

**GROUP PROJECT**

**DATA WAREHOUSING (BSD2343)**

**TITLE : FOOTBALL DATABASE OF TOP 5**

**EUROPEAN LEAGUES**

**LECTURER’S NAME : DR. NOR AZUANA RAMLI**

**PREPARED BY**

|  |  |
| --- | --- |
| **NAME** | **MATRIC ID** |
| MUHAMMAD FARHAD BIN KHAHARRUDDIN | SD20028 |
| HAFIZ ASYRAAF BIN GHANI | SD20033 |
| MUHAMMAD ISYHRAF BIN AZMIN | SD20065 |
| KUEH FUK CHEN | SD20049 |

Contents

[1.0 Background 2](#_Toc106172712)

[1.1 Introduction 2](#_Toc106172713)

[1.2 Problem to be solved 2](#_Toc106172714)

[1.3 Objective 2](#_Toc106172715)

[1.4 Description of dataset 3](#_Toc106172716)

[1.5 Data Schema 6](#_Toc106172717)

[2.0 Architecture 12](#_Toc106172719)

[2.1 Pipeline structure 12](#_Toc106172720)

[2.2 Description of the process flow 12](#_Toc106172721)

[3.0 Database 14](#_Toc106172722)

[3.1 Relational model and relationship between data 14](#_Toc106172723)

[3.1.1 Relational Model 14](#_Toc106172724)

[3.1.2 The Relationship Between Data 15](#_Toc106172725)

[3.1.3 Identification of data warehouse schema 15](#_Toc106172726)

[4.0 ETL pipeline 16](#_Toc106172727)

[5.0 Result and data analysis 29](#_Toc106172728)

[5.1 Data Visualization 32](#_Toc106172729)

[6.0 Conclusion 35](#_Toc106172730)

[7.0 References 36](#_Toc106172731)

# 

# **1.0 Background**

## **1.1 Introduction**

Football, also called association footballorsoccer, is a game in which two teams of 11 players, using any part of their bodies except their hands and arms, try to maneuver the ball into the opposing team’s goal. The name of the sport was derived by the way it is played and this sport took its shape in the 18th century. Football is the sport that reaches the most people and it is the world’s most popular ball game in numbers of participants and spectators. A football game generates many more events and it is very important and interesting to take into account the context in which those events were generated.

By the early 20th century, football had spread across Europe, but it was in need of [international organization](https://www.britannica.com/topic/international-organization). We can see that the football games have spread around the world. There are international competitions established in Europe in which numerous clubs compete for a prestigious cup. In this case study, we just focus on the top 5 leagues in Europe which are Premier League, Serie A, Bundesliga, La Liga and Ligue 1 from 2014 to 2020. We need to visualize all the datasets to identify the correlations between the relationship of the dataset that we collect. This dataset contains football-related data covering the top 5 leagues in Europe from 2014-2020.

## **1.2 Problem to be solved**

As we know, football is the most popular sport worldwide and many countries take part in this game. There are many types of tournaments organized to find out which team is the best in football around the world such as FIFA World Cup, Olympic Football Tournament and European Championship. So, we can see that many countries took part in these tournaments to win the games. In this study, we focus on the top 5 leagues in Europe from 2014-2020. In this project, we need to show the relationship between 5 leagues from the dataset that we collect and make the comparison based on the datasets.

## **1.3 Objective**

* To build expected goals models and compare players
* To make a comparison between top 5 European leagues
* To show out the relationship from the datasets
* To cleanup and transform the data
* To create an interactive dashboard that allows users to input certain criteria regarding football dataset

## **1.4 Description of dataset**

From the table “games”

|  |  |  |
| --- | --- | --- |
| Data Attribute | Data Type | Description |
| GAME\_ID | Integer | The unique ID for a game played |
| LEAGUES**\_**ID | Integer | The leagues’ ID |
| SEASON | Integer | The game’s season |
| HOME\_TEAM\_ID | Integer | The unique ID for home team |
| AWAY\_TEAM\_ID | Integer | The unique ID for away team |
| HOME\_GOALS | Integer | The unique ID for home goals |
| AWAY\_GOALS | Integer | The unique ID for away goals |

From the table “shots”

|  |  |  |
| --- | --- | --- |
| Data Attribute | Data Type | Description |
| GAME\_ID | Integer | The unique ID for a game played |
| SHOOTER**\_**ID | Integer | The unique ID for shooter |
| ASSISTER\_ID | String | The assister ID |
| MINUTES | Integer | The time that team score |
| LAST\_ACTION | Char | The last action |
| SITUATION | Char | The situation from the team |

From the table “teamstats”

|  |  |  |
| --- | --- | --- |
| Data Attribute | Data Type | Description |
| GAME\_ID | Integer | The unique ID for a game played |
| TEAM\_ID | Integer | The unique ID for team |
| SEASON | Integer | The game’s season |
| DATE | Date | The game’s date |
| LOCATION | Char | The game’s location |
| GOALS | Integer | The goals that the team score |
| SHOTS | Integer | The number of shots |
| SHOTS\_ON\_TARGET | Integer | The number of shots on target |

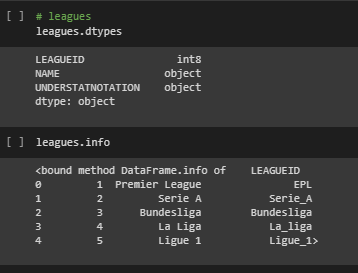
From the table “appearances”

|  |  |  |
| --- | --- | --- |
| Data Attribute | Data Type | Description |
| GAME\_ID | Integer | The unique ID for a game played |
| PLAYER\_ID | Integer | The unique ID for player |
| GOALS | Integer | The goals that the team score |
| OWN\_GOALS | Integer | The own goals that the team score |
| SHOTS | Integer | The number of shots |
| ASSISTS | Integer | The number of assists |
| KEY\_PASSES | Integer | The number of key passes |
| POSITION | Char | The position of the player |
| POSITION\_ORDER | Integer | The number of position order |
| YELLOW\_CARD | Integer | The number of yellow cards |
| RED\_CARD | Integer | The number of red cards |
| TIME | Integer | The game’s time |
| SUBSTITUTE\_IN | Integer | The number of substitutes in |
| SUBSTITUTE\_OUT | Integer | The number of substitutes out |
| LEAGUE\_ID | Integer | The league’s ID |

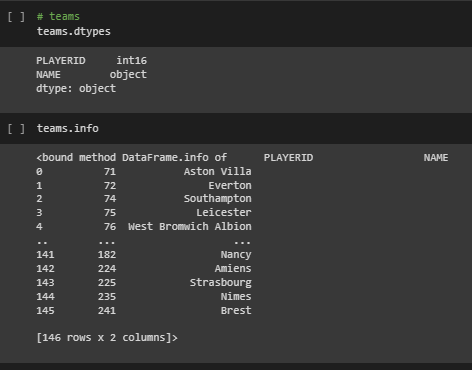
## **1.5 Data Schema**

# 

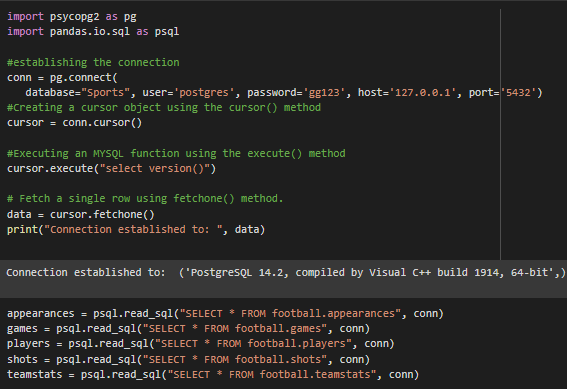
A database schema defines how data is organized within a relational database; this is inclusive of logical constraints such as table names, fields, data types, and the relationships between these entities. Basically we have 7 tables which are appearances, players, teamstats, games, leagues, teamstats and shots. But, 2 tables are imported from snowflake to pandas, which is leagues and teams by using snowflake.connector.



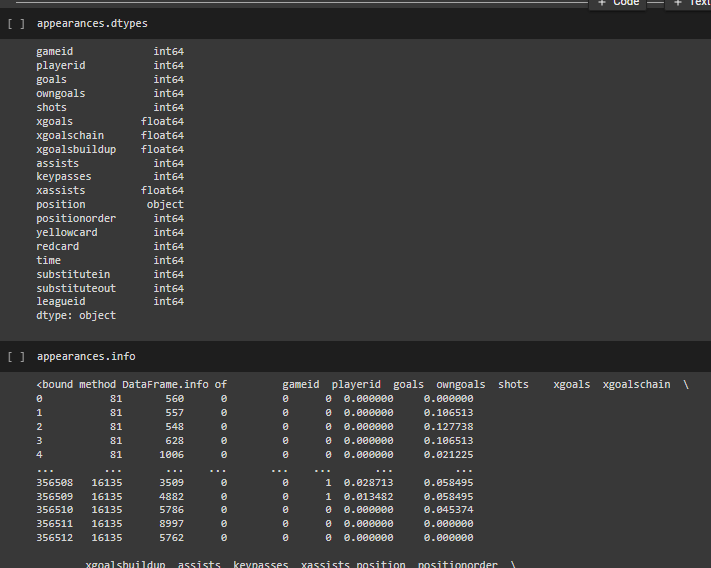
These data are about the leagues, which is the only 5 top european leagues. The outcome has 3 columns which is LEAGUEID, NAME, UNDERSTATNOTATION. That has 2 of the data types are int8 and object.



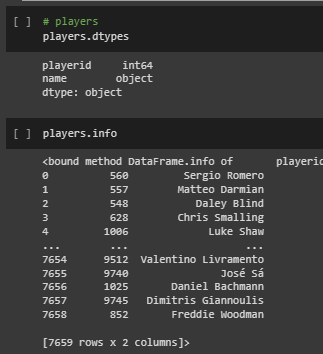
These data are about the teams in the league selected from the dataset. The outcome has 2 columns which are PLAYERID, NAME. That has 2 of the data types are int16 and object.

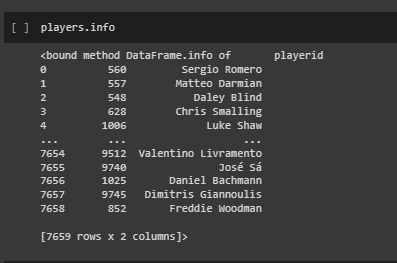


The second tool of importing dataset is from postgre to pandas by using psycopg2 and pandas.io.sql. The tables imported are appearances, games, players, shots and teamstats.

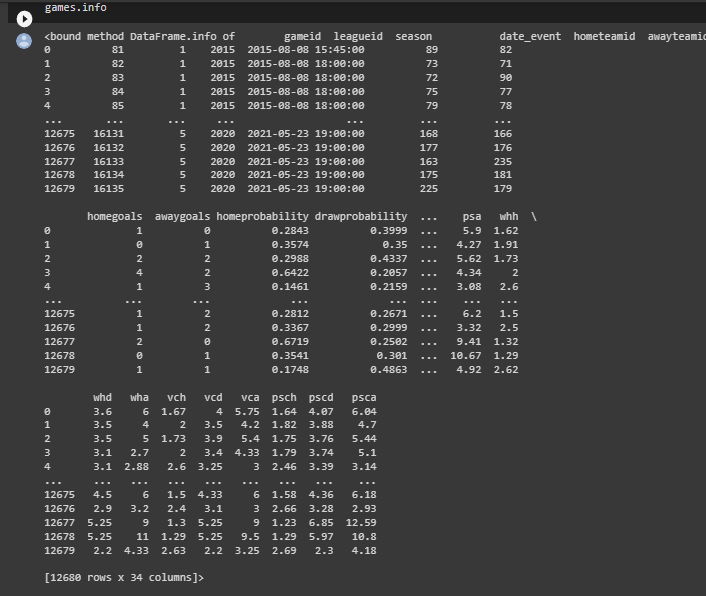
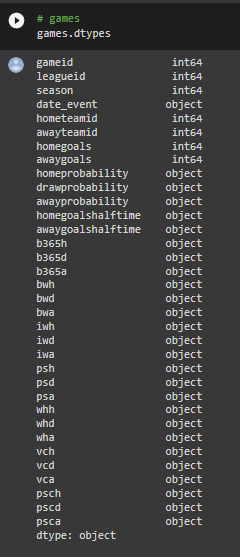


This table is about the appearances selected from the dataset. The outcome has 19 columns which is gameid, playerid, goals, owngoals, shots, xgoals, xgoalschain, xgoalsbulldup, assists, keypasses, assists, position, positionorder, yellowcard, redcard, time, substitutein, substituteout, leagueid. That has 2 of data type are int64 and object.

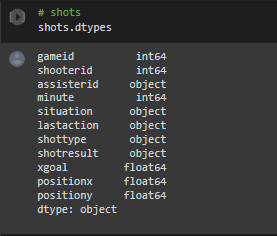


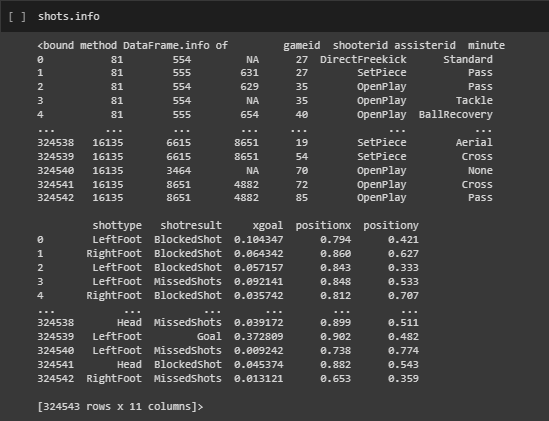


This data is about the players in the league selected from the dataset. The outcome has 2 columns which are playerid, name. That has 2 of data type are int64 and object.

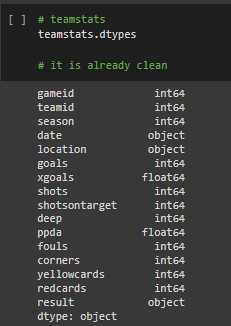


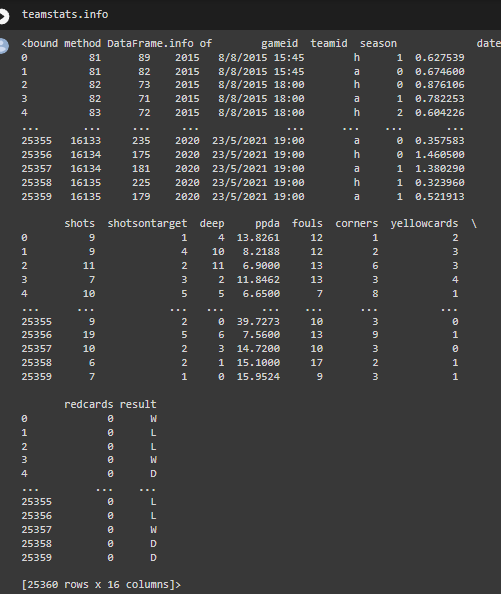
This table is about the games in the team selected from the dataset. The outcome has 34 columns which is gameid, leagueid, season,date\_event, hometeamid, awayteamid, homegoals, awaygoals, homeprobability, drawprobability, awayprobability, homegoalshalftime, awaygoalshalftime, b365h, b365d, b365a, bwh, bwd, bwa, iwh, iwa, psh, psd, psa, whh,whd,wha, vch, vcd, vca, psch, pscd, psca. That has 2 of data type are int64 and object.





This data is about the shots in the league selected from the dataset. The outcome has 11 columns which are gameid, shooterid, assisterid, minute, situation, lastaction, shottype, shotresult, xgoal, positionx, positiony. The 3 of data type are int64, float64 and object.

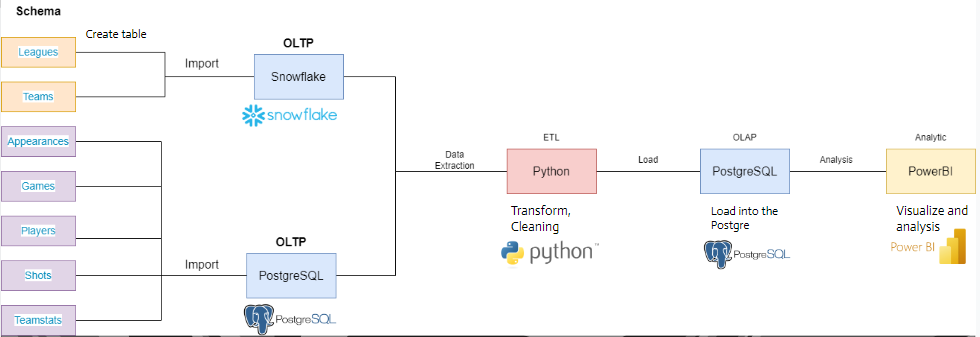




This data is about the shots in the teamstats selected from the dataset. The outcome has 16 columns which are gameid, teamid, season, date, location, goals, xgoals, shotsontarget, deep, ppda, fouls, corners, yellowcards, redcards, result. That has 3 of data type are int64, float64 and object.

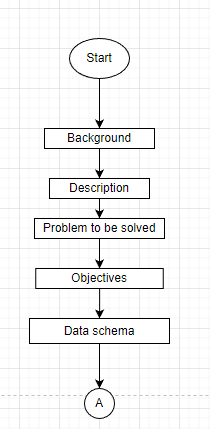
# **2.0 Architecture**

## **2.1 Pipeline structure**



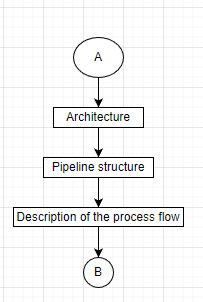
We have used a few tools to complete this project. In the OLTP process, we have used Snowflake for 2 tables while another 5 tables, we used Postgre. We extract the data from the database to Python to do Transforming and cleaning. We have used some packages during the Transform process, such as pandas, psycog2, numpy, snowflake.connector. We used Postgre to do OLAP process. Then, in the visualization process, we used Power BI to show the graph and analyze the data.

## **2.2 Description of the process flow**



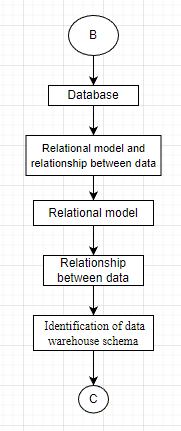
About the introduction to this project, the background of the project is where the project starts and gives the audience hints about what this project is all about. This part of the background has four parts: an explanation of the assignment, a description of the dataset by table, the objectives , the problem that needs to be solved, and the data schema.

Figure 2.2.1



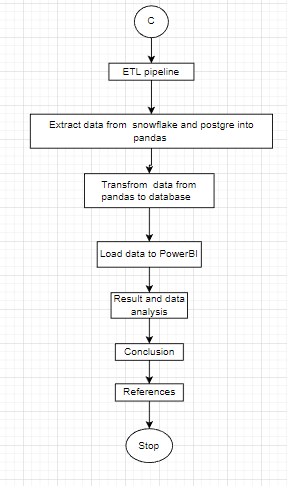
The architecture process is to show the pipeline structure. Description of the process shows the details of the whole process in this project.

**Figure 2.2.2**



The dataset in every table has the relationship between each attribute. The identification of the data schema is to know the type of the schema whether it is Star schema, Snowflake schema and Galaxy schema

**Figure 2.2.3**

**Figure 2.2.4 **

ETL pipeline is to show the Extract, Transform and Load details of the process. OLAP and vizualization of the data analysis shown during this step. The conlusion also know from the vizualization.

# **3.0 Database**

## **3.1 Relational model and relationship between data**

### **3.1.1 Relational Model**

### **3.1.2 The Relationship Between Data**

|  |  |  |
| --- | --- | --- |
| Entity | Entity | Relationship |
| games | shots | 1:M |
| games | teamstats | 1:M |
| games | appearances | 1:M |
| players | appearances | 1:M |
| leagues | appearances | 1:M |
| leagues | games | 1:M |
| teams | teamstats | 1:M |

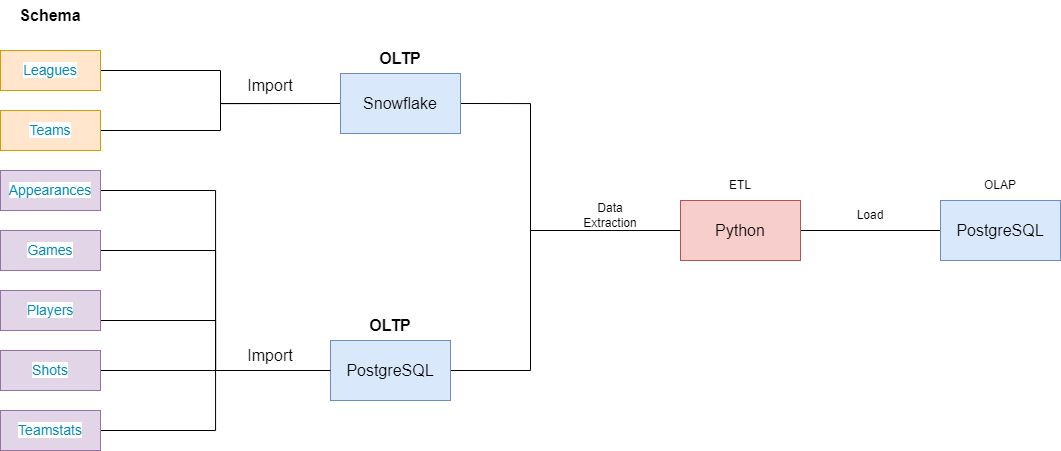
### **3.1.3 Identification of data warehouse schema**

The data warehouse schema of this dataset is **Snowflake Schema.**

# 

# **4.0 ETL pipeline**

For the first stage of our ETL process, we are going to need multiple databases to demonstrate the data integration. We choose PostgreSQL and Snowflake for our database.



**Figure 1** Data split

In figure 1, it shows the split process of our tables to load to two different database. For our project, we choose the **Top Down** approach. The data flow in the top-down OLAP environment begins with the data extraction from the ODS. This data is load into the staging area and validated and consolidated for ensuring a level of correctness and then moved to the ODS. The ODS stage is sometimes skipped if it is another copy of the operational databases.

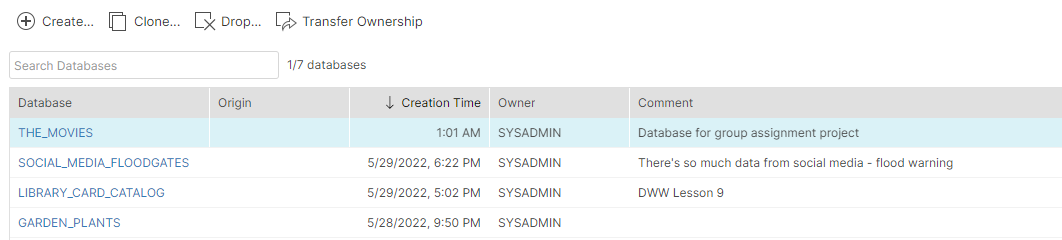
The process of data import into Snowflake will be shown below.

**Snowflake**

1. First we want to create the database and schema to hold our data by running create query.

CREATE DATABASE SPORTS;

CREATE SCHEMA Football;



2. Next, we will create table template to load our data later.

CREATE TABLE "SPORTS"."FOOTBALL"."APPEARANCES" ("GAMEID" INTEGER NOT NULL, "PLAYERID" INTEGER, "GOALS" INTEGER, "OWNGOALS" INTEGER, "SHOTS" INTEGER, "XGOALS" INTEGER, "XGOALSCHAIN" INTEGER, "XGOALSBUILDUP" INTEGER, "ASSISTS" INTEGER, "KEYPASSES" INTEGER, "XASSISTS" INTEGER, "POSITION" VARCHAR(3), "POSITIONORDER" INTEGER, "YELLOWCARD" INTEGER, "REDCARD" INTEGER, "TIME" INTEGER, "SUBSTITUTEIN" INTEGER, "SUBSTITUTEOUT" INTEGER, "SUBSTITUTEOUT2" INTEGER, "LEAGUEID" INTEGER)

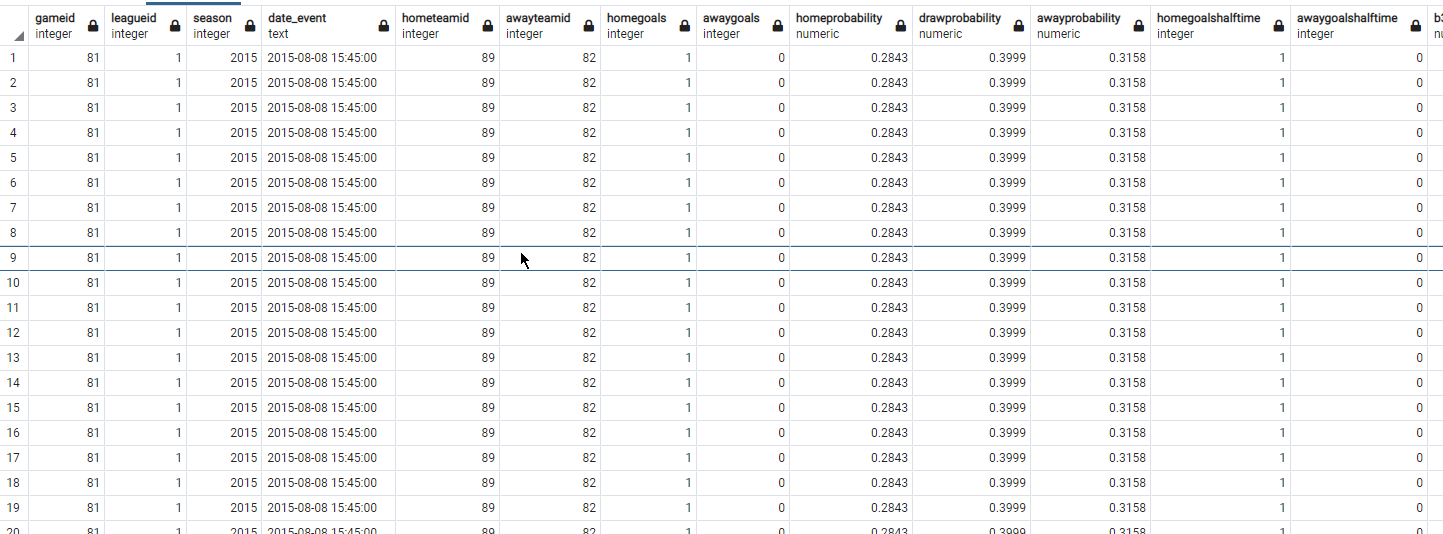
3. We create a file format according to our csv parameters.

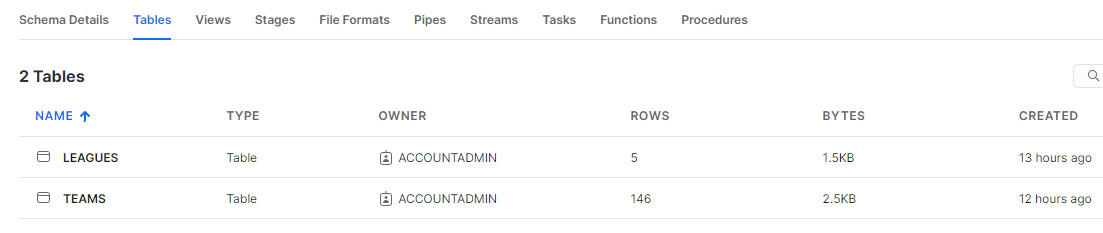
CREATE FILE FORMAT "SPORTS"."FOOTBALL".CSV TYPE = 'CSV' COMPRESSION = 'AUTO' FIELD\_DELIMITER = ',' RECORD\_DELIMITER = '\n' SKIP\_HEADER = 1 FIELD\_OPTIONALLY\_ENCLOSED\_BY = '\042' TRIM\_SPACE = FALSE ERROR\_ON\_COLUMN\_COUNT\_MISMATCH = TRUE ESCAPE = 'NONE' ESCAPE\_UNENCLOSED\_FIELD = '\134' DATE\_FORMAT = 'AUTO' TIMESTAMP\_FORMAT = 'AUTO' NULL\_IF = ('\\N');

4. Lastly. We upload our csv files into the database by using automated staging of Snowflake and by specifying file format that we create previously.

PUT file://<file\_path>/credits.csv @CREDITS/ui1654967844033

COPY INTO "THE\_MOVIES"."PUBLIC"."CREDITS" FROM @/ui1654967844033 FILE\_FORMAT = '"THE\_MOVIES"."PUBLIC"."CSV"' ON\_ERROR = 'ABORT\_STATEMENT' PURGE = TRUE;





Snowflake Tables Structure

**PostgreSQL**

1. First, we create a table called ‘Sports’ and schema called ‘football’.

create database ‘Sports’;

set schema 'football';

2. We proceed to create all the tables that we are going to import.

CREATE TABLE games (

gameID int,

leagueID int,

season int,

date\_event text,

homeTeamID int,

awayTeamID int,

homeGoals int,

awayGoals int,

homeProbability text,

drawProbability text,

awayProbability text,

homeGoalsHalfTime text,

awayGoalsHalfTime text,

B365H text,

B365D text,

B365A text,

BWH text,

BWD text,

BWA text,

IWH text,

IWD text,

IWA text,

PSH text,

PSD text,

PSA text,

WHH text,

WHD text,

WHA text,

VCH text,

VCD text,

VCA text,

PSCH text,

PSCD text,

PSCA text

)

CREATE TABLE shots (

gameID int,

shooterID int,

assisterID text,

minute int,

situation text,

lastAction text,

shotType text,

shotResult text,

xGoal numeric(7,6),

positionX numeric(4, 3),

positionY numeric(4, 3)

)

CREATE TABLE teamstats(

gameid int,

teamid int,

season int,

date text,

location text,

goals int,

xgoals decimal,

shots int,

shotsontarget int,

deep int,

ppda decimal,

fouls int,

corners int,

yellowcards int,

redcards int,

result text

);

CREATE TABLE appearances (

gameID int,

playerID int,

goals int,

ownGoals int,

shots int,

xGoals numeric(7,6),

xGoalsChain numeric(7,6),

xGoalsBuildup numeric(7,6),

assists int,

keyPasses int,

xAssists numeric(7,6),

position text,

positionOrder int,

yellowCard int,

redCard int,

time int,

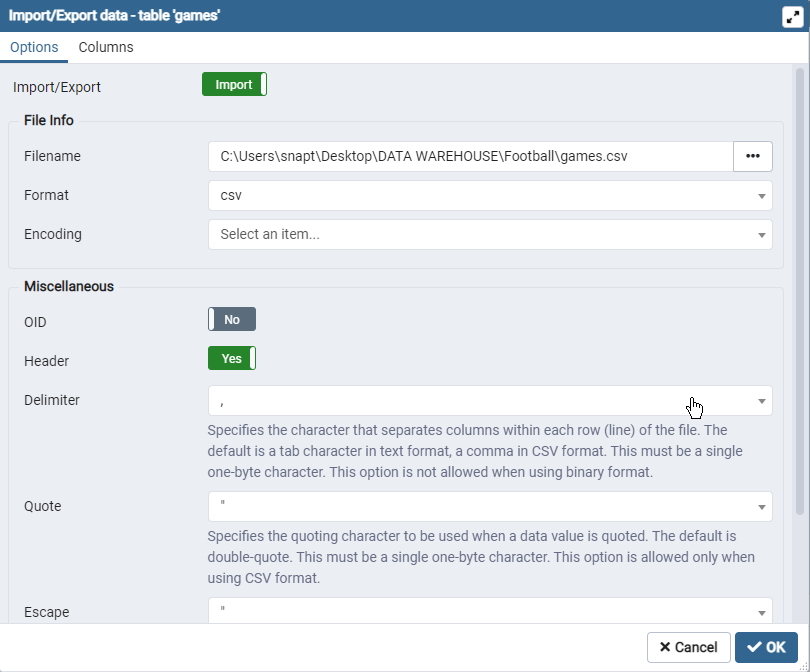
substituteIn int,

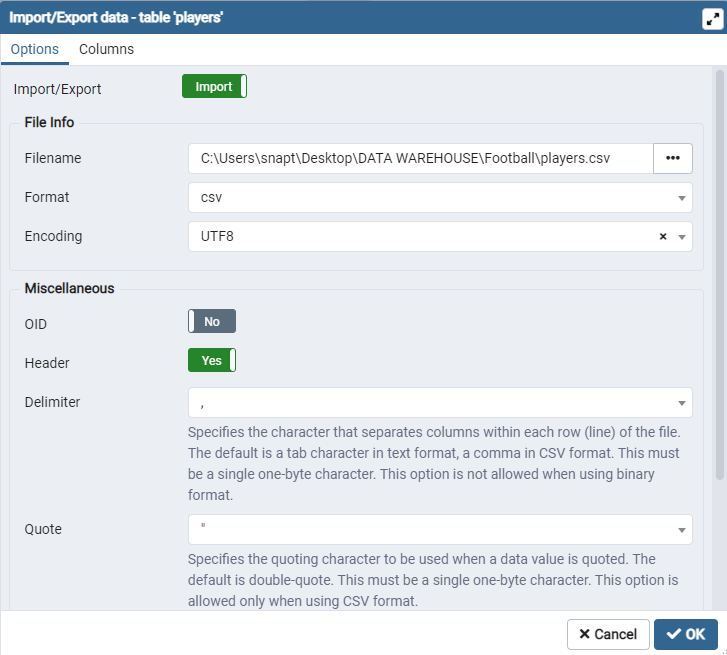
substituteOut int,

leagueID int

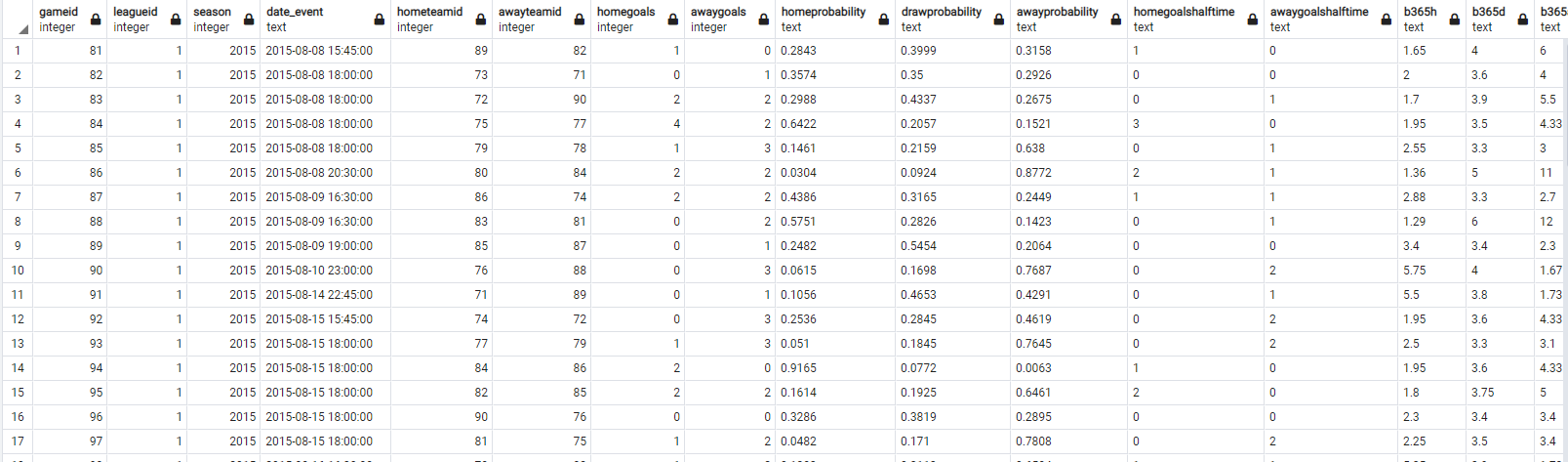
)

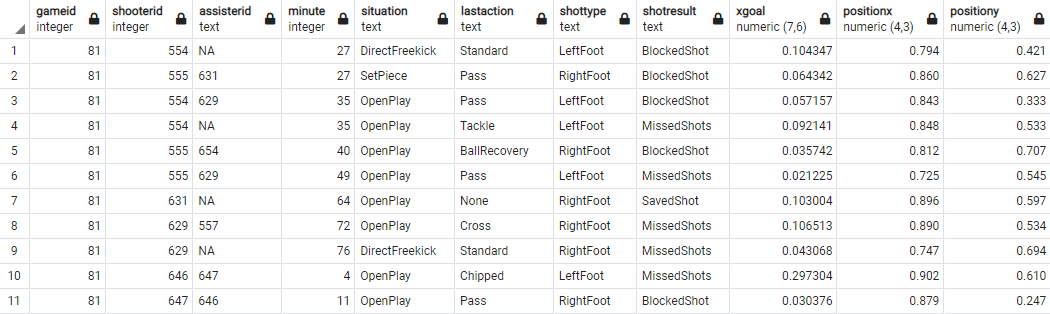
3. Now, we proceed to import the csv files using the import wizard.

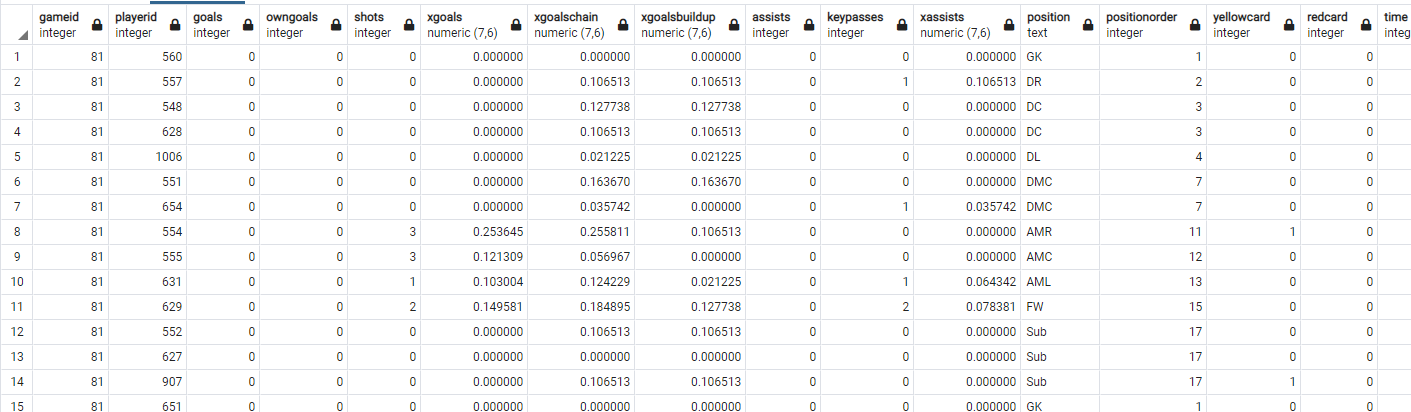


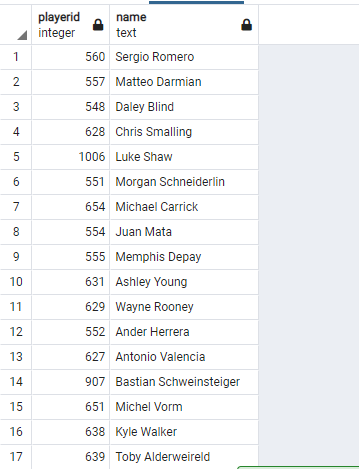


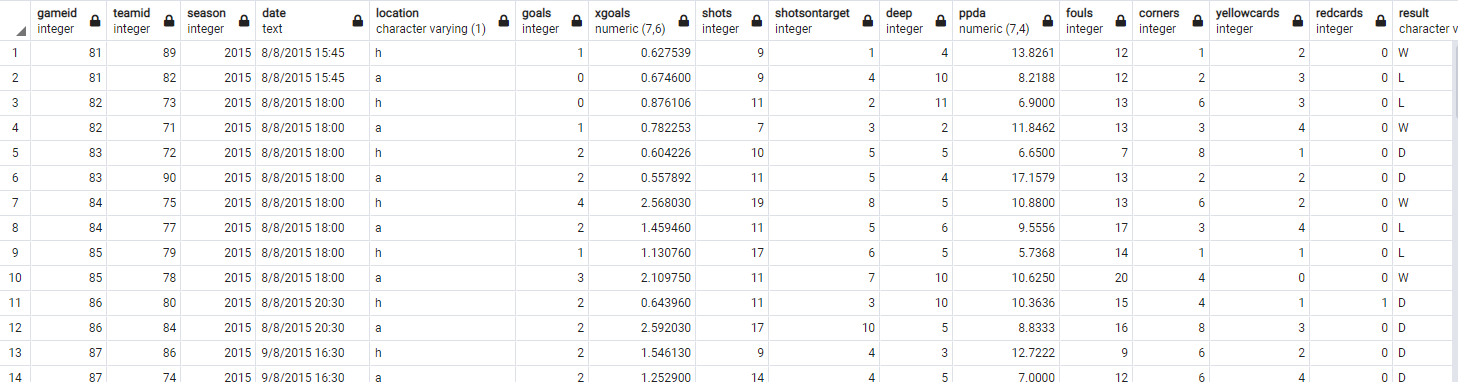
Data Preview

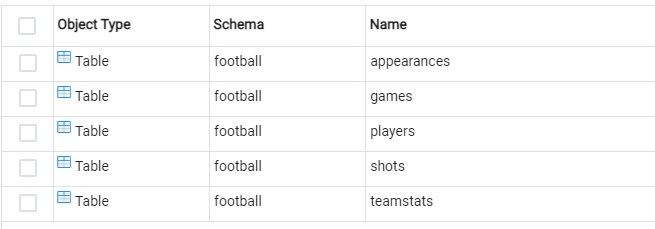












PostgreSQL Tables Structure

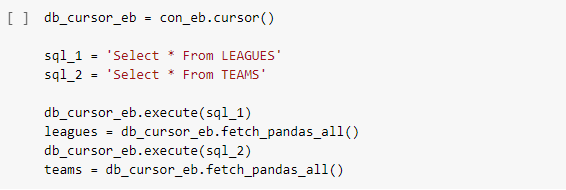
3. For ETL process, we are going to use Python utilising Pandas and Numpy library to conduct data extraction, cleaning and export.

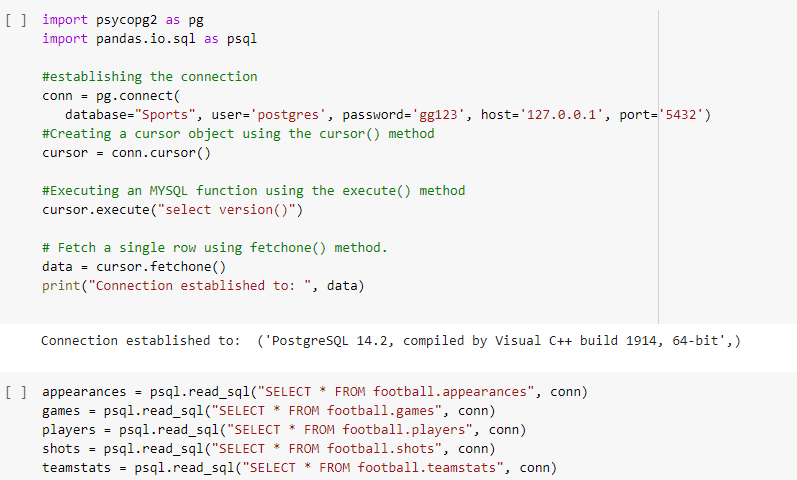
Data Extraction

1. We need to connect to our snowflake account and fetch the schema using snowflake.connector

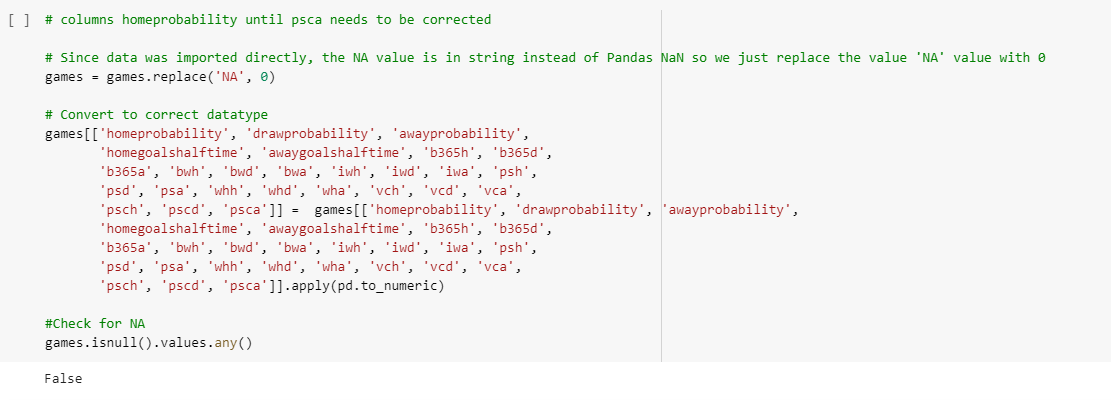


2. We use SQL query to fetch **Leagues** and **Teams** tables from schema and convert to pandas dataframe



3. We fetch tables from postgreSQL using psycopg2 

4. We perform cleaning on all tables that have invalid value

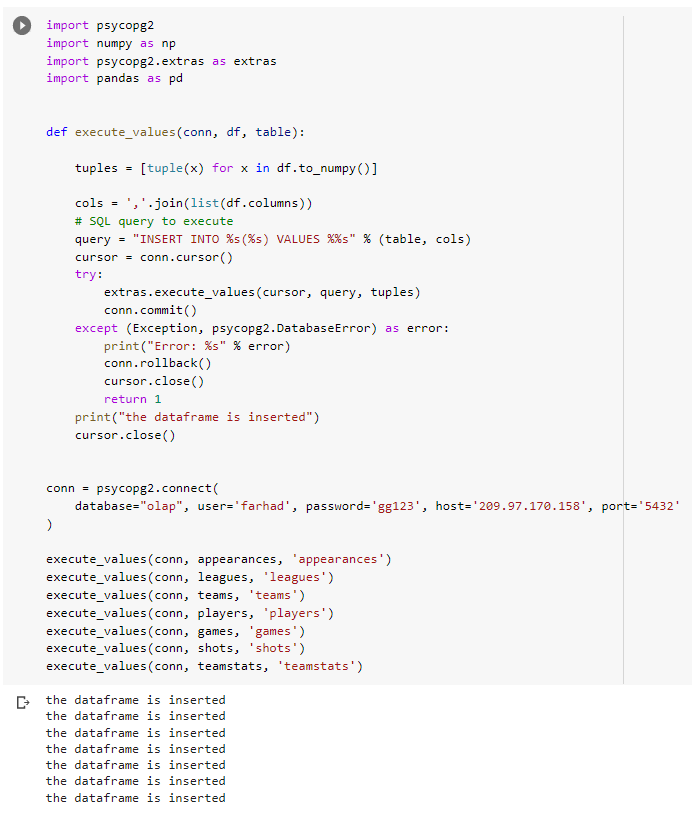


Games table clean

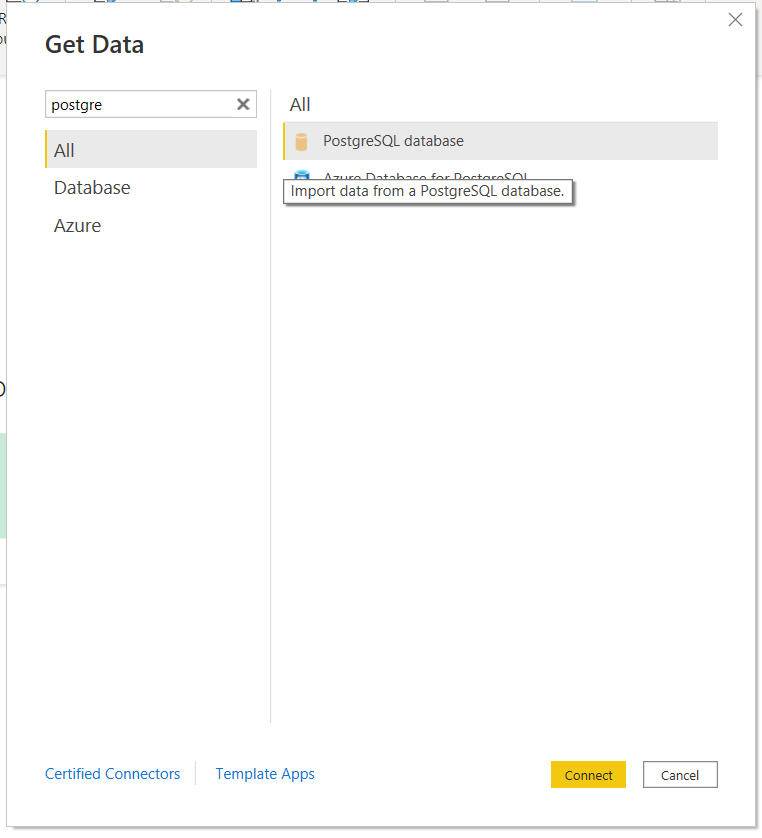


Shots table cleaned

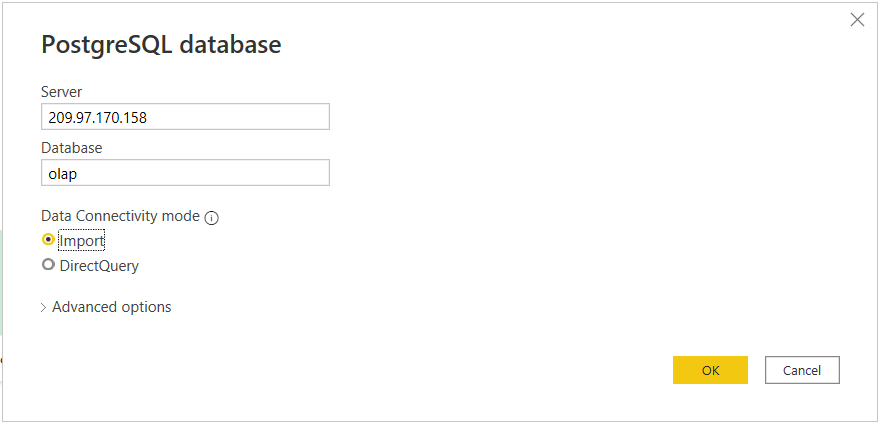
5. After we cleaned our data, it is time to export our data into PostgreSQL. For this, we decided to host our database on cloud using Linux server so that our analytic team can remotely connect to the database.



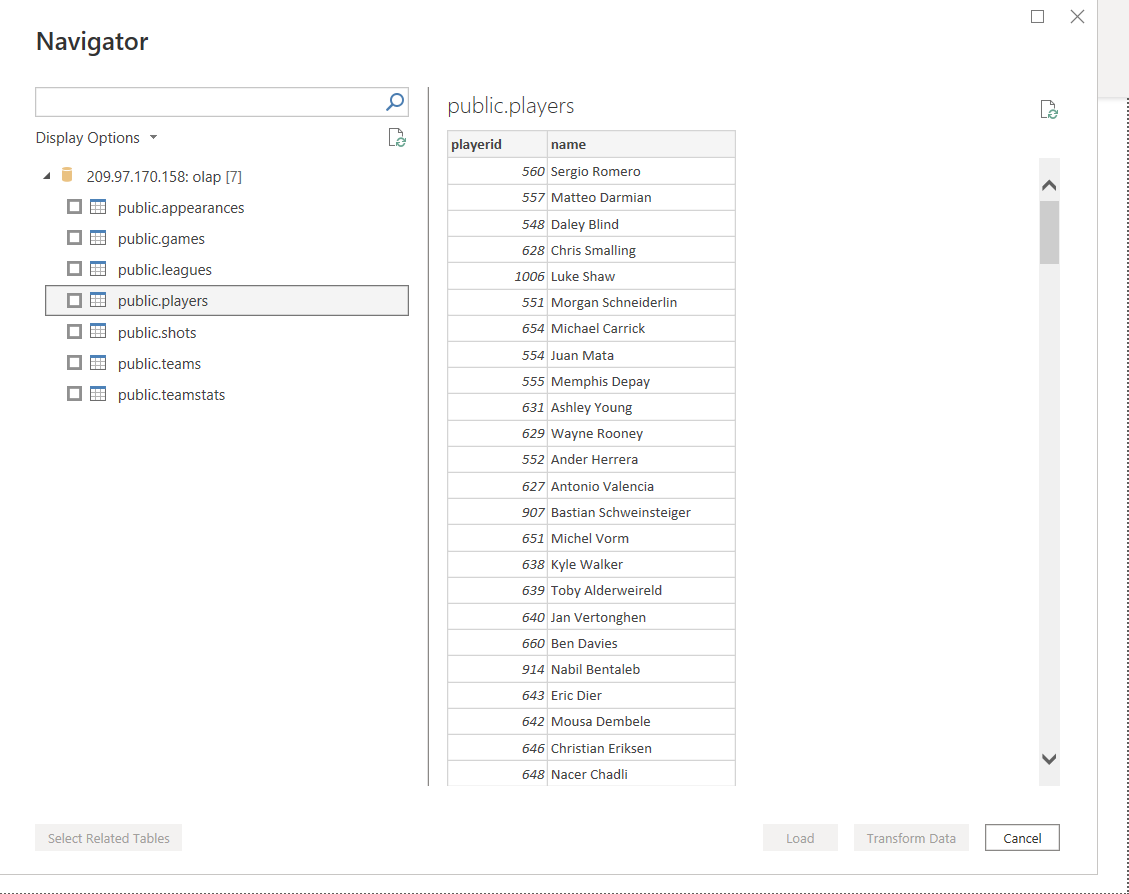
6. In PowerBI, we choose PostgreSQL database.



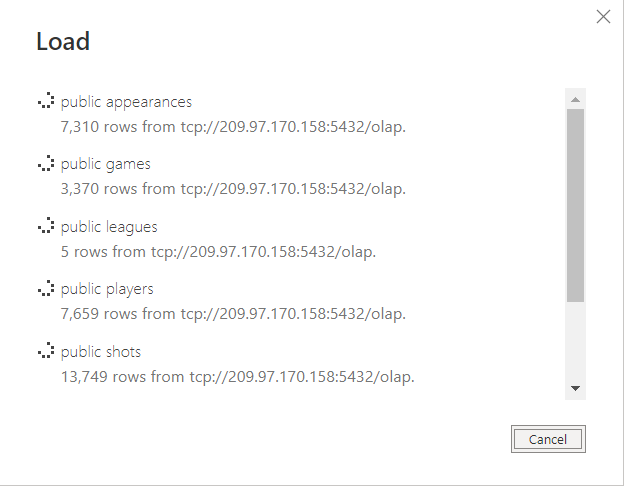
7. We input the IP address and database that we are going to access into.



8. After database successfully connected, we choose our tables that we want to import.



9. Data will be loaded into our PowerBI and ready to conduct analysis.



# 

# **5.0 Result and data analysis**

For OLAP operations, we will be performing rollup, slicing and dicing for early analysis on our data in PostgreSQL (OLAP).

Rollup

SELECT

leagues.name as League, SUM(appearances.goals) as Total\_Goals

FROM

appearances

INNER JOIN leagues ON appearances.leagueid = leagues.leagueid

GROUP BY appearances.leagueid, leagues.name

ORDER BY SUM(appearances.goals) DESC;



From the query above, we can see that from year 2014 until 2020, Seria A league has totals goals of 7255 being the highest compared to Premiere League. This is probably due to the fact that Premiere League (EPL) being the most prestigious league in the world featuring the world-best teams competing in their matches. Hence, it is harder to score more goals on Premiere League compared to Seria A.

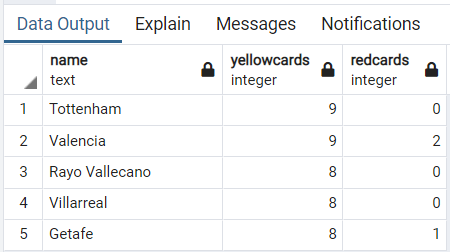
Slicing

SELECT teams.name, teamstats.yellowcards, teamstats.redcards FROM teamstats

INNER JOIN teams ON teamstats.teamid = teams.teamid

ORDER BY yellowcards DESC

LIMIT 5;



From the table above, we can see that Tottenham and Valencia has the highest yellows card in all of leagues combined.

Dicing

SELECT SUM(teamstats.goals) as Goals, teams.name, count(teamstats.result) as Win FROM teamstats

INNER JOIN teams ON teamstats.teamid = teams.teamid

AND teamstats.result = 'W'

GROUP BY teams.name

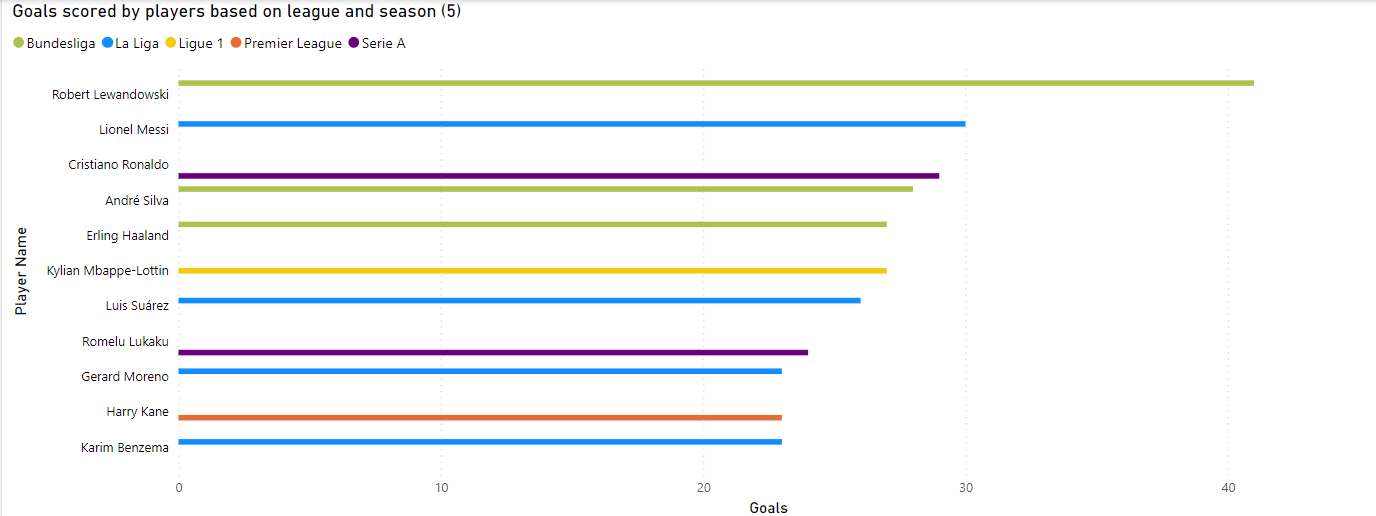
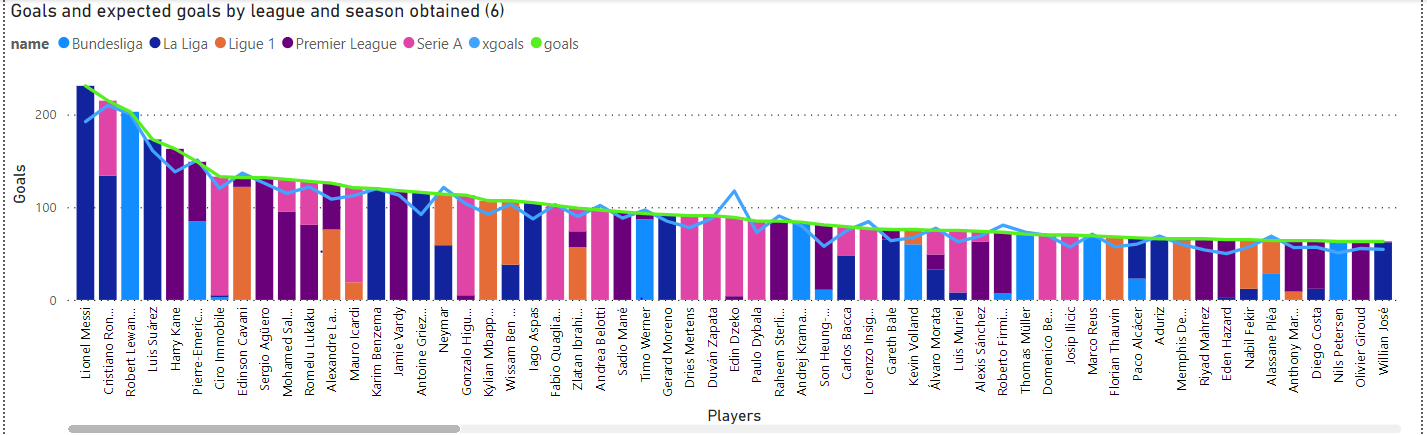
ORDER BY win desc

LIMIT 10;



From the result, we can see that Juventus has won the highest matches in all of the leagues during 2014 until 2020 despite the goals being lower than Barcelona and Paris Saint Germain.

## **5.1 Data Visualization**



**Visualizations analysis :**

1. Visualization (1) : Fouls by season based on league(1) ;-

* The highest total foul by season is in 2014 which is 50763 fouls has been made while the lowest fouls is in season 2019 which is 43,462 fouls.

1. Visualization (2) : Fouls and yellow card by league (2) ;-

* ﻿Sassuolo had the highest fouls (3960) and Athletic Club had the highest yellowcard (708).
* The league with the highest fouls was Serie A and the league with the highest yellowcard was La Liga.
* Most of the leagues were above 2250 fouls and above 345 yellow card.

1. Visualization (3) & (4) : Yellow and red card by season (3) & (4) ;-

* In 2015 season had the highest yellowcard and was 12.73% higher than season 2020, which had the lowest yellowcard at 6781.
* In 2015 season had the highest red card and was 40.25% higher than 2020. Which had the lowest red card at 318.

1. Visualization (5) : Top 10 goals scorer by players based on league and season (5) ;-

* La liga had the highest total goals 102, followed by Bundesliga at 96 and Serie A at 53.
* Lionel Messi from La liga has contributed 11.93% of goals from the entire 2014-2020 season.
* In 2019 season, Bundesliga had the highest average goals at 31, followed by Serie A, Laliga and Premier League.

1. Visualization (6) : Top 10 goals scorer by players based on league and season (6) ;-

* Goals and expected goals diverged the most as Lionel Messi’s goals were 39 higher than expected goals.
* Goals and total expected goals are negatively correlated with each other as it means players with low expectation goals, doesn’t mean it they can’t be the top scorer.
* Edin Dzeko has higher expected goals than goals but somehow he didn’t managed to be the top scorer even his expected goals is higher than his goals.

# **6.0 Conclusion**

Football is always the most famous and popular sport in various European and South American countries that has been played and viewed. As we know that there are many countries involved in this game and there are many datasets for each country that joins the tournament. At the end of this project, we already did the relational model and relationship between the top 5 leagues from 2014 to 2020. We also identify the data warehouse schema to show the relationship between the data. We also use ETL to cover a process of how the data is loaded from the source system to the data warehouse.

However, before completing the project, we face some challenges during our data import. The raw data that we got does not have a proper csv format to directly import into Snowflake. Snowflake will prompt error stating that something is wrong in the format specifications. Eventually, we figure out the correct format of the csv and use the correct settings to import our csv.

In ETL process, we are having hard time trying to import from Snowflake using Python. We had to do some research in the internet to figure out a proper coding to import into our snowflake. In the end, we successfully connected to our Snowflake account and convert it into Pandas dataframe.

In the process of exporting our PostgreSQL (OLAP) into PowerBI, we forgot that our database was hosted in localhost instead of internet making it impossible to pass our data to our data analytics teams. Hence, our leader decided that we host our data warehouse on the cloud by using virtual private server. It took some elbow grease trying to install PostgreSQL on Linux and setup the network access to allow our data analytic to be able to remotely connect. In the end, our analytics team able to connect to our data warehouse remotely using different devices.

# **7.0 References**

“Football - Overview - Tutorialspoint.” *Www.tutorialspoint.com*, www.tutorialspoint.com/football/football\_overview.htm.

Joy, Bernard, and Eric Weil. “Football | History, Rules, & Significant Players.” *Encyclopædia Britannica*, 23 Jan. 2019, www.britannica.com/sports/football-soccer.

Syed Ali Fathima S J, Syed Ali. “Data Analytics in Football Sport to Identify Gaps for the Improvement of Quality Opportunities throughout World-Wide Teams.” *ResearchGate*, https://www.researchgate.net/institution/Kumaraguru\_College\_of\_Technology, Jan. 2018, www.researchgate.net/publication/331100508\_Data\_analytics\_in\_football\_sport\_to\_identify\_gaps\_for\_the\_improvement\_of\_quality\_opportunities\_throughout\_world-wide\_teams.