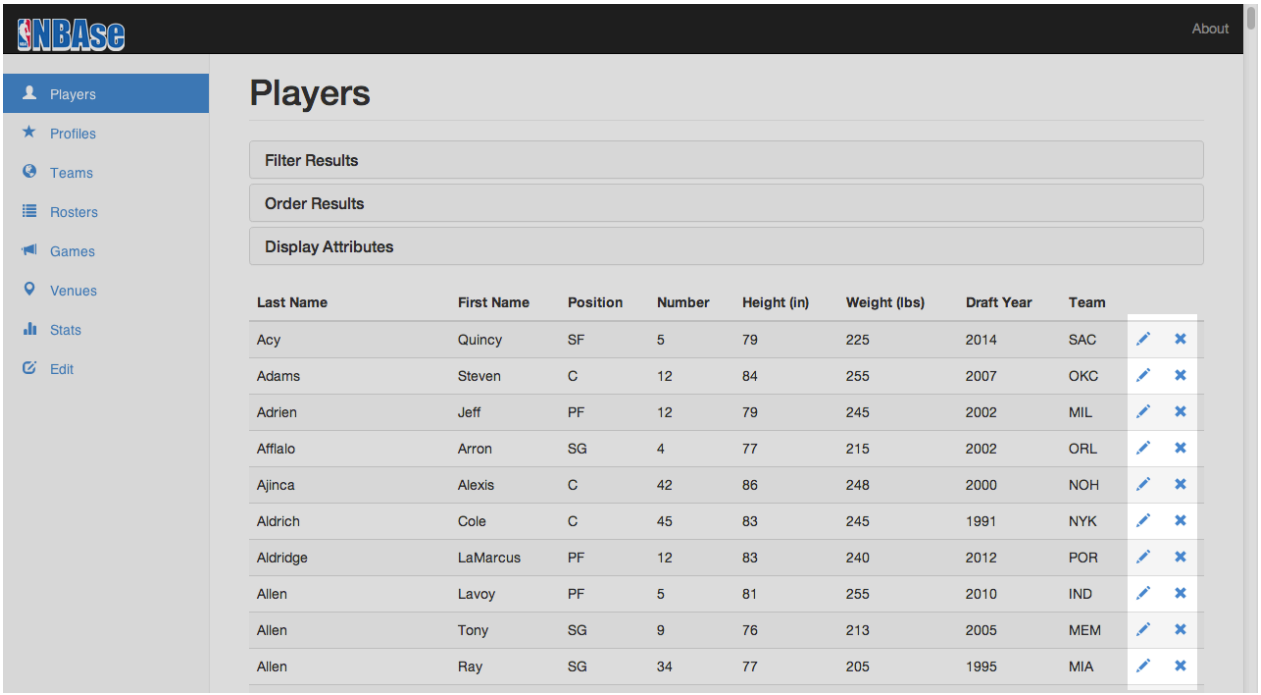

Figure 4.8: Delete operation in delete_player.php

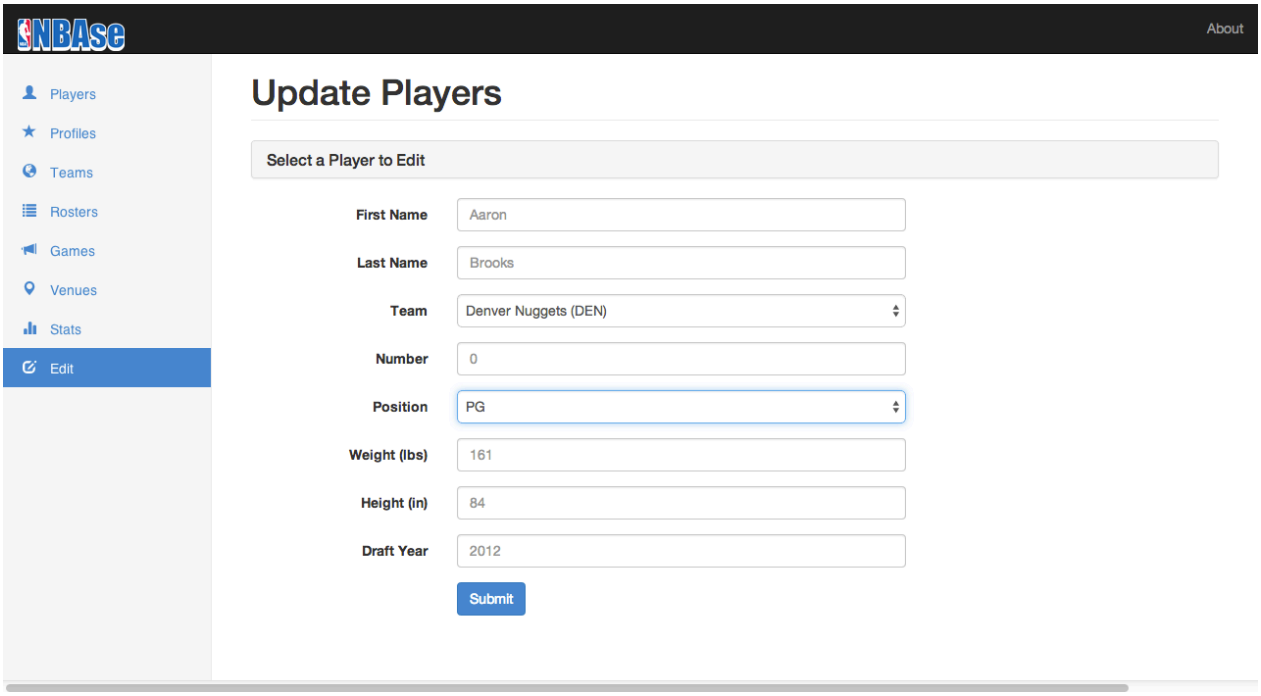

Figure 4.9: Update operation in update.php

Other Queries and Extra Features

These are miscellaneous queries that were used in the project that were not explicitly required, but included here for completeness.

We also we implemented retrieval of Google Images and Google News articles for each player as shown in the following screenshot.

File location: `venues.php`

```
INSERT INTO Venue
VALUES('Rogers Arena', 'Vancouver', '800 Griffiths Way')
```

File location: `rosters.php`

```
SELECT *
FROM NBAPlayer_PlaysFor
WHERE team = 'TOR';

SELECT *
FROM NBAStaff_WorksFor
WHERE team = 'TOR'
ORDER BY job DESC;

SELECT company
FROM NBATeam_BelongsTo N, Sponsor_Endorses S
WHERE N.abbreviation = S.team AND team = 'TOR';
```

File location: `teams.php`

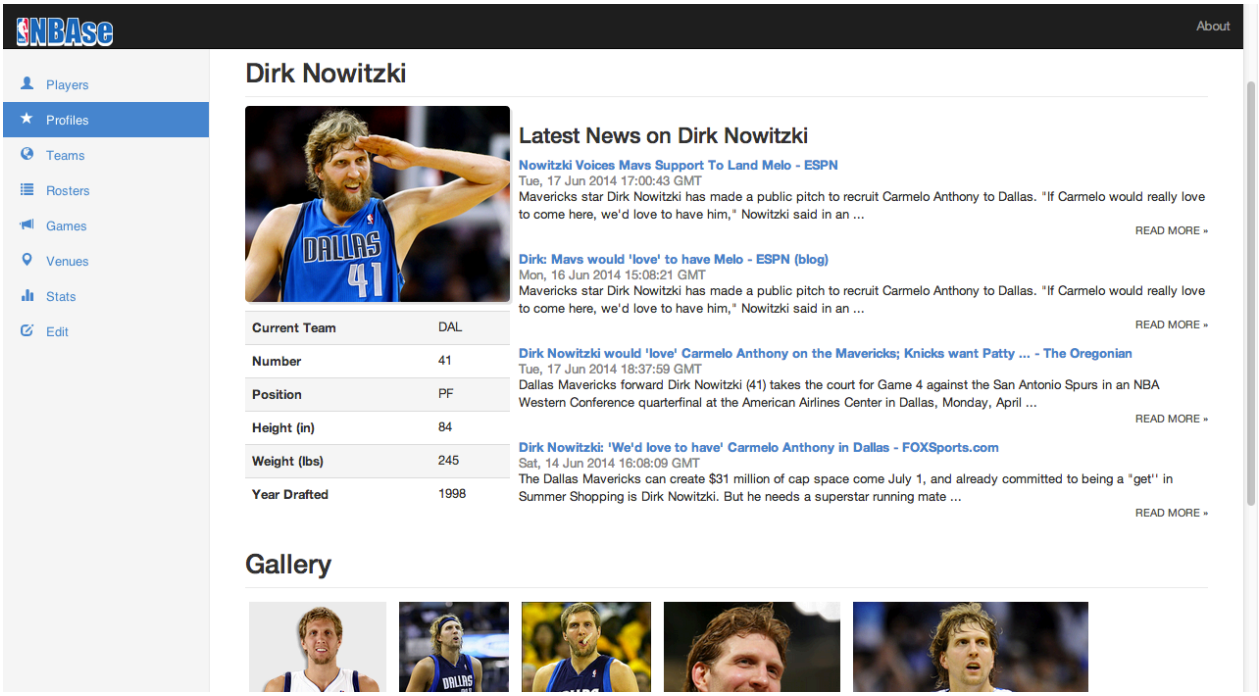

Figure 4.10: Google Images and Google News retrieval in profiles.php

```
SELECT city, abbreviation, teamName, divisionName
FROM NBATeam_BelongsTo
```

File location: `update.php`

```
SELECT firstName, lastName, number, team
FROM NBAPlayer_PlaysFor
ORDER BY firstName, lastName;

SELECT *
FROM NBATeam_BelongsTo;

SELECT *
FROM NBAPlayer_PlaysFor
WHERE number = 0 AND team = 'DAL';

SELECT *
FROM NBATeam_BelongsTo
WHERE abbreviation = 'DAL';
```

File location: `forms/rosters_select.php`

```
SELECT abbreviation, teamName, city
FROM NBATeam_BelongsTo
```

File location: `forms/stats_team.php`

```
SELECT DISTINCT company
FROM Sponsor_Endorses
```

Chapter 5

Functional Dependencies

All tables are in BCNF because all of their non-trivial functional dependencies have attributes on the LHS that are superkeys of their respective tables.

Division

No non-trivial functional dependencies were contained in the `Division` table.

Venue

1. venueName, city -> address
    - Meaning in English: The name of the venue along with the city determines the street address of that venue.

NBATeam_BelongsTo

1. abbreviation -> city, teamName, divisionName
    - Meaning in English: The 3-letter abbreviation of a team determines the city, team name, and division that the team is in.
2. teamName -> abbreviation, city, divisionName
    - Meaning in English: The team name determines the 3-letter abbreviation, city, and division that the team is in.

NBAPlayer_PlaysFor

1. number, team -> firstName, lastName, position, height, weight, draftYear
    - Meaning in English: The jersey number and the team of a player determines the full name, position, height, weight, and draft year of that player

NBAStaff_WorksFor

1. lastName, firstName, team -> job
    - Meaning in English: The full name of a staff member along with the team that they work for determines their job.

Sponsor_Endorses

No non-trivial functional dependencies were contained in the `Sponsor_Endorses` table.

NBAGame_Plays_PlayedAt

1. gameDate, homeTeam, awayTeam -> homeScore, awayScore, venueName, city
    - Meaning in English: The date of the NBA game, along with the home team and away team that played at that game determines the home score, away score, venue name, and city that the game was played at.
2. venueName, city, gameDate -> homeTeam, awayTeam, homeScore, awayScore
    - Meaning in English: The venue name, city, and game date determines the home team, away team, home score, and away score of that game.

NBAReferee

1. number -> firstName, lastName
    - Meaning in English: The jersey number of a referee determines his or her full name.

Referees

No non-trivial functional dependencies were contained in the `Referees` table.

Chapter 6

SQL Table Contents

The first 10 rows of each SQL table is included below for convenience.

Division

```
mysql> select * from division limit 10;
+--------------+
| divisionName |
+--------------+
| Atlantic     |
| Central      |
| Northwest    |
| Pacific      |
| Southeast    |
| Southwest    |
+--------------+
6 rows in set (0.00 sec)
```

Venue

```
mysql> select * from venue limit 10;
+---------------------------+----------------+-----------------------+
| venueName                 | city           | address               |
+---------------------------+----------------+-----------------------+
| Air Canada Centre         | Toronto        | 40 Bay St             |
| American Airlines Arena   | Miami          | 601 Biscayne Blvd     |
| American Airlines Center  | Dallas         | 2500 Victory Ave      |
| Amway Center              | Orlando        | 400 West Church St    |
| AT&T Center               | San Antonio    | 1 AT&T Center Pkwy    |
| Bankers Life Fieldhouse   | Indianapolis   | 125 S Pennsylvania St |
| Barclays Center           | Brooklyn       | 620 Atlantic Avenue   |
| BMO Harris Bradley Center | Milwaukee      | 1001 N 4th St         |
| Chesapeake Energy Arena   | Oklahoma City  | 100 W Reno Ave        |
| EnergySolutions Arena     | Salt Lake City | 301 W South Temple    |
+---------------------------+----------------+-----------------------+
10 rows in set (0.00 sec)
```

NBATeam_BelongsTo

```
mysql> select * from nbateam_belongsto limit 10;
+--------------+--------------+-----------+--------------+
| abbreviation | city         | teamName  | divisionName |
+--------------+--------------+-----------+--------------+
| ATL          | Atlanta      | Hawks     | Southeast    |
| BKN          | Brooklyn     | Nets      | Atlantic     |
| BOS          | Boston       | Celtics   | Atlantic     |
| CHA          | Charlotte    | Bobcats   | Southeast    |
| CHI          | Chicago      | Bulls     | Central      |
| CLE          | Cleveland    | Cavaliers | Central      |
| DAL          | Dallas       | Mavericks | Southwest    |
| DEN          | Denver       | Nuggets   | Northwest    |
| DET          | Detroit      | Pistons   | Central      |
| GSW          | Golden State | Warriors  | Pacific      |
+--------------+--------------+-----------+--------------+
10 rows in set (0.00 sec)
```

NBAPlayer_PlaysFor

```
mysql> select * from nbaplayer_playsfor limit 10;
+--------+----------+-----------+----------+--------+--------+-----------+------+
| number | position | firstName | lastName | height | weight | draftYear | team |
+--------+----------+-----------+----------+--------+--------+-----------+------+
|      0 | PG       | Jeff      | Teague   |     74 |    181 |      1995 | ATL  |
|      0 | C        | Andray    | Blatche  |     83 |    260 |      2010 | BKN  |
|      0 | PG       | Avery     | Bradley  |     74 |    180 |      1993 | BOS  |
|      0 | C        | Bismack   | Biyombo  |     81 |    245 |      1992 | CHA  |
|      0 | SG       | C.J.      | Miles    |     78 |    231 |      2012 | CLE  |
|      0 | SF       | Shawn     | Marion   |     79 |    228 |      2007 | DAL  |
|      0 | PG       | Aaron     | Brooks   |     84 |    161 |      2012 | DEN  |
|      0 | C        | Andre     | Drummond |     82 |    270 |      1998 | DET  |
|      0 | PF       | Glen      | Davis    |     81 |    289 |      1999 | LAC  |
|      0 | SF       | Nick      | Young    |     79 |    210 |      1992 | LAL  |
+--------+----------+-----------+----------+--------+--------+-----------+------+
10 rows in set (0.00 sec)
```

NBAStaff_WorksFor

```
mysql> select * from nbastaff_worksfor limit 10;
+-----------+------------+-----------------+------+
| firstName | lastName   | job             | team |
+-----------+------------+-----------------+------+
| Adrian    | Griffin    | Assistant Coach | CHI  |
| Alex      | McKechnie  | Assistant Coach | TOR  |
| Alvin     | Gentry     | Assistant Coach | LAC  |
| Andy      | Greer      | Assistant Coach | CHI  |
| Armond    | Hill       | Assistant Coach | LAC  |
| Bernie    | Bickerstaff | Assistant Coach | CLE  |
| Bill      | Bayno      | Assistant Coach | TOR  |
```

```
| Bob       | Bender      | Assistant Coach | MIL  |
| Bob       | Beyer       | Assistant Coach | CHA  |
| Bob       | Beyer       | Assistant Coach | DET  |
+-----------+-------------+-----------------+------+
10 rows in set (0.00 sec)
```

Sponsor_Endorses

```
mysql> select * from sponsor_endorses limit 10;
+-----------------------+------+
| company               | team |
+-----------------------+------+
| Hertz                 | ATL  |
| Love's Country's      | ATL  |
| Mercedes Benz         | ATL  |
| Northwest Ford Stores | ATL  |
| Sandridge Energy      | ATL  |
| State Farm Insurance  | ATL  |
| SWBC                  | ATL  |
| Texas Ford Dealers    | ATL  |
| Anheuser-Busch        | BKN  |
| Audi                  | BKN  |
+-----------------------+------+
10 rows in set (0.00 sec)
```

NBAGame_Plays_PlayedAt

```
mysql> select * from nbagame_plays_playedat limit 10;
+------------+-----------+-----------+----------+----------+---------------------------+--------------+
| gameDate   | homeScore | awayScore | homeTeam | awayTeam | venueName                 | city         |
+------------+-----------+-----------+----------+----------+---------------------------+--------------+
| 2013-10-29 |        97 |        87 | IND      | ORL      | Bankers Life Fieldhouse   | Indianapolis |
| 2013-10-29 |       116 |       103 | LAL      | LAC      | Staples Center            | Los Angeles  |
| 2013-10-29 |       107 |        95 | MIA      | CHI      | American Airlines Arena   | Miami        |
| 2013-10-30 |        98 |        94 | CLE      | BKN      | Quicken Loans Arena       | Cleveland    |
| 2013-10-30 |       118 |       109 | DAL      | ATL      | American Airlines Center  | Dallas       |
| 2013-10-30 |       113 |       102 | DET      | WAS      | The Palace of Auburn Hills | Detroit      |
| 2013-10-30 |       125 |        94 | GSW      | LAL      | Oracle Arena              | Oakland      |
| 2013-10-30 |        96 |        83 | HOU      | CHA      | Toyota Center             | Houston      |
| 2013-10-30 |       120 |       115 | MIN      | ORL      | Target Center             | Minneapolis  |
| 2013-10-30 |        90 |        95 | NOH      | IND      | Smoothie King Center      | New Orleans  |
+------------+-----------+-----------+----------+----------+---------------------------+--------------+
10 rows in set (0.00 sec)
```

NBAReferee

```
mysql> select * from nbareferee limit 10;
+--------+-----------+-----------+
| number | firstName | lastName  |
+--------+-----------+-----------+
|      6 | Tony      | Brown     |
|      7 | Derrick   | Stafford  |
|      8 | Marc      | Davis     |
|     10 | Ron       | Garretson |
|     11 | Derrick   | Collins   |
```

```
|     12 | Violet    | Palmer    |
|     13 | Monty     | McCutchen |
|     14 | Ed        | Malloy    |
|     15 | Bennett   | Salvatore |
|     16 | David     | Guthrie   |
+--------+-----------+-----------+
10 rows in set (0.00 sec)
```

Referees

```
mysql> select * from referees limit 10;
+-----------+------------+----------+----------+
| refNumber | gameDate   | homeTeam | awayTeam |
+-----------+------------+----------+----------+
|        45 | 2013-10-29 | IND      | ORL      |
|         8 | 2013-10-29 | LAL      | LAC      |
|        73 | 2013-10-29 | MIA      | CHI      |
|        68 | 2013-10-30 | CLE      | BKN      |
|        37 | 2013-10-30 | DAL      | ATL      |
|        30 | 2013-10-30 | DET      | WAS      |
|        20 | 2013-10-30 | GSW      | LAL      |
|        14 | 2013-10-30 | HOU      | CHA      |
|        44 | 2013-10-30 | MIN      | ORL      |
|         8 | 2013-10-30 | NOH      | IND      |
+-----------+------------+----------+----------+
10 rows in set (0.00 sec)
```

Chapter 7

Demo Script

Note: Times for each section are approximate.

Before the demo

- For Norman: Print 3 copies of `project-final-report` which should contain:
  - ER diagram
  - relation instances (table of data)
  - SQL for all queries used in demo
  - SQL for creating tables
- For Norman: Have windows on your laptop open with:
  - Amazon EC2 dashboard
  - Apache HTTP Server config file: `/opt/bitnami/apache2/conf/httpd.cnf`
  - Apache HTTP Server config file: `/opt/bitnami/apache2/conf/bitnami/bitnami.cnf`
  - MySQL config file: `/opt/bitnami/mysql/my.cnf`
  - Terminal to restart `apache` and/or `mysql` with commands:
    * sudo bash ctlscript.sh restart apache
    * sudo bash ctlscript.sh restart mysql

Setup (2 minutes)

- Run scripts to create and populate database
- Show TA database instances before and after

Selection and Projection Query (2 minutes)

- Navigate to **Players** page

1. From the **Filter Results** menu, specify a selection constraint (any is fine)
2. Repeat the previous step with a different selection constraint
   - Demonstrate type checking on user input (i.e. input must be a number >= 0)
3. *Optional*: Order the results by some attribute in the **Order Results** menu
4. Choose some attributes to display in the **Display Attributes** menu

Join Query (2 minutes)

- Navigate to **Games** page

1. Show TA that data has been joined from 4 tables: `nbagame_plays_playedat`, `nbareferee`, and `referees`
2. If necessary to show a second example, navigate to **Rosters** page and show the list of sponsors for any team
    - This query joins data from 2 tables: `nbateam_belongsto` and `sponsor_endorses`

Division Query (2 minutes)

- Navigate to **Venues** page

1. Show TA *Teams that have played at all venues* section (and explain how this is a divison query if necessary)
2. Add a new venue under the **Add a Venue** menu
3. Return to **Venues** page, show TA that there are no longer any teams under the *Teams that have played at all venues* section

Aggregation Query (2 minutes)

- Navigate to **Stats** page

1. From the **League-Wide** menu, choose an aggregate operator and an attribute
    - Show that this result is consistent with our relation instance
2. Repeat the previous step with a different operator and attribute
    - Show that this result is consistent with our relation instance

Nested Aggregation with Group By (2 minutes)

- While still on the **Stats** page

1. Show TA nested aggregation query on printout
2. From the **Team-Wide** menu, choose an aggregate operator, an attribute, and a sponsor
    - Show that the results are consistent with our relation instances
3. Repeat the previous step with different parameters
    - Show that the results are consistent with our relation instances

Delete Operation (4 minutes)

- Navigate to **Teams** page

1. Explain the constraints for our tables
    - Deleting a team should delete all players and games that reference that team (`ON DELETE CASCADE`)
    - Cannot delete a game that is referenced by the referees table (`ON DELETE NO ACTION`)
    - Deleting a team should fail if a game has at least one referee
2. Delete a team that participates in a game with a referee
    - Show that this deletion fails (e.g. show that the team still exists and/or any players & games associated with that team still exist)
    - Show that error message is located in popup box
3. Delete a team that only participates in games without referees
    - Show that this deletion succeeds (e.g. show that the team is gone from **Teams** page and that there are no players on the **Players** page with that team)

- Navigate to **Players** page

1. Delete a player from the table (deletion without cascade)

Update Operation (2 minutes)

- Navigate to **Edit** page

1. Choose any player to edit
2. Attempt to change that player's draft year to a year that is greater than the current year
    - Show TA that this update is rejected since it violates a constraint
    - Show that error message is located in popup box
3. Reattempt to change that player's draft year to a valid year
    - Show that this update succeeded (e.g. navigate to **Players** page and show the player with new draft year)
4. Show additional type checking for the update form
    - First and Last names must have only alphabet characters and have length >= 1
    - Position is selected from dropdown menu
    - Number must be numeric and have value: 0 <= value <= 99
    - Note: Could also attempt to update player number to a number that already exists on that team (violates primary key constraint)
    - Height must be numeric with value >= 0
    - Weight must be numeric with value >= 0
    - Draft year must be numeric with 1946 <= value <= currentYear
    - Team is selected from dropdown

Extra Features (2 minutes)

- Navigate to **Players page**

1. Click on any player from list of players
2. Show that the page displays unique information from the database about that player
3. Explain that the profile picture is dynamically retrieved from Google images
4. Explain that the latest news for each player is dynamically retrieved from Google news
5. Show the profile page of another (one or more) players