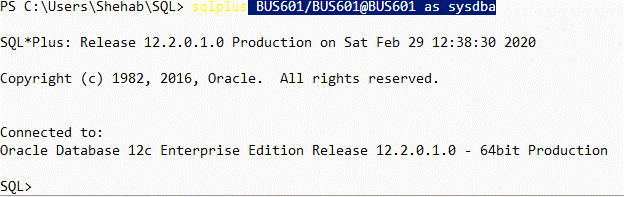
**BUS 601 Project –** Oracle Database for Rec League

**Shehab Hasan** - CSUID 2632766

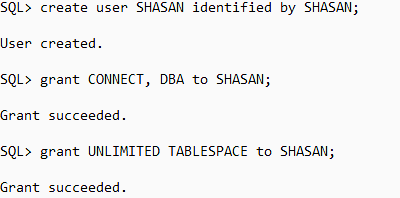
**Step One – Installation:**

Oracle 12c was installed using windows installation package. Using the packaged database configuration tool, I created a database named “BUS601” with an administrator username and password of “BUS601”.

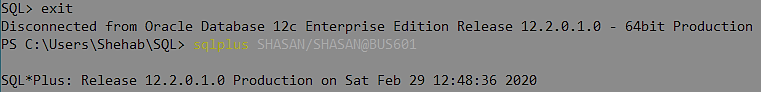
Here is a screenshot showing a connection to that database with username “BUS601”.



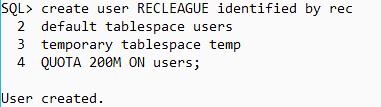
Next, I want to create my own USER ID with my first initial and last name. This user will be a dba user with unlimited tablespace. Here is a screenshot of the sql commands and results:



I will now exit sql and login using this new user.



I now want to create a user with the name of the application – RECLEAGUE. This will not be a dba user, and have limits placed on tablespace.



**Step Two – Design:**

A municipal recreation center wants to encourage physical exercise and community interaction by organizing an ongoing league where participants can form teams and compete against other teams across different games and events, such as basketball matches, track and field competitions, swimming races, etc. They need a database to record information about the teams, match results, and different contests organized by the recreation center.

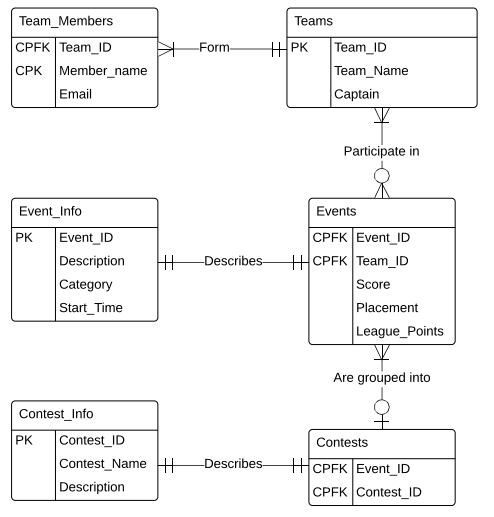
All participants must be part of a team, and every team must have at least one member.

Events must have at least one participating team but have no upper limit. While some events, like a basketball game, may only have 2 teams, others, like a race, can have any number of teams.

In addition to recording the score and placement of each team, the rec center wants to award league points to each team based on their results in the event. These league points are used to determine the overall winner of contests and team’s overall placement on the leaderboard.

The rec center will also organize specific contests, such as a basketball tournament, that are comprised of multiple associated events. Not all rec league matches are part of contests, but all contests must have at least one event.

**RECLEAGUE ENTITY-RELATIONSIHP DIAGRAM:**



**Step 3 – Coding the Application:**

Table creation and sequence statements were prepared into .sql script files. These include primary key, composite primary key and foreign key constraints. Here are the contents of the .sql files:

**createteams.sql**

CREATE TABLE RECLEAGUE.TEAMS(

TEAM\_ID NUMBER(6) PRIMARY KEY,

TEAM\_NAME VARCHAR2(20) NOT NULL,

CAPTAIN VARCHAR2(10)

);

CREATE TABLE RECLEAGUE.TEAM\_MEMBERS(

TEAM\_ID NUMBER(6) NOT NULL,

MEMBER\_NAME VARCHAR2(16) NOT NULL,

EMAIL VARCHAR2(25) NOT NULL,

CONSTRAINT FK\_TEAM

FOREIGN KEY (TEAM\_ID)

REFERENCES RECLEAGUE.TEAMS(TEAM\_ID),

CONSTRAINT PK\_TEAM

PRIMARY KEY (TEAM\_ID, MEMBER\_NAME)

);

CREATE SEQUENCE team\_seq

START WITH 1

INCREMENT BY 1

MINVALUE 1

MAXVALUE 999999

CYCLE;

**createevents.sql**

CREATE TABLE RECLEAGUE.EVENT\_INFO(

EVENT\_ID NUMBER(9) PRIMARY KEY,

DESCRIPTION VARCHAR2(1000) NOT NULL,

CATEGORY VARCHAR2(20) NOT NULL,

START\_TIME DATE NOT NULL

);

CREATE TABLE RECLEAGUE.EVENTS(

EVENT\_ID NUMBER(9) NOT NULL,

TEAM\_ID NUMBER(6) NOT NULL,

SCORE VARCHAR2(15),

PLACEMENT NUMBER(4),

LEAGUE\_POINTS NUMBER(3),

CONSTRAINT EVENT\_TEAM\_FK

FOREIGN KEY (TEAM\_ID)

REFERENCES RECLEAGUE.TEAMS(TEAM\_ID),

CONSTRAINT EVENT\_INFO\_FK

FOREIGN KEY (EVENT\_ID)

REFERENCES RECLEAGUE.EVENT\_INFO(EVENT\_ID),

CONSTRAINT PK\_EVENTS

PRIMARY KEY (TEAM\_ID, EVENT\_ID)

);

CREATE SEQUENCE event\_seq

START WITH 1

INCREMENT BY 1

MINVALUE 1

MAXVALUE 999999999

CYCLE;

**creatcontests.sql**

CREATE TABLE RECLEAGUE.CONTEST\_INFO(

CONTEST\_ID NUMBER(6) PRIMARY KEY,

CONTEST\_NAME VARCHAR2(40) NOT NULL,

DESCRIPTION VARCHAR2(1000)

);

CREATE TABLE RECLEAGUE.CONTESTS(

EVENT\_ID NUMBER(9) NOT NULL,

CONTEST\_ID NUMBER(6) NOT NULL,

CONSTRAINT CONTESTS\_INFO\_FK

FOREIGN KEY (CONTEST\_ID)

REFERENCES RECLEAGUE.CONTEST\_INFO(CONTEST\_ID),

CONSTRAINT CONTESTS\_EVENTS\_FK

FOREIGN KEY (EVENT\_ID)

REFERENCES RECLEAGUE.EVENT\_INFO(EVENT\_ID),

CONSTRAINT PK\_CONTESTS

PRIMARY KEY (CONTEST\_ID, EVENT\_ID)

);

CREATE SEQUENCE contest\_seq

START WITH 1

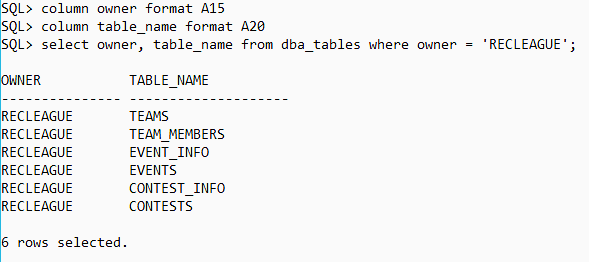
INCREMENT BY 1

MINVALUE 1

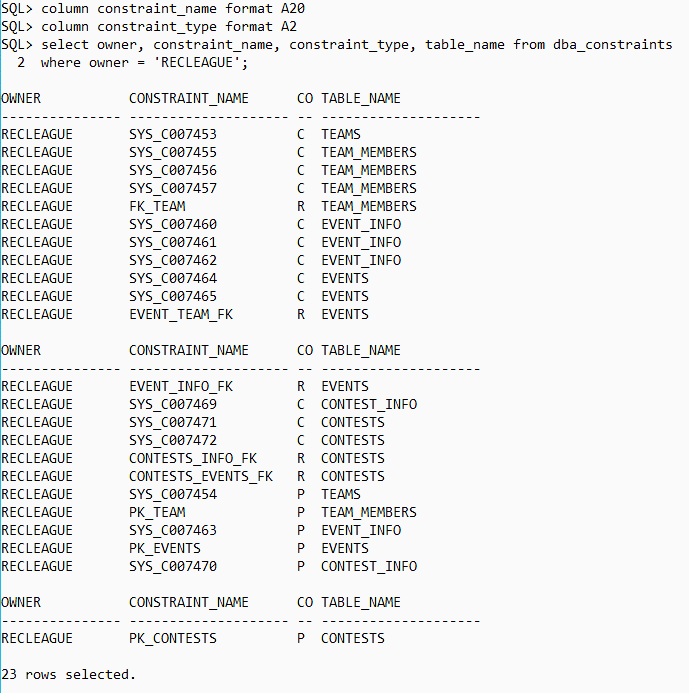
MAXVALUE 999999

CYCLE;

After all these scripts where run, I query dba\_tables to verify the tables are all there:



Next, I check query dba\_constraints to verify the creation of the primary, composite and foreign keys:



The tables are now ready to handle insertion of data.

**Step Three, part 2 – Inserting data:**

Again, I use script files to make the insertion of data easier. Here are the contents:

**Insertteams.sql**

INSERT INTO RECLEAGUE.TEAMS(Team\_ID, Team\_Name, Captain)

VALUES(team\_seq.NEXTVAL, 'Da GOATS', 'LeBron');

INSERT INTO RECLEAGUE.TEAMS(Team\_ID, Team\_Name, Captain)

VALUES(team\_seq.NEXTVAL, 'Nintendudes', 'Mario');

INSERT INTO RECLEAGUE.TEAMS(Team\_ID, Team\_Name, Captain)

VALUES(team\_seq.NEXTVAL, 'Foodies', 'Burger');

commit;

**inserteammembers.sql**

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (1,'LeBron','goat@cavs.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (1,'Love', 'love@cavs.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (1,'Kyrie','flat@cavs.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (1,'JR Smith','pipe@cavs.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (2,'Mario','mario@nintendo.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (2,'Luigi','sadbro@nintendo.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (2,'Link','hya@nintendo.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (2,'Yoshi','dino@nintendo.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (2,'Zelda','princess@nintendo.com');

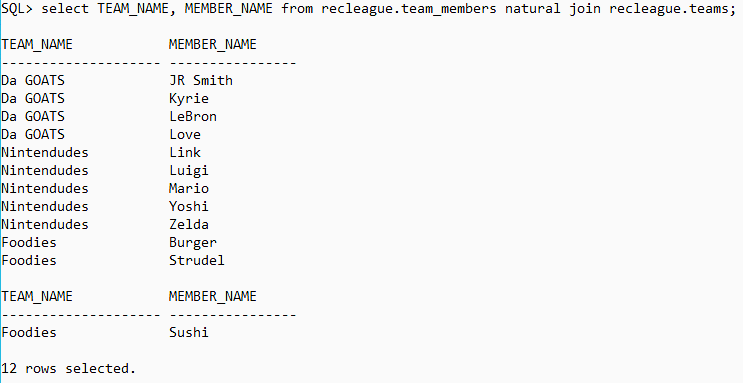
INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (3,'Burger','burg@foodies.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (3,'Strudel','strude@foodies.com');

INSERT INTO RECLEAGUE.TEAM\_MEMBERS(Team\_ID, Member\_name, email) VALUES (3,'Sushi','sushi@foodies.com');

commit;

To verify insertions I do a select query which lists each team member along their team name:



**Insertcontestinfo.sql**

INSERT INTO RECLEAGUE.CONTEST\_INFO(Contest\_ID, Contest\_Name, Description)

VALUES(contest\_seq.NEXTVAL, 'Basketball Day', '3v3 basketball games every half hour');

INSERT INTO RECLEAGUE.CONTEST\_INFO(Contest\_ID, Contest\_Name, Description)

VALUES(contest\_seq.NEXTVAL, 'Track and Field Day', 'Various track and field events');

commit;

**inserteventinfo.sql**

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, '3v3 basketball game with two 10-minute halves', 'Basketball',

TO\_DATE('2020/01/04 12:00:00', 'yyyy/mm/dd hh24:mi:ss'));

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, '3v3 basketball game with two 10-minute halves', 'Basketball',

TO\_DATE('2020/01/04 12:30:00', 'yyyy/mm/dd hh24:mi:ss'));

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, '3v3 basketball game with two 10-minute halves', 'Basketball',

TO\_DATE('2020/01/04 13:00:00', 'yyyy/mm/dd hh24:mi:ss'));

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, '100m dash', 'Track and Field',

TO\_DATE('2020/01/05 12:00:00', 'yyyy/mm/dd hh24:mi:ss'));

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, 'Long Jump', 'Track and Field',

TO\_DATE('2020/01/05 12:30:00', 'yyyy/mm/dd hh24:mi:ss'));

INSERT INTO RECLEAGUE.EVENT\_INFO(Event\_ID, Description, Category, Start\_Time)

VALUES(event\_seq.NEXTVAL, '50m hurdles', 'Track and Field',

TO\_DATE('2020/01/05 13:00:00', 'yyyy/mm/dd hh24:mi:ss'));

commit;

**insertcontests.sql**

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (1, 1);

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (1, 2);

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (1, 3);

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (2, 4);

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (2, 5);

INSERT INTO RECLEAGUE.CONTESTS(CONTEST\_ID, EVENT\_ID) VALUES (2, 6);

commit;

**insertevents.sql**

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(1, 1, '40 pts', 1, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(1, 2, '4 pts', 2, 1);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(2, 1, '50 pts', 1, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(2, 3, '2 pts', 2, 1);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(3, 2, '19 pts', 1, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(3, 3, '14 pts', 2, 1);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(4, 1, '11.5 s', 1, 5);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(4, 2, '14 s', 1, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(4, 3, '15.5 s', 1, 1);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(5, 1, '8.31 m', 2, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(5, 2, '16.31 m', 1, 5);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(5, 3, '4.31 m', 3, 1);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(6, 1, '8.31 s', 2, 3);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(6, 2, '6.45 s', 1, 5);

INSERT INTO RECLEAGUE.EVENTS(Event\_ID, Team\_ID, Score, Placement, League\_Points)

VALUES(6, 3, '12.21 s', 3, 1);

commit;

Note: each batch of insertions was confirmed using a Select \* from RECLEAGUE.<tablename> statement. For the sake of brevity, I have not included the outputs of all these. Outputs from the advanced sql will show the data. However, I will describe the data in terms of the ‘business intelligence’ they represent.

There are two contests – a day of basketball matches and a track and field day.

There are three teams total participating in the contests.

On the basketball day, each team faces each other team once, getting 3 pts for a win and 1 for a loss.

On the track and field day, each team participates in every event, getting 5 pts for 1st, 3 for 2nd and 1 for 3rd place.

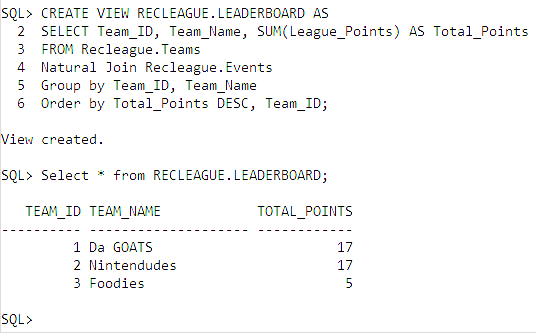
The “Da GOATS” team is extremely good at basketball and pretty good at track and field.

The “Nintendudes” team is good at track and field.

The “foodies” team is just there to participate and have a good time.

**Step Four – SQL queries:**

The rec center wants a leaderboard which tracks each team’s total points. I create a view for this purpose called leaderboard:



The rec center would also like to know the results of each contest by the total points earned by each team. Another view is created for this purpose:

**Contestresults.sql**

CREATE VIEW RECLEAGUE.CONTEST\_RESULTS AS

SELECT CI.CONTEST\_ID, CI.CONTEST\_NAME, TE.TEAM\_NAME, TE.TEAM\_ID, SUM(EV.LEAGUE\_POINTS) AS CONTEST\_TOTAL

FROM RECLEAGUE.EVENTS EV

INNER JOIN RECLEAGUE.CONTESTS CO

ON EV.EVENT\_ID = CO.EVENT\_ID

INNER JOIN RECLEAGUE.TEAMS TE

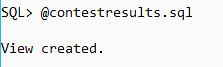
ON EV.TEAM\_ID = TE.TEAM\_ID

INNER JOIN RECLEAGUE.CONTEST\_INFO CI

ON CO.CONTEST\_ID = CI.CONTEST\_ID

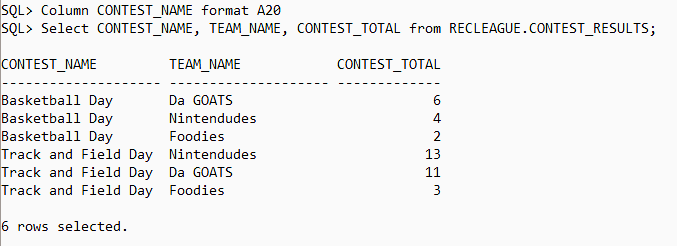
GROUP BY TE.TEAM\_ID, TE.TEAM\_NAME, CI.CONTEST\_ID, CI.CONTEST\_NAME

ORDER BY CI.CONTEST\_ID, CONTEST\_TOTAL DESC;

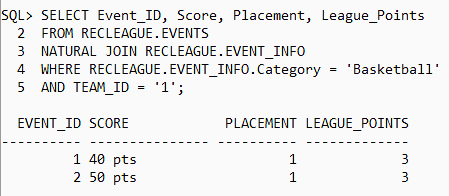


**Step Five – Data Reports:**

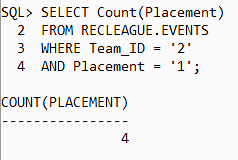
The Rec Center wants an easy to read list of the total points earned by each team. This can be done by querying the CONTEST\_RESULTS view we created in step four.



The ‘Da GOATS’ team wants to know how much they scored in all their basketball games.



The ‘Nintendudes’ team wants to know how many times they’ve gotten 1st place in an event



The Rec Center wants to have a list of the contests with just the winner of each contest and the number of league points they scored in that contest. This can be achieved by using the MAX function over the CONTEST\_RESULTS view created earlier.

**Contestwinners.sql**

CREATE VIEW RECLEAGUE.CONTEST\_WINNERS AS

SELECT CONTEST\_NAME, CONTEST\_ID, TEAM\_NAME, TEAM\_ID, CONTEST\_TOTAL

FROM(

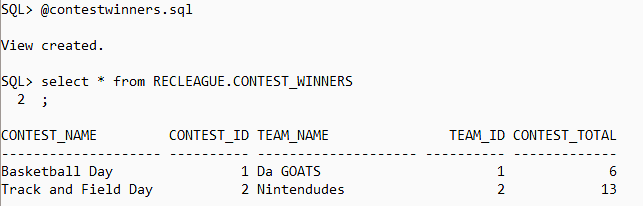
SELECT CONTEST\_NAME, CONTEST\_ID, TEAM\_NAME, TEAM\_ID, CONTEST\_TOTAL,

MAX(CONTEST\_TOTAL) OVER (PARTITION BY CONTEST\_ID) AS POINTS

FROM RECLEAGUE.CONTEST\_RESULTS

)

WHERE CONTEST\_TOTAL = POINTS;



This view can be further queried to get the total number of contests won by each team has at least 1 win.

