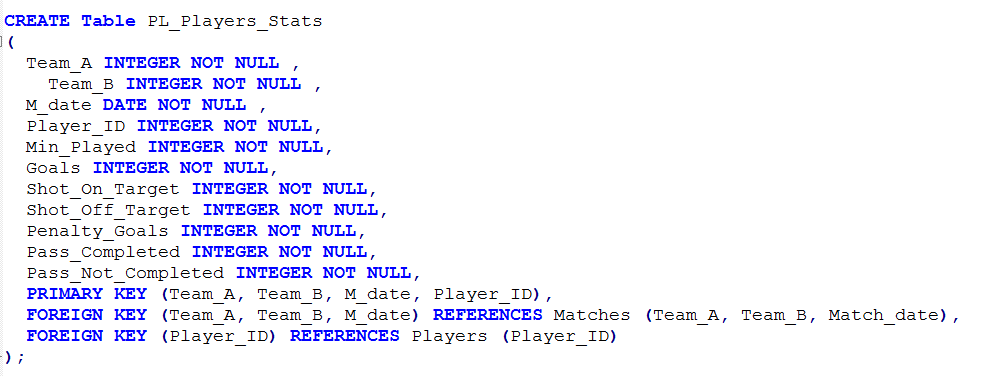
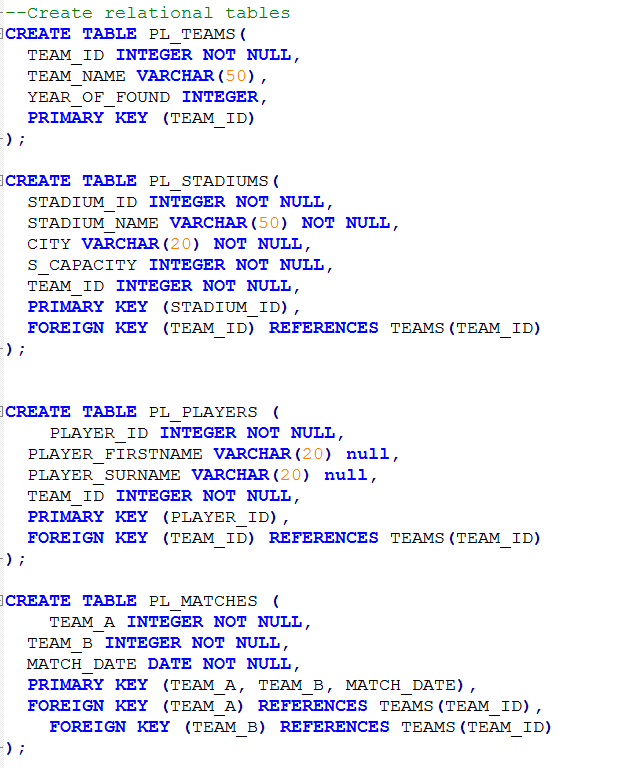
DATABASES ASSIGNMENT

**Student name:** Ronan Dillon

**Student number:** C12335251

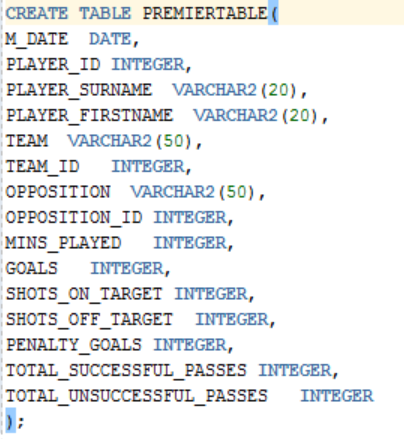
**PART A**

For part A I used 5 create table statements to make up the relational model. I entered all their fields and gave the tables primary and foreign keys with the correct corresponding tables.

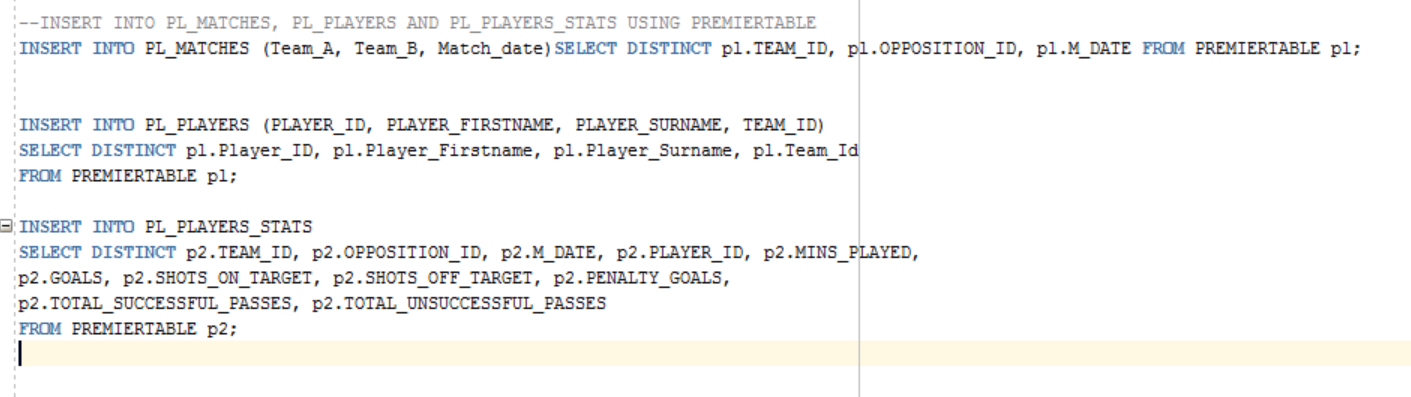


**PART B**

I started part B by creating a denormalized table. I then imported the csv file premier.csv into the denormalized table.

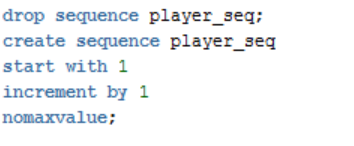


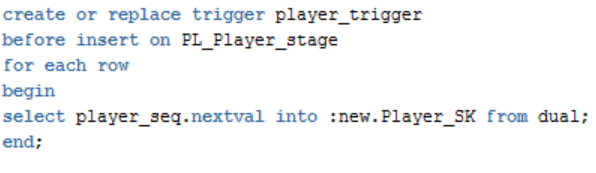
I then used the statements in insert.sql to insert data into the PL\_TEAMS and PL\_STADIUMS tables. After that I wrote insert statements to insert into the other three tables using data in PREMIERTABLE.

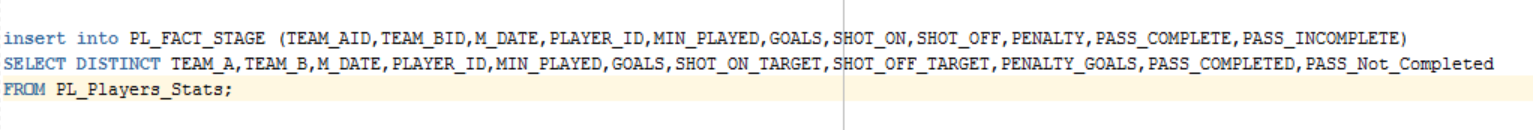


**PART C**

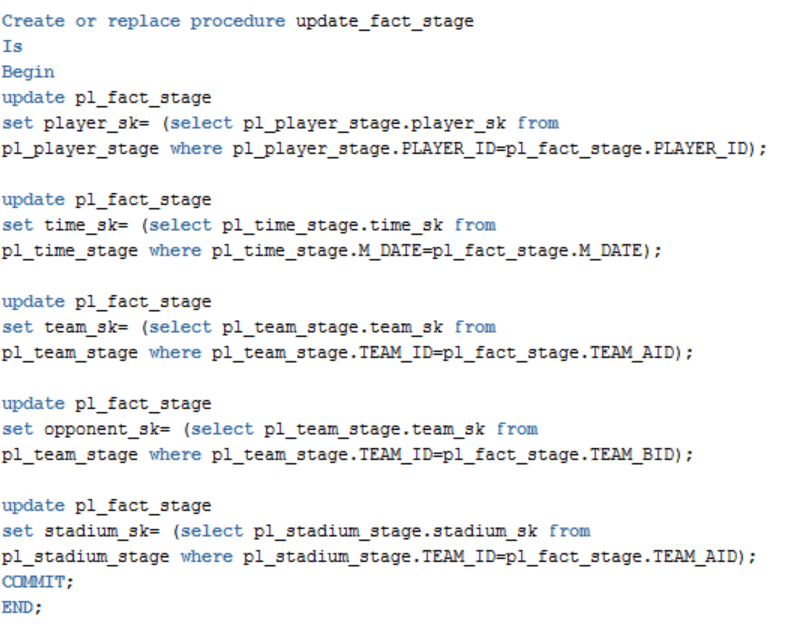
For part C I created stage tables, using the data in my relational model I moved data from the relational tables into the stage tables. Using sequences and triggers a new and unique SK was given to each stage table. The example below of the player trigger shows that before inserting into the PL\_PLAYER\_STAGE table the next value in the player sequence is added the player\_sk. After adding to the stage table a procedure is called to update the SKs in the fact stage table so that they match up with the SKs in the other stage tables.



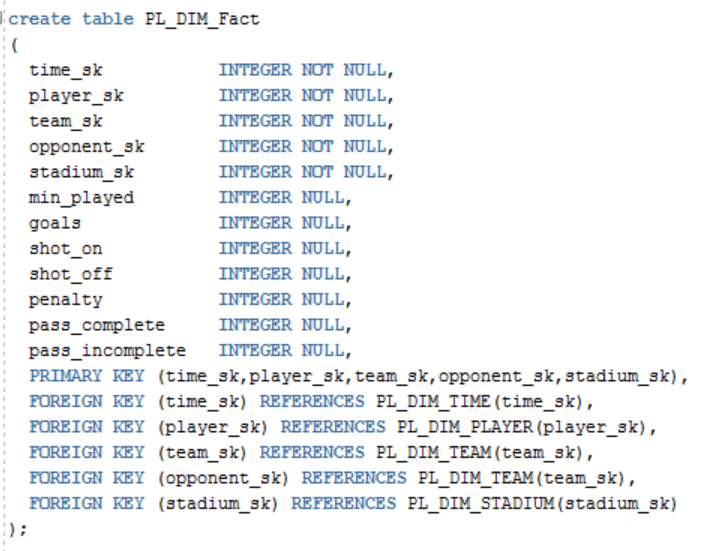


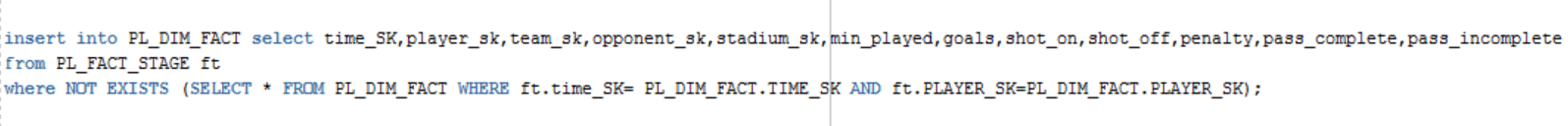


Although none of the dimensional tables contain team A and B IDS, match date or player ID, I have put them in the stage tables to use for adding the SKs to the fact stage table. I use the values in the fact stage to match up with the corresponding stage tables.



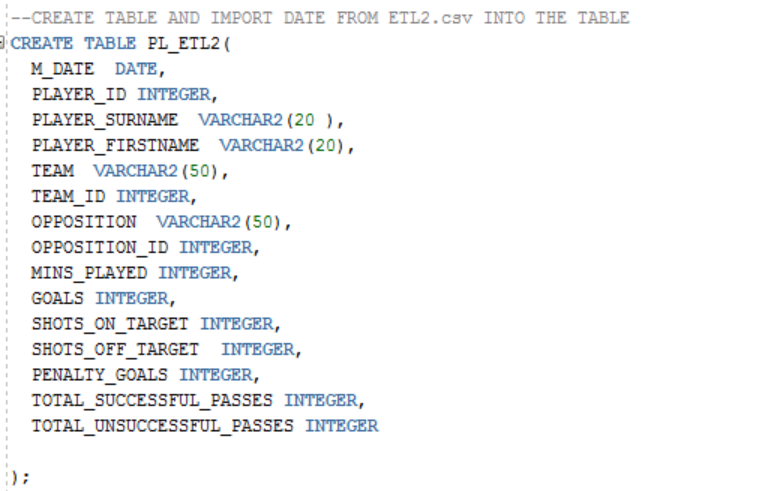
After that I created the dimensional tables and moved the data from the stage tables into their corresponding dimensional table.

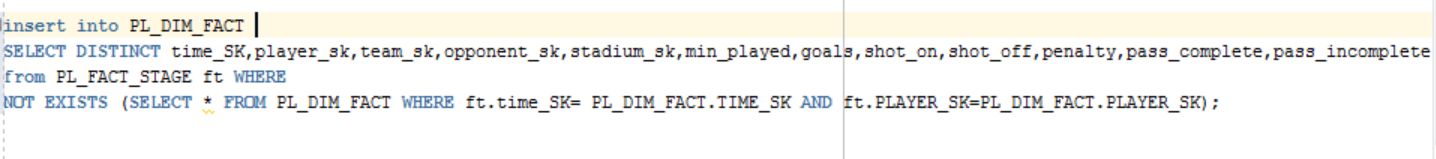




**PART D**

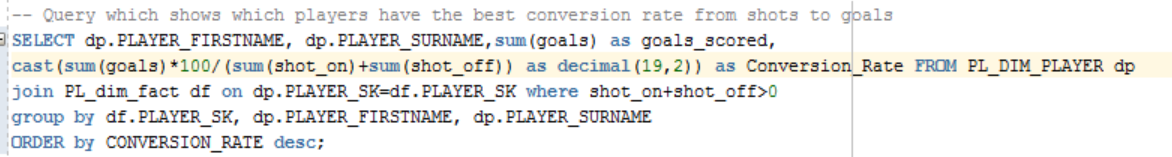
In part D I created a denormalized table like in part A and imported data from etl2.sql into the table. From there I moved data from the denormalized tables into the stage tables. I recalled the procedure for the fact stage table and then moved all the new data over to the dim tables. Using NOT EXISTS to make sure that there would be no dupes in the dim table from the new set of players.



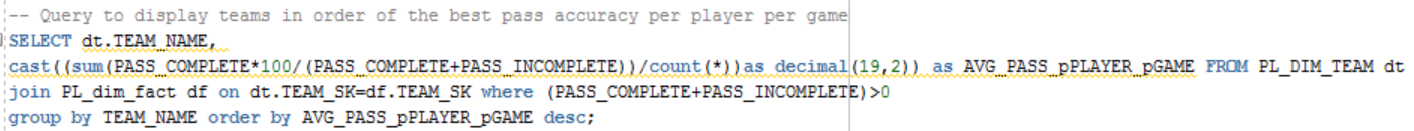


**PART E**

For the two queries I had to create which can be executed over the dimensional model to create reports about teams and players I chose to create one to show which players have the best conversion rate i.e. which player has the best goal to shot ratio. E.g. a player who has scored 3 goals and taken a total of 6 shots has a conversion rate of 50%. The query takes the sum of each player’s goals from every game and divides it by the sum of shots on and off target in every game the player has played. This is multiplied by 100 to get a percentage. There is a where clause which specifies that if a player hasn’t had any shots on or off target that it is not included as no player has scored with no shots.



The second query I made is to create a report about teams. This query returns the average pass accuracy per player per game. The query takes the pass accuracy of each player in every match they played and adds them all together for each team. This total is divided by the amount of players matches in the sum (count(\*)). This will return the average pass accuracy for each team, and it is sorted in descending order so the team with the best average pass accuracy per player per game is at the top and that team is Manchester United.



**Summary**

To summarise the assignment we were required to complete five tasks which were;

1. Create a relational model using Oracle or MySQL. Implementing the primary and foreign key constraints.
2. Populate the data by loading the data from the sql script and from the csv into the relational model.
3. Perform an initial ETL process to move data into the data warehouse. Define a sql script that automatically executes the process, creating the required stage tables and all the data matching procedures.
4. Perform a second ETL using the additional data contained in the etl2.csv files. Load the data into some staging table and then perform the second ETL reusing some of the stage tables and procedures used for the first ETL.
5. Provide 2 sample queries that could be executed over the dimensional model to create reports about teams and players.

I feel I have completed the required tasks with each part working as it is expected to.