BigQuery Data Analysis

Overview

In this lab you will learn more fundamentals of sports data science by writing and executing queries to query data stored in BigQuery tables. The emphasis of the lab is to illustrate how the database works and answer some interesting questions related to the following topics in soccer.

- Most total goals scored
- Most attempted passes
- Best penalty success rate

The data used in this lab comes from the following sources:

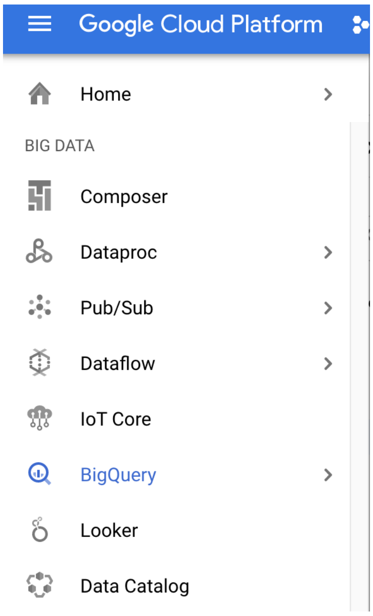
- Pappalardo et al., (2019) **A public data set of spatio-temporal match events in soccer competitions**, Nature Scientific Data 6:236, https://www.nature.com/articles/s41597-019-0247-7
- Pappalardo et al. (2019) **PlayerRank: Data-driven Performance Evaluation and Player Ranking in Soccer via a Machine Learning Approach**. ACM Transactions on Intelligent Systems and Technologies (TIST) 10, 5, Article 59 (September 2019), 27 pages. DOI: https://doi.org/10.1145/3343172

In this lab, you will:

- Query soccer match event data in BigQuery
- Write and execute queries that join information from multiple tables

Open BigQuery

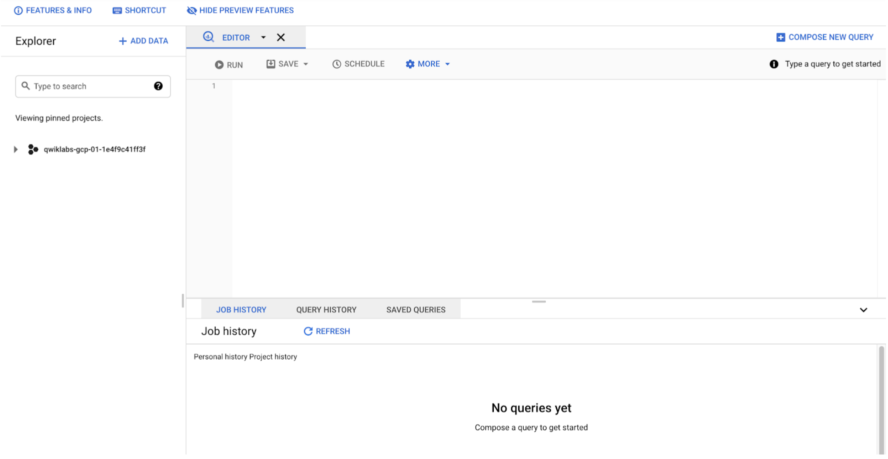
The BigQuery console provides an interface to query tables, including public datasets offered by BigQuery. In the Cloud Console, from the **Navigation menu** select **BigQuery**:



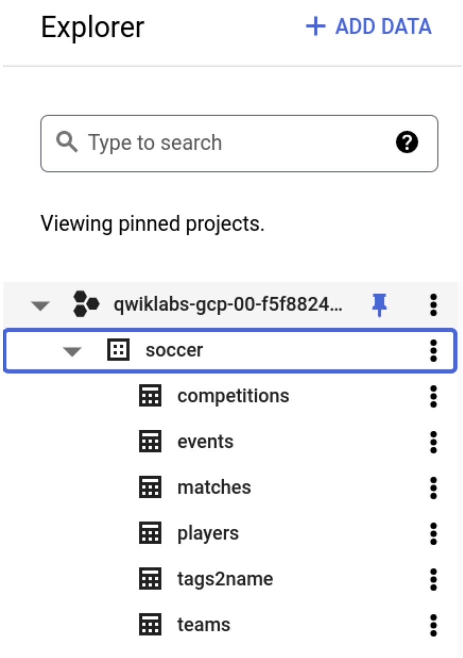
The **Welcome to BigQuery in the Cloud Console** message box opens. This message box provides a link to the quickstart guide and the release notes.

Click **Done**.

The BigQuery console opens.



The process for creating the dataset and tables is taught in the BigQuery Soccer Data Ingestion lab. In this lab the focus is on learning how to query the information. Once the tables are created the display will be similar to this:



Explorer

+ ADD DATA

Type to search

Viewing pinned projects.

qwiklabs-gcp-00-f5f8824...

soccer

competitions

events

matches

players

tags2name

teams

In the next section, begin to learn the fundamentals of creating queries in BigQuery.

Matches with the most goals

In this section, create a query that joins together multiple tables featuring soccer data. Based on the information available, you can perform some basic analytics such as the most total goals scored in a match by both teams (in a specific league).

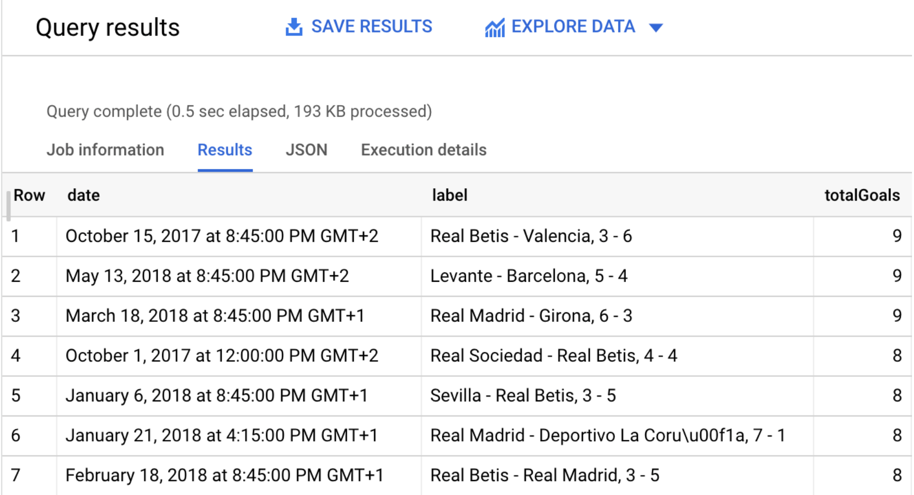
1. In the Query editor, click **Compose new query**.
2. Add the following query to the query **Editor**.

```
SELECT
 date,
 label,
 (team1.score + team2.score) AS totalGoals
FROM
 `soccer.matches` Matches
LEFT JOIN
 `soccer.competitions` Competitions ON
   Matches.competitionId = Competitions.wyId
WHERE
 status = 'Played' AND
 Competitions.name = 'Spanish first division'
ORDER BY
 totalGoals DESC, date DESC
```

Here is what the query will do:

- joins the **matches** table (which has final scores) with the **competitions** table.
- filter down to "Spanish first division" matches only.
- order by a calculated field that represents total goals in a match.

3. Click **Run**. The results are displayed below the query window.



Query complete (0.5 sec elapsed, 193 KB processed)

Row	date	label	totalGoals
1	October 15, 2017 at 8:45:00 PM GMT+2	Real Betis - Valencia, 3 - 6	9
2	May 13, 2018 at 8:45:00 PM GMT+2	Levante - Barcelona, 5 - 4	9
3	March 18, 2018 at 8:45:00 PM GMT+1	Real Madrid - Girona, 6 - 3	9
4	October 1, 2017 at 12:00:00 PM GMT+2	Real Sociedad - Real Betis, 4 - 4	8
5	January 6, 2018 at 8:45:00 PM GMT+1	Sevilla - Real Betis, 3 - 5	8
6	January 21, 2018 at 4:15:00 PM GMT+1	Real Madrid - Deportivo La Coru\u00f1a, 7 - 1	8
7	February 18, 2018 at 8:45:00 PM GMT+1	Real Betis - Real Madrid, 3 - 5	8

In this section BigQuery was used to illustrate how to define a query that shows soccer information. The query creates a filter that displays specific information about matches from a specific league and enables the information to be categorized by a defined field.

Players with the most passes

In this section, create a query that joins together multiple tables featuring soccer data. Based on the information available, you can perform some basic analytics such as total passes by players.

1. In the Query editor, click **Compose new query**.
2. Add the following query into the query **Editor**.

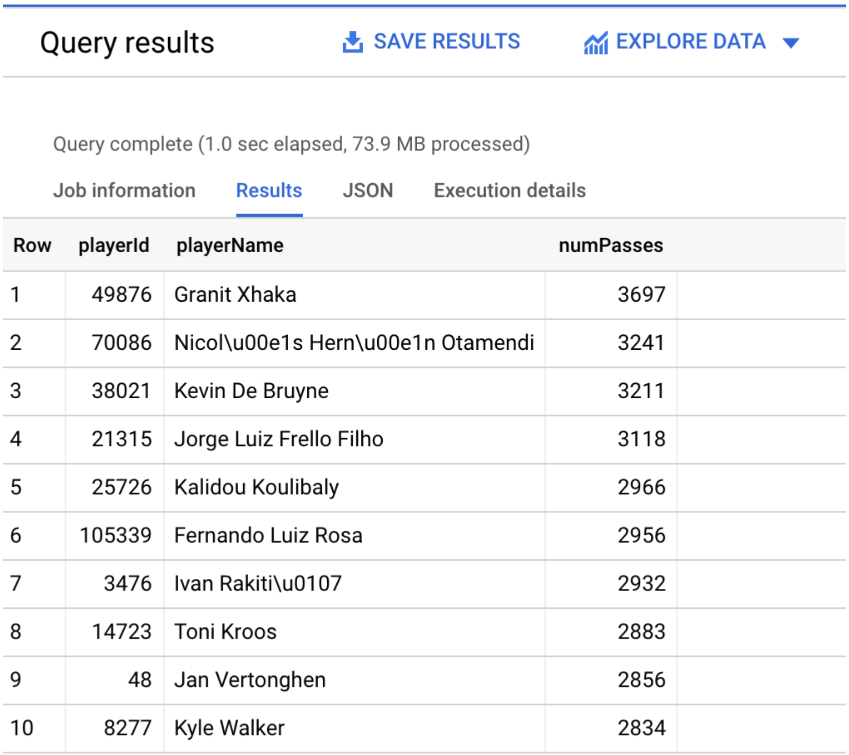
This query:

- joins the **events** table (which has a record of every pass) with the **players** table to get player names from their IDs
- groups by player
- counts the number of passes for each one
- orders by the players with the most passes

```
SELECT
```

```
 playerId,
 (Players.firstName || ' ' || Players.lastName) AS playerName,
 COUNT(id) AS numPasses
FROM
 `soccer.events` Events
LEFT JOIN
 `soccer.players` Players ON
   Events.playerId = Players.wyId
WHERE
 eventName = 'Pass'
GROUP BY
 playerId, playerName
ORDER BY
 numPasses DESC
LIMIT 10
```

3. Click **Run**. The results are displayed below the query window.



Query complete (1.0 sec elapsed, 73.9 MB processed)

Row	playerId	playerName	numPasses
1	49876	Granit Xhaka	3697
2	70086	Nicol\u00e1s Hern\u00e1n Otamendi	3241
3	38021	Kevin De Bruyne	3211
4	21315	Jorge Luiz Frello Filho	3118
5	25726	Kalidou Koulibaly	2966
6	105339	Fernando Luiz Rosa	2956
7	3476	Ivan Rakiti\u0107	2932
8	14723	Toni Kroos	2883
9	48	Jan Vertonghen	2856
10	8277	Kyle Walker	2834

*In this section BigQuery was used to illustrate how to define a query that shows player information. The query creates a join that displays specific information about a **playerId** and enables the information to be categorized by a defined field. In the next section learn more about the existing dataset and explore how it can be used to determine the penalty kick success rate of players.*

Determine penalty kick success rate

In this section, create a query that joins together multiple tables featuring soccer data. Based on the information available, you can perform some analytics such as the success rate on penalty kicks by each player.

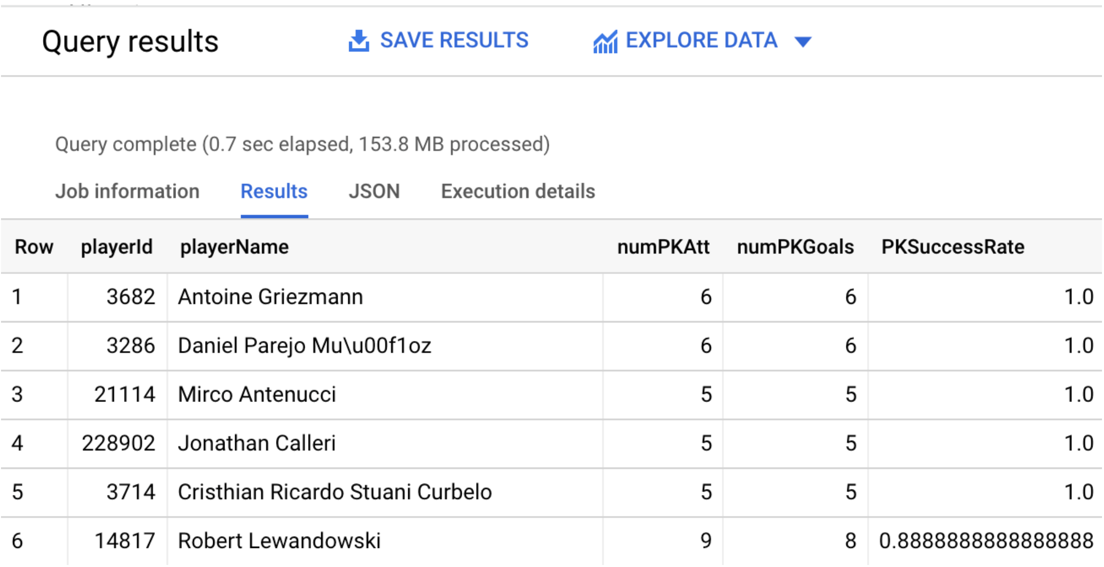
1. In the Query editor, click **Compose new query**.
2. Copy and paste the following query into the query **Editor**:

```
SELECT
 playerId,
 (Players.firstName || ' ' || Players.lastName) AS playerName,
 COUNT(id) AS numPKAtt,
 SUM(IF(101 IN UNNEST(tags.id), 1, 0)) AS numPKGoals,
 SAFE_DIVIDE(
   SUM(IF(101 IN UNNEST(tags.id), 1, 0)),
   COUNT(id)
   ) AS PKSuccessRate
FROM
 `soccer.events` Events
LEFT JOIN
 `soccer.players` Players ON
   Events.playerId = Players.wyId
WHERE
 eventName = 'Free Kick' AND
 subEventName = 'Penalty'
GROUP BY
 playerId, playerName
HAVING
 numPkAtt >= 5
ORDER BY
 PKSuccessRate DESC, numPKAtt DESC
```

The query aggregates the number of penalty kick attempts and successful ones by player and filters to those with at least 5 penalty kick attempts before ordering by success rate.

The above query joins the ***events*** *table, in this case filtered to only penalty kicks, with the* ***players*** *table to get player names from their IDs.*
The tags field in the events table uses BigQuery's array functionality (allowing more than 1 tag to be stored per event), so it must be unnested to determine if the kick was good or not (one can confirm that tag 101 represents a goal using the ***tags2name*** *table).*

3. Click **Run**. The results are displayed below the query window.



Query complete (0.7 sec elapsed, 153.8 MB processed)

Row	playerId	playerName	numPKAtt	numPKGoals	PKSuccessRate
1	3682	Antoine Griezmann	6	6	1.0
2	3286	Daniel Parejo Mu\u00f1oz	6	6	1.0
3	21114	Mirco Antenucci	5	5	1.0
4	228902	Jonathan Calleri	5	5	1.0
5	3714	Cristhian Ricardo Stuani Curbelo	5	5	1.0
6	14817	Robert Lewandowski	9	8	0.8888888888888888

In this section BigQuery was used to illustrate how to define a query that shows player information relating to penalty kicks. The query creates a join that displays specific information about a ***playerId*** *and enables more detailed information to be displayed.*