**CASE SETUP:**

Football is one of the most popular games played throughout the world. The viewership and the demand is increasing rapidly by the day and maintaining the information in a highly secured environment is one of the primary needs for any league conducting the events. In order to achieve this and to amplify the performance I am implementing Football Analytics that aims to:

1. Build a robust system that can store and retrieve data effectively and conveniently.
2. Create “constraints” to minimize data duplications.
3. Provide improved data security.

Football Analytics focuses on developing a database that maintains various events and information revolving around Football maintenance such as player information, event details etc. Though there are many databases that maintain this information in a secured level, my database deals with an effective/normalized way of storing and retrieving the data. Maintaining the records of various actors in Football is a cumbersome process and the presence of ambiguity and redundancy in the data further aggravates the difficulty. Hence, automating the entire system by building a robust database, makes data maintenance a less difficult job. The main users for accessing my database system will be Football players, supporting staff of the team. Apart from that, registered users will also be using the system in order to learn about various details of the players, events and other information involved in Football.

The main objective is to develop a database that would maintain records of various components related to Football. It is only logical if system is defined in terms of relationships rather than a mere representation of the components. The various components would include club information, club details, and player information among others. Each component has a distinct functionality and is inter-related. So, having a system that supports concurrent updating and minimizes duplicity is absolutely essential. Moreover, since, the league has planned to expand its base to areas that have seen a recent spurt in Football subscriptions. To accommodate the throngs of audience and supporters, the system has to have a robust system. My database holds top-level data of the football matches such as match date, competition name, participating teams, venues and environmental conditions. It also captures all macro-events occurring during the match such as goals, penalties, and disciplinary incidents. The biographical and physical data of players, clubs, club managers and supporting staff are also available in the database.

Finally, we intend to improve upon the proposed model as Football Analytics system is more evolutionary than revolutionary.

**BUSINESS AND DATA REQUIREMENT:**

The most essential requirement for maintaining the database is to have all the information about various events in one common place. Football has many leagues conducted throughout the world and it is ideal if all this information can be stored and retrieved in a single database. This prevents Football Franchisees to visit multiple systems resulting in time saving and as well as saving cost.

The system covers two leagues and each league has four teams. There are 5 players per team and the database was developed adhering to the common norms and establishing atomicity while meeting the business requirements. The person table holds the information of various players involved in the league, supporting staff, owner details and common users. The user category includes managers, players, club chairman, club secretary, club doctor, and common user. This table basically holds the information about all the active persons involved in accessing the Football database. This table captures all demographic information such as name, age, nationality etc.

The league table holds the information about various leagues whose information needs to be stored. The club directory table holds the information about various clubs and their history. The club table has various information such as its position in the league, number of games played/won/drawn/lost and its points for a particular league season. The most important table of all is the Squad, which holds the information of various players of a particular club. This table holds the information of player’s position in the field, whether forward, defense, midfield or attacking, goals scored, matches played, cards received for foul play, and wages of individual players. This information enables managers and various other clubs in the world to look into player’s profile and cast their interest for buying new player. They can look into each and every activity of the players, track their information match by match and even know about the past clubs where they played. As we all know, players form the basic foundation of Football and this system enables various users to knowing the information about players at the touch of single button.

There are other tables such as attacking, defending and goals which captures various information about the clubs such as total number of shots passed, number of crosses made, offside, goals scored, goals conceded etc. This information helps various teams to plan and build a strategy when they face the opponent.

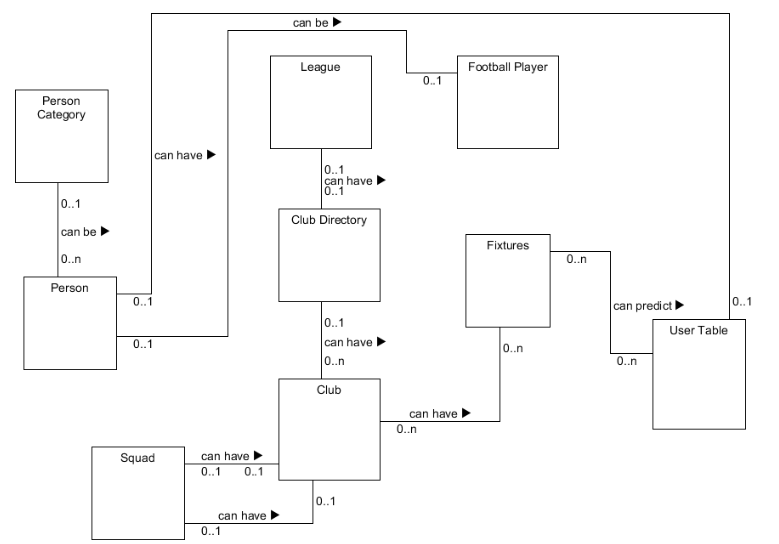
Other important tables are Fixtures and View Results where all the details about the matches played in the past and matches in future will be available. These are helpful to gather the information about the past matches for common users and as well as other actors accessing the database. The Franchise table holds the information about sponsors name, their budget and contract year with a particular club. This enables rest of the franchises to bid for sponsorship for any club in future.

There is a unique table for users to attract their attention. The users can forecast the winner of a match by predicting the goals scored in a match.

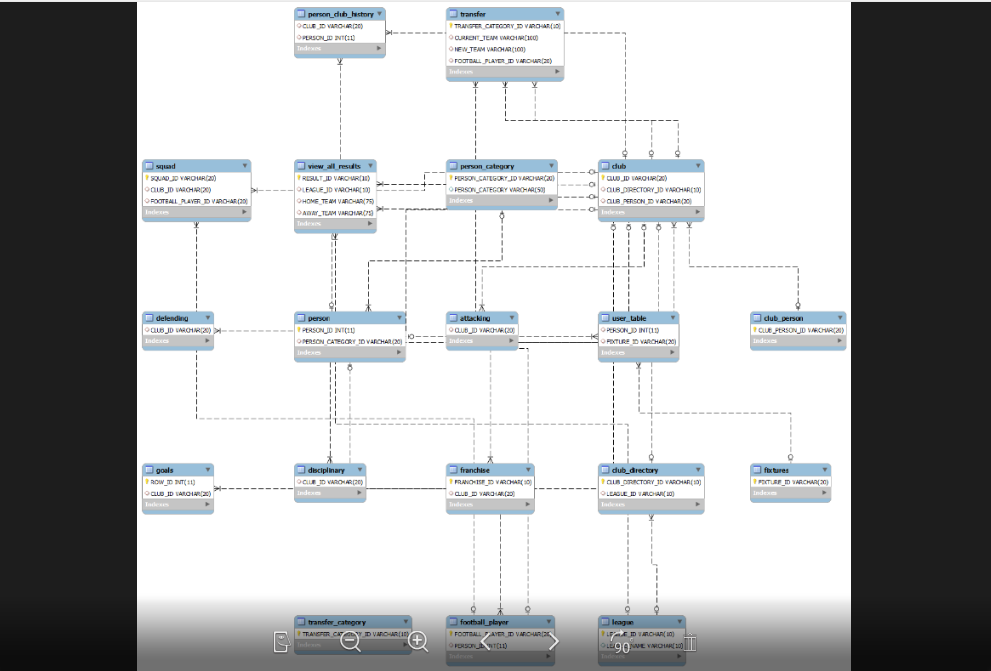
All table covers both super type and sub type. Tables like Club directory, Fixtures, League, and Person Category are super types. Tables such as club, squad are sub types that are dependent on super type tables and the link between these two tables are created using foreign keys.

There are various roles created in database and these roles have several privileges set to access the database. All the tables are normalized making sure the data redundancy, data complexity are reduced and faster retrieval of data can be achieved. Example, Player information are stored in a separate table named Football\_Player. The Squad table has the primary key of Football\_Player set as foreign key in it, so that the player information can be obtained by joining two table. The player may be changing the club from club to club and it would be ideal if their data are maintained in separate table. Similar pattern is followed for the creation of other tables in order to ensure that the tables are normalized.

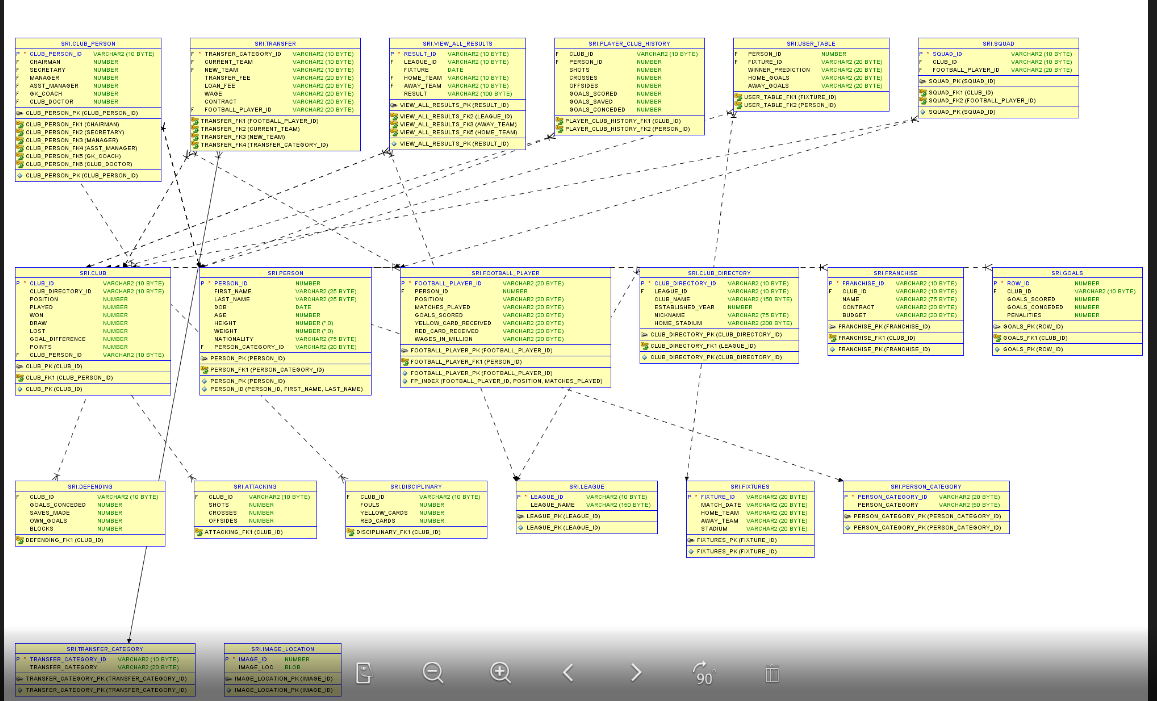
**CONCEPTUAL MODEL:**



**LOGICAL MODEL:**



**PHYSICAL MODEL:**



For better clarity, all these three models are provided as separate image files.

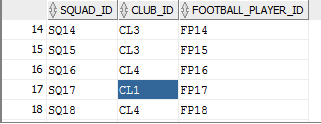
**Queries Output:**

Whenever a player is transferred from one club to another club (bought by a new club), a trigger is ‘after\_transfer\_insert’ executed which will call a procedure ‘update\_squad’.

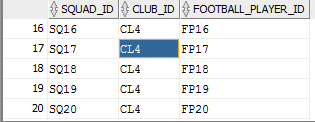
This procedure will change the club name in the club table to new club name. Apart from that, there is another table ‘PLAYER\_CLUB\_HISTORY’ which tracks the players statistics club wise. Here a new record will be inserted for the player for new team to which he joined.

Here, in this example, football player of ID FP17 is transferred from Club1(CL1) to club4(CL4).

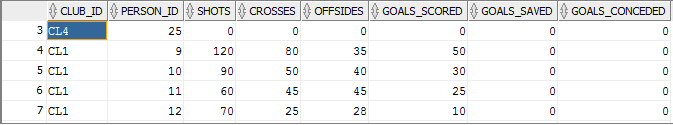
Before Trigger, the player is in club CL1.



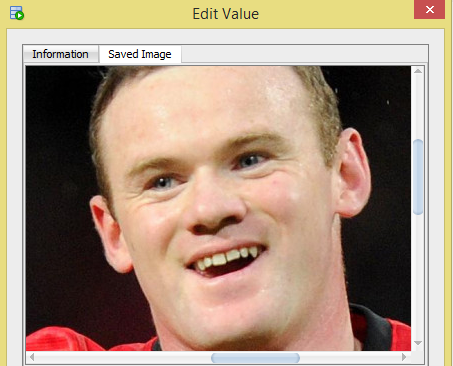
After Trigger, the player has been changed to new club CL4.

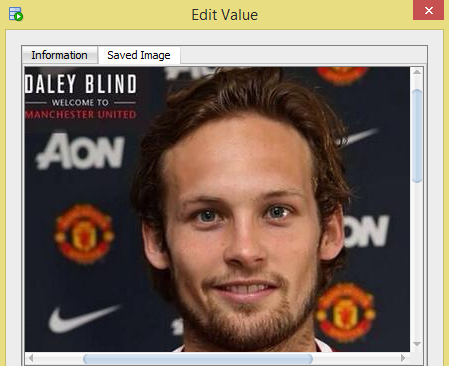


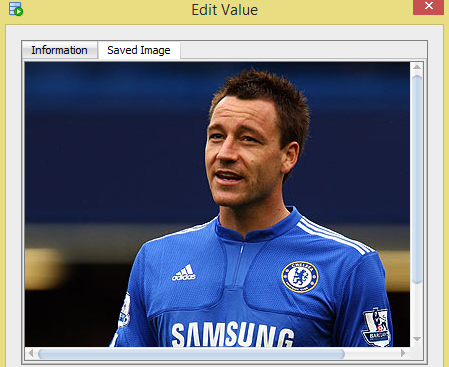
As the player is moved to new club CL4, his record (Goals, shots etc.) should be tracked for that particular club. Hence, the procedure updates the table with new club with all the record set to 0, so that it can be tracked from now on.



Another Procedure IMAGE\_INSERTION has been created to store the images of the players.

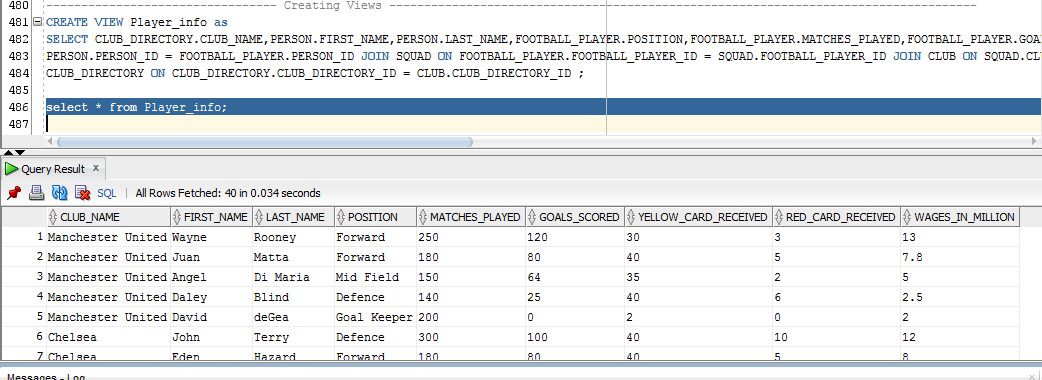




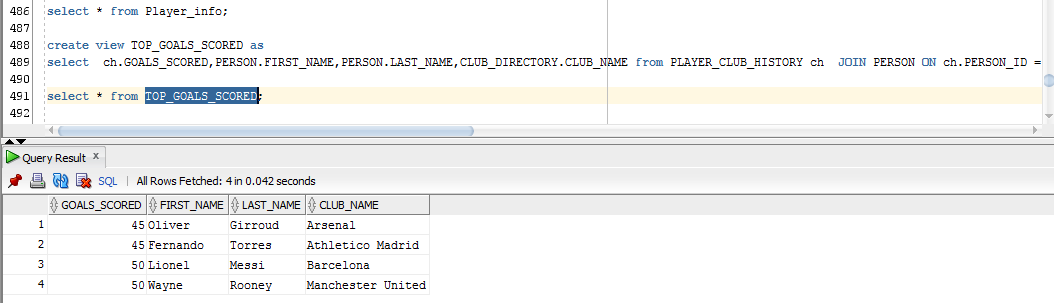


VIEWS:

A view Player\_info has been created to get the information of players and their club history.

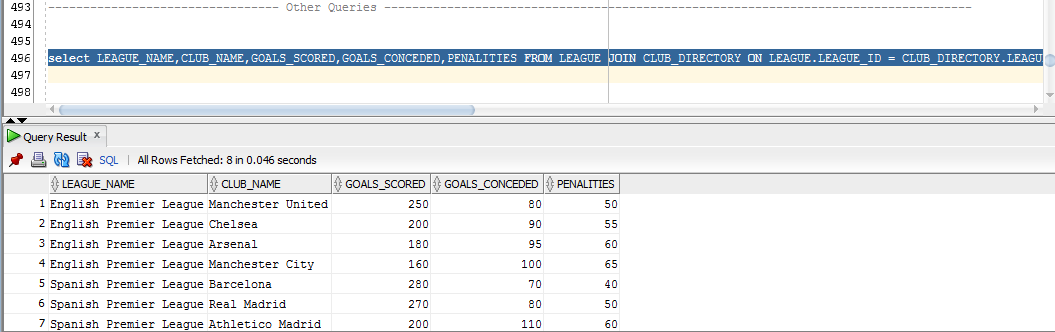


Another view TOP\_GOALS\_SCORED has been created to identify top 4 goals scored by player belonging to both the leagues.

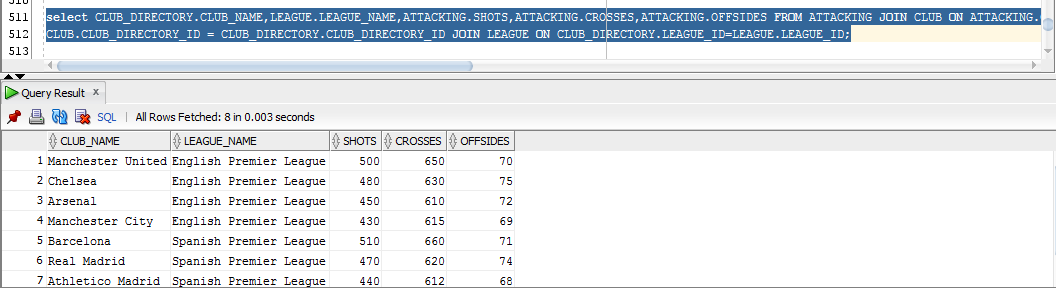


Other Queries:

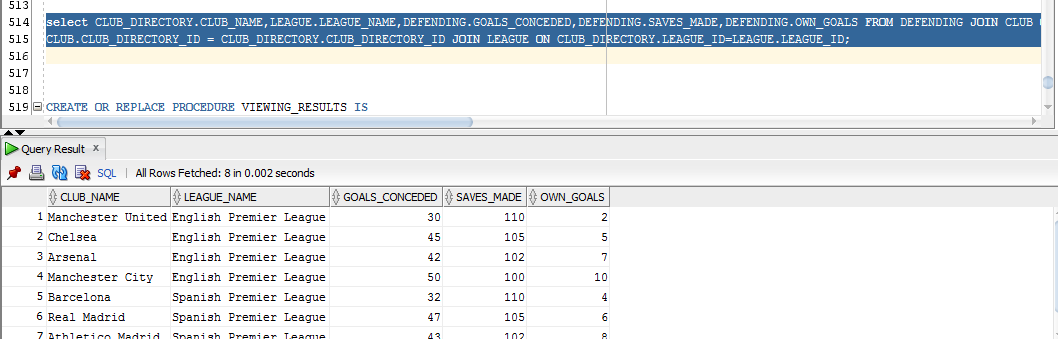
Below query provides the detail of individual clubs and their records.



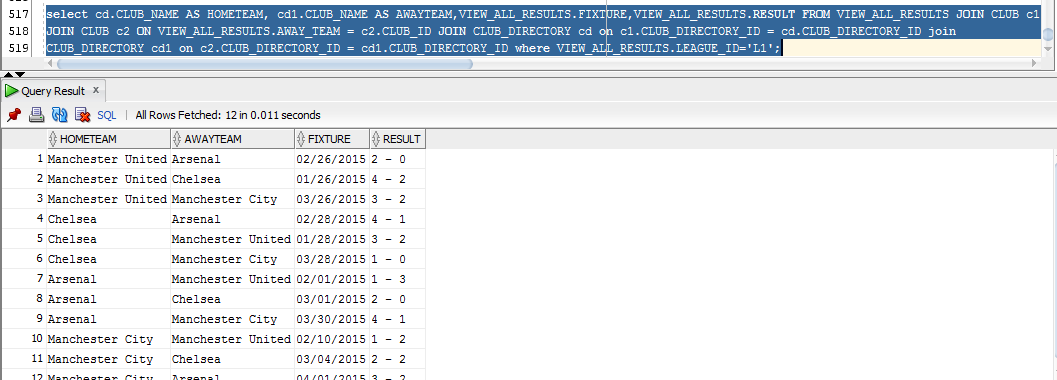
Below query tracks another record of the clubs from Attacking perspective.



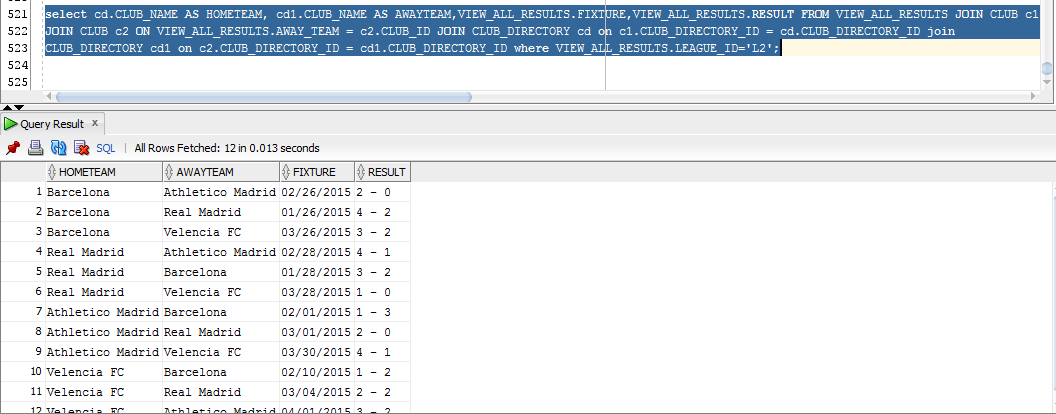
From Defending perspective:



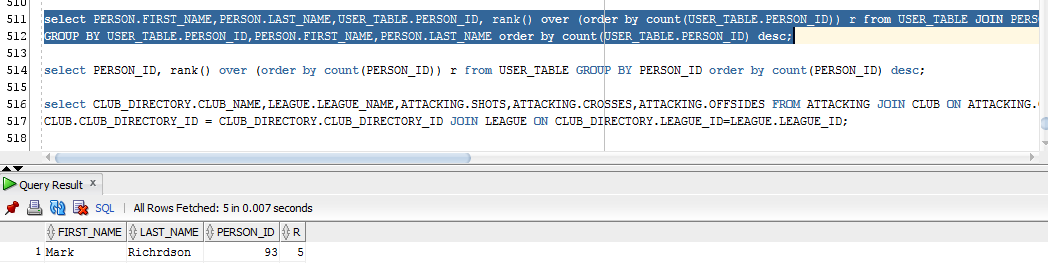
Below query retrieves all the results of the games played in League L1(English Premiere League)



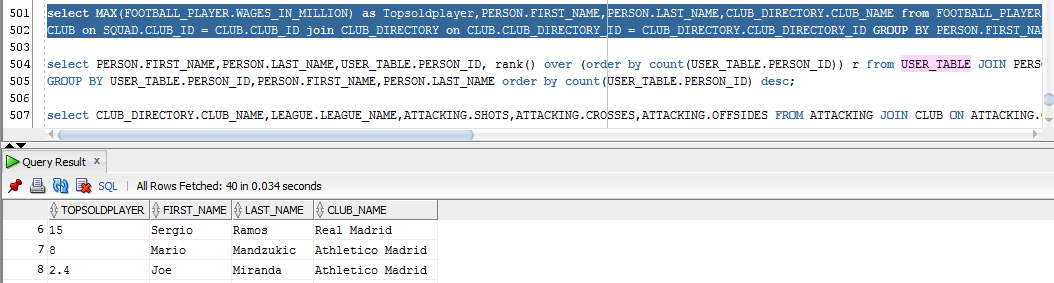
Below query retrieves all the results of the games played in League L2(Spanish Premiere League)



Query to find the user who predicted the result many times:



Query to find the wages of every Foot Ball.

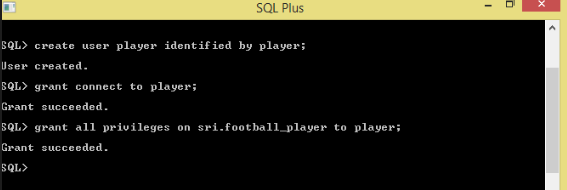


Indexes:

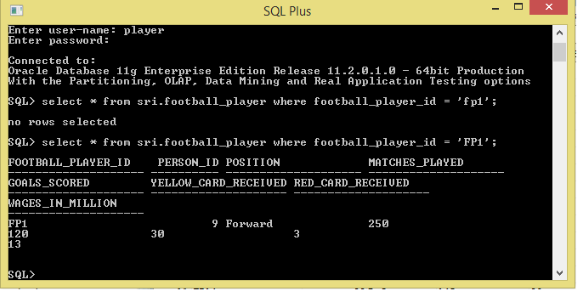
Indexes person\_id and fp\_index have been created for the tables PERSON and FOOTBALL\_PLAYER as these two tables have more number of records and will be helpful for the faster retrieval of data from these tables.

GRANTS:

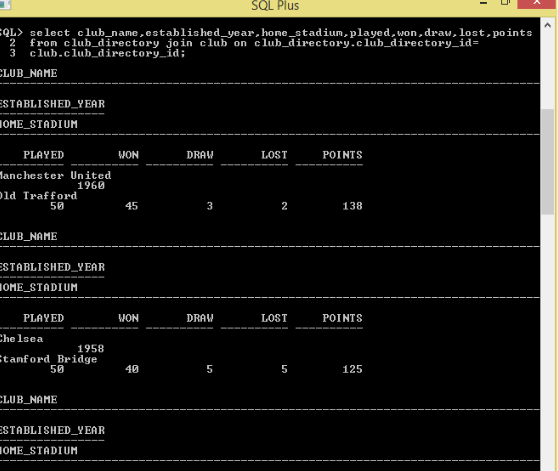
New user Player has been created and he has been given all privileges to FOOTBALL\_PLAYER table.



He now has access to FOOTBALL\_PLAYER table.



Another generic query executed on SQLPLUS:





**Database Maintenance plan:**

The database should be up and available most of the times in order to make sure that all the users accessing it should be benefited. The database maintenance plan should be designed and installed so that there would be always as backup plan in case of any real time disaster or any other database calamities.

Oracle Database has several automated maintenance tasks that are performed by database administrators. Normally these maintenance are executed whenever the system load is light. These tasks start automatically at regular intervals in order to perform maintenance operation in database. The time intervals for running the tasks are predefined and these tasks run in this window. These task are achieved by:

Automatic Optimizer Statistics Collection

Automatic Segment Advisor

Automatic SQL Tuning Advisor.

These days in a real time production environment, a technique named “Data Mirroring” is followed where all the production data will be mirrored into an alternate back up environment. All the data will be updated and saved in this environment similar to that of production. The mirroring will be taken in a particular interval but during recovery, last available image will be used. Apart from that, there are image copies which takes the backup of the database. These image copies in the form jobs or any other means can be executed prior to updating the database and it contains all the information of database prior to current day. Once after the execution of these image copies, the current day database can be updated. In this manner, back up can be made on a daily basis.

Whenever a disaster occurs, the recent image prior to the disaster timing will be recovered in the Business Continuation site. Verification will be made base on the time of disaster and whatever transactions that got hung up will be restarted. The database can also be recovered from recent images taken by the image copies and can be brought up and available in production. In order to make sure that the recovery process takes place in a normal manner, business continuation test should be executed in a test environment to make sure the proper recovery is done. These tests can be conducted twice or thrice a year.

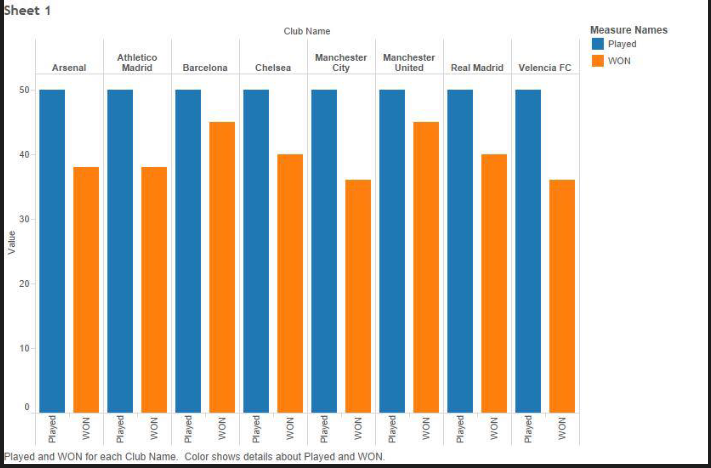
A failover is when primary database fails and one of the standby database is transitioned to take the primary role. Failover is performed whenever primary database fails and primary database cannot be recovered in timely manner. While performing a fail over, the standby database that has most archived data in it should be chosen. By doing this, amount of data loss can be minimized. If the configuration has both physical and logical databases, it is always effective to choose a physical standby database because all standby databases in the configuration will still be available as standby databases to the new primary database after the failover operation completes.

**BI TOOLS:**

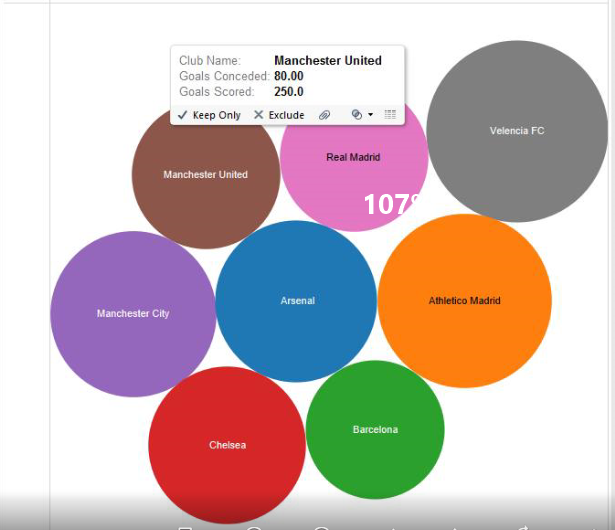
The data from the tables can be loaded into BI tools and we can do various analysis using that.

Here I have used Tableao tool to generate the below mentioned charts.

Below graph shows the number of matches won by each club in all the two leagues.

****

The below graph shows the number of goals scored and conceded by each team in the league.



The below graph shows the distribution of players across the world. The players are from region that has green shades.

