Group 3 - Project Milestone 2
Database Systems CPSC 5021
Alex Peterson, Chris Downing & Ross Hoyt
Fall Quarter 2018

MySQL *Amateur Basketball League*: Milestone 2

Part 1) Database Queries

-- 1) This query obtains the names of seasons with scheduled matches and the date and times of the first and last match of the season.

SQL Statement:

```
SELECT
basketballleagueproject.GAME.season AS SeasonID,
basketballleagueproject.SEASON.seasonName AS SeasonName,
MIN(basketballleagueproject.GAME.matchdateTimeStart) AS FirstMatch,
MAX(basketballleagueproject.GAME.matchdateTimeStart) AS LastMatch
FROM
basketballleagueproject.GAME
INNER JOIN
basketballleagueproject.SEASON ON basketballleagueproject.GAME.season =
basketballleagueproject.SEASON.idSEASON
GROUP BY
basketballleagueproject.GAME.season;
```

Output:

SeasonID	SeasonName	FirstMatch	LastMatch
1	Late-Fall Open Soft-core	2018-11-21 19:00:00	2018-12-19 19:00:00
4	Early Winter Open Soft-core	2019-01-02 19:00:00	2019-01-30 19:00:00

-- 2) Show roster for specific team (team 1 in this example). Displays all members of team 1.

SQL Statement:

```
SELECT sportsleagueproject.PLAYER.idPLAYER,
sportsleagueproject.PLAYER.firstName,
sportsleagueproject.PLAYER.lastName,
sportsleagueproject.TEAM.idTEAM, sportsleagueproject.TEAM.teamName
FROM ((sportsleagueproject.PLAYER_TEAM
INNER JOIN sportsleagueproject.PLAYER ON
sportsleagueproject.PLAYER_TEAM.player =
sportsleagueproject.PLAYER.idPLAYER)
INNER JOIN sportsleagueproject.TEAM ON
sportsleagueproject.PLAYER_TEAM.team = sportsleagueproject.TEAM.idTEAM)
WHERE sportsleagueproject.TEAM.idTEAM = 1
ORDER BY sportsleagueproject.PLAYER.idPLAYER;
```

Output:

idPLAYER	firstName	lastName	idTEAM	teamName
1	Javier	Rodriguez	1	Daggers
8	Fred	Kuczmarski	1	Daggers
16	Steve	Jobs	1	Daggers
17	Ryan	Gosling	1	Daggers
26	Melissa	Nealy	1	Daggers
36	Margaret	Thatcher	1	Daggers

-- 3) Team schedule for given team (team 1 in this example). Displays all future scheduled games for team 1.

SQL Statement:

```
SELECT T1.seasonName, T1.matchdateTimeStart, T1.facilityName,
T1.matchNumber, T2.OpposingTeam FROM
(SELECT basketballleagueproject.SEASON.seasonName,
basketballleagueproject.GAME.matchdateTimeStart,
basketballleagueproject.FACILITY.facilityName,
basketballleagueproject.GAME_TEAM.match AS matchNumber
FROM  ((((basketballleagueproject.GAME_TEAM
INNER JOIN basketballleagueproject.TEAM ON
basketballleagueproject.GAME_TEAM.team =
basketballleagueproject.TEAM.idTEAM)
INNER JOIN basketballleagueproject.GAME ON
basketballleagueproject.GAME_TEAM.match =
basketballleagueproject.GAME.idGAME)
INNER JOIN basketballleagueproject.FACILITY ON
basketballleagueproject.GAME.facility =
basketballleagueproject.FACILITY.idFACILITY)
INNER JOIN basketballleagueproject.SEASON ON
basketballleagueproject.GAME.season =
basketballleagueproject.SEASON.idSEASON)
WHERE basketballleagueproject.TEAM.idTEAM = 1 &&
basketballleagueproject.GAME.matchdateTimeStart >= NOW()) AS T1
INNER JOIN
(SELECT basketballleagueproject.GAME_TEAM.match AS matchNumber2,
basketballleagueproject.TEAM.teamName AS OpposingTeam FROM
(((SELECT basketballleagueproject.GAME_TEAM.match AS matchNumber FROM
basketballleagueproject.GAME_TEAM WHERE
basketballleagueproject.GAME_TEAM.team = 1) AS T1
INNER JOIN basketballleagueproject.GAME_TEAM ON T1.matchNumber =
basketballleagueproject.GAME_TEAM.match)
INNER JOIN basketballleagueproject.TEAM ON
basketballleagueproject.GAME_TEAM.team =
basketballleagueproject.TEAM.idTEAM)
WHERE basketballleagueproject.GAME_TEAM.team != 1 ) AS T2
```

```
ON T1.matchNumber = T2.matchNumber2;
```

Output:

seasonName	matchdateTime Start	facility Name	matchNumber	OpposingTeam
Late-Fall Open Soft-core	2018-11-21 19:00:00	Seattle Sporting Center	1	Community Team2 Late Fall Open Soft-core
Late-Fall Open Soft-core	2018-11-28 19:00:00	Seattle Sporting Center	4	Community Team1 Late Fall Open Soft-core
<<...>>	<<...>>	<<...>>	<<...>>	<<...>>
Early Winter Open Soft-core	2019-01-23 19:00:00	Seattle Sporting Center	25	BasketWeavers
Early Winter Open Soft-core	2019-01-30 19:00:00	Seattle Sporting Center	28	Hard Core Ballers

-- 4) Player's schedule for all upcoming games (playerid 16 in this example).

SQL Statement:

```
SELECT T2.seasonName, T2.matchDateTimeStart as MatchStartTime,
T2.facilityName AS Facility,
T1.teamName as PlayingAs FROM
(SELECT basketballleagueproject.SEASON.seasonName,
basketballleagueproject.TEAM.idTEAM,
basketballleagueproject.GAME.matchdateTimeStart,
basketballleagueproject.FACILITY.facilityName,
```

```
basketballleagueproject.GAME_TEAM.match AS matchNumber
FROM
((((basketballleagueproject.GAME_TEAM
INNER JOIN basketballleagueproject.TEAM ON
basketballleagueproject.GAME_TEAM.team =
basketballleagueproject.TEAM.idTEAM)
INNER JOIN basketballleagueproject.GAME ON
basketballleagueproject.GAME_TEAM.match =
basketballleagueproject.GAME.idGAME)
INNER JOIN basketballleagueproject.FACILITY ON
basketballleagueproject.GAME.facility =
basketballleagueproject.FACILITY.idFACILITY)
INNER JOIN basketballleagueproject.SEASON ON
basketballleagueproject.GAME.season =
basketballleagueproject.SEASON.idSEASON)
WHERE basketballleagueproject.GAME.matchdateTimeStart >= Now()) AS T2
INNER JOIN
(SELECT basketballleagueproject.TEAM.teamName,
basketballleagueproject.TEAM.idTEAM FROM
((basketballleagueproject.PLAYER_TEAM
INNER JOIN basketballleagueproject.PLAYER ON
basketballleagueproject.PLAYER_TEAM.player =
basketballleagueproject.PLAYER.idPLAYER)
INNER JOIN basketballleagueproject.TEAM ON
basketballleagueproject.PLAYER_TEAM.team =
basketballleagueproject.TEAM.idTEAM)
WHERE basketballleagueproject.PLAYER.idPLAYER = 40) AS T1
ON T2.idTEAM = T1.idTEAM;
```

Output:

seasonName	MatchStartTime	Facility	PlayingAs
Late-Fall Open Soft-core	2018-11-21 19:00:00	Seattle Sporting Center	Community Team2 Late Fall Open Soft-core

Late-Fall Open Soft-core	2018-11-28 19:00:00	Seattle Sporting Center	Community Team2 Late Fall Open Soft-core
<<...>>	<<...>>	<<...>>	<<...>>
Early Winter Open Soft-core	2019-01-23 19:00:00	Seattle Sporting Center	Masters of the Court
Early Winter Open Soft-core	2019-01-30 19:00:00	Seattle Sporting Center	Masters of the Court

-- 5) Show schedule of all games (date/time and location) for a given season (season id 1 in this example).

SQL Statement:

```
SELECT basketballleagueproject.GAME.idGAME,
basketballleagueproject.SEASON.seasonName,
basketballleagueproject.FACILITY.facilityName,
basketballleagueproject.GAME.matchDateTimeStart AS DateTimeStart
FROM basketballleagueproject.GAME
INNER JOIN basketballleagueproject.SEASON
ON basketballleagueproject.GAME.season =
basketballleagueproject.SEASON.idSEASON
INNER JOIN basketballleagueproject.FACILITY
ON basketballleagueproject.GAME.facility =
basketballleagueproject.FACILITY.idFACILITY
WHERE basketballleagueproject.GAME.season = 1;
```

Output:

idGAME	seasonName	facilityName	DateTimeStart
1	Late-Fall Open Soft-core	Seattle Sporting Center	2018-11-21 19:00:00
2	Late-Fall Open Soft-core	Seattle Sporting Center	2018-11-21 19:00:00
<<...>>	<<...>>	<<...>>	<<...>>
14	Late-Fall Open Soft-core	Seattle Sporting Center	2018-12-19 19:00:00
15	Late-Fall Open Soft-core	Seattle Sporting Center	2018-12-19 19:00:00

-- 6) Retrieves list of all female players.

SQL Statement:

```
SELECT * FROM basketballleagueproject.PLAYER
WHERE basketballleagueproject.PLAYER.sex = 'F';
```

Output (not all columns or rows shown):

idPLAYER	firstName	lastName	dateOfBirth	sex	email
26	Melissa	Nealy	1994-03-28	F	mellisaN@yahoo.com
27	Linda	Anderson	1969-05-10	F	lindahanderson@gmail.com
<<...>>	<<...>>	<<...>>	<<...>>	<.>	<<...>>

39	Rebecca	Herivel	1985-12-13	F	rebeccah@gmail.com
40	Christine	James	1981-07-01	F	ChristineJ@yahoo.com

-- 7) Retrieves all male players in league.

SQL Statement:

```
SELECT * FROM basketballleagueproject.PLAYER
WHERE basketballleagueproject.PLAYER.sex = 'M';
```

Output (not all columns or rows shown):

idPLAYER	firstName	lastName	dateOfBirth	sex	email
1	Javier	Rodriguez	1985-06-22	M	javrod@gmail.com
2	David	George	1990-03-12	M	davidg@hotmail.com
<<...>>	<<...>>	<<...>>	<<...>>	<.>	<<<...>>
24	Lebron	James	1985-02-28	M	lebronjameyjames@gmail.com
25	Eric	Cartman	1998-05-12	M	ericc@gmail.com

-- 8) Player stats for individual game (assumption is that particular game is known)

SQL Statement:

```
SELECT * FROM basketballleagueproject.PLAYER_PERFORMANCE WHERE
basketballleagueproject.PLAYER_PERFORMANCE.player = 16
&& basketballleagueproject.PLAYER_PERFORMANCE.team_match = 13;
```

Output (not all columns shown)

idPLAYER_PERFORMANCE	player	team_match	playedInGame	pts	FGM	FGA
3	16	13	Y	28	11	18

-- 9) Player stats for all time.

SQL Statement:

```
SELECT SUM(basketballleagueproject.PLAYER_PERFORMANCE.FGM) AS 'FGM:
All-Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.FGA) AS 'FGA: All-Time',
(SUM(basketballleagueproject.PLAYER_PERFORMANCE.FGM) /
SUM(basketballleagueproject.PLAYER_PERFORMANCE.FGA) * 100.0) AS 'Field
Goal %: All-Time',
-- SUM(sportsleagueproject.PLAYER_PERFORMANCE.3PM) AS '3PM: All-Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.FTM) AS 'FTM: All-Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.FTA) AS 'FTA: All-Time',
(100.0 * SUM(basketballleagueproject.PLAYER_PERFORMANCE.FTM) /
SUM(basketballleagueproject.PLAYER_PERFORMANCE.FTA)) AS 'Free Throw %:
All Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.OFF_REB) AS 'OFF_REB: All
Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.DEF_REB) AS 'DEF_REB: All
Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.AST) AS 'Assists: All
Time',
SUM(basketballleagueproject.PLAYER_PERFORMANCE.TOV) AS 'TOV: All Time',
```

```
SUM(basketballleagueproject.PLAYER_PERFORMANCE.STL) AS 'Steals: All
Time'
FROM basketballleagueproject.PLAYER_PERFORMANCE
WHERE basketballleagueproject.PLAYER_PERFORMANCE.player = 16;
```

Output (not all columns shown):

FGM: All-Time	FGA: All-Time	Field Goal %: All-Time	FTM: All-Time	FTA: All-Time	Free Throw %: All Time
51	70	72.85714	20	31	64.51613

-- 10) Retrieves each score that every team got in every game played.

SQL Statement:

```
SELECT basketballleagueproject.GAME_TEAM.score,
basketballleagueproject.GAME_TEAM.match,
basketballleagueproject.TEAM.teamName
FROM basketballleagueproject.GAME_TEAM
INNER JOIN basketballleagueproject.TEAM ON
basketballleagueproject.GAME_TEAM.team =
basketballleagueproject.TEAM.idTEAM
WHERE basketballleagueproject.GAME_TEAM.score IS NOT NULL;
```

Output:

score	match	teamName
110	1	Daggers
78	4	Daggers
75	7	Daggers

<<...>>	<<...>>	<<<..>>>
90	12	Community Team2 Late Fall Open Soft-core
97	14	Community Team2 Late Fall Open Soft-core
115	16	Masters of the Court

-- 11) All names and email addresses of players for league-wide communications.

SQL Statement:

```
SELECT basketballleagueproject.PLAYER.firstName,
basketballleagueproject.PLAYER.lastName,
basketballleagueproject.PLAYER.email FROM
basketballleagueproject.PLAYER;
```

Output:

firstName	**lastName**	**email**
Javier	Rodriguez	javrod@gmail.com
David	George	davidg@hotmail.com
<<...>>	<<...>>	<<...>>
Rebecca	Herivel	rebeccah@gmail.com
Christine	James	ChristineJ@yahoo.com

Part 2) Stored Procedure

Stored procedure setGameScores():

```
CREATE DEFINER=`root`@`localhost` PROCEDURE `setGameScores`(game INT,
team INT, score INT)
BEGIN
    UPDATE basketballleagueproject.GAME_TEAM SET
basketballleagueproject.GAME_TEAM.score = score
    WHERE basketballleagueproject.GAME_TEAM.match = game
    && basketballleagueproject.GAME_TEAM.team = team;
END
```

Two entries from GAME_TEAM before call of stored procedure. Score column is null because match has not occured yet:

match	team	score
16	1	NULL
16	8	NULL

Calling stored procedure to update scores for teams 1 and 8 after match 16 is played:

```
call basketballleagueproject.setGameScores(16, 1, 107);
call basketballleagueproject.setGameScores(16, 8, 115);
```

Entries from GAME_TEAM After call of stored procedure showing updated scores:

match	team	score
16	1	107

16	8	115

Part 3) Business Rules

-Player Rules

- A player must be 18 years of age or older to join the league
- A male player may play in seasons designated men's, co-ed, or open
- A female player may plan in seasons designated women's, co-ed, or open teams
- A player may be on multiple teams in the league, but may only play on one team for any given season

-Team Rules

- A team may have any arbitrary number of players (however, a team may or may not be eligible to play in certain seasons depending on the team size requirements)
- A team designated "Community TeamX [Name of season]" (where X is the team number) is a team open for sign-up of individual players from the community. Community teams exist for one season only and are then disbanded (if desired, the players on the community team may create their own team for future participation in the league)
- Teams may participate in multiple seasons throughout time, though they may only participate in one season at any time

Game Scheduling Rules

- A team may not be scheduled for more than one game per day
- Unless otherwise stated, scheduling will be done using the round robin system, cycling teams as many times as possible for the duration of the season
- The number of teams playing at one time in one facility may not exceed the number of basketball courts the facility has available
- Each game will be referred by at least one staff referee
- A referee may only ref one game at a time

Season Administration Rules

- The number of teams may not be under the minimum nor over the maximum number of teams specified upon season declaration
- The number of players per team may not be under the minimum nor over the maximum number of players allowed per team for this season

- Seasons are defined by a ‘type’ which is a description of who can play (men’s, women’s, co-ed, or open) and the skill level (soft-core, mid-core, advanced, etc).
- Seasons of the ‘Men’s’ type may only have teams that consist only of men
- Seasons of the ‘Women’s’ type may only have teams that consist only of women
- Seasons of the ‘Coed’ type must have enough men and women to achieve 50-50 gender parity rounded down to the nearest integer (so in a game with five player teams, two women and 3 men is acceptable, but one women and four men is unacceptable)
- Seasons of the ‘Open’ type have no restriction on the number of men and women allowed on the teams that participate

Part 4) ER Diagram