

For this project, I have created a database which records data of European soccer players and their teams and leagues they play in, the countries the games are being played in, and attributes of both a player and a team.

This database project idea came from the website projectpro.io, where I was re-directed to a link to a website called Kaggle and able to download a dataset. That dataset had a file in SQLite format, which contained the following tables:

- country: a table that lists out countries where soccer games are played in.
- league: a table that lists out the leagues that games are played under
- match: a table that lists out performance metrics for all of the players
- player: a table that lists out a player's name and its profile
- player_attributes: a table that lists out
- team: a table that lists out teams in the database
- team_attributes: a table that lists out how well each team does in certain attributes of soccer, with the ones on in

The dataset contained more than 25,000 matches, more than 10,000 players, 11 European Countries and their leagues, players that played in the years between 2008-2016., and player and team attributes that came from the EA Sports FIFA video games.

The structure of the tables looks like this:

Country

Attribute	Type	Description
id	integer	table identifier
name	text	country name

League

Attribute	Type	Description
id	integer	table identifier
Name	text	name of league

Match

Attribute	Type	Description
id	integer	table identifier
country_id	integer	foreign key for country table
league_id	integer	foreign key for league table
season	text	year of season

stage		
date		
home_team_api_id	integer	foreign key for the home team
away_team_api_id	integer	foreign key for the away team
home_team_goal	integer	number of goals for the home team
away_team_goal	integer	number of goals for the away team
home_player_x1	integer	integer position of the home player x-axis
away_player_x1	integer	integer position of the away player x-axis
home_player_y1	integer	integer position of the home player y-axis
.....
away_player_y1	integer	integer position of the away player y-axis
.....
home_player_1	integer	foreign key for the player table represents the first player of the home team
.....
away_player_1	integer	foreign key for the player table represents the first player of the away team
.....
shot_on	text	XML text to represent the id of each player who made a shot on the target
shot_off	text	XML text to represent the id of each player who made a shot off the target
goal	text	XML text to represent the ids of the players who scored goals and how many
foulcommit	text	XML text to represent the ids of the players who commit fouls
card	text	XML text to represent the ids of the players who took a [yellow-red] card
corner	text	XML text to represent the ids of the players who played corners
possession	text	XML text to represent home and away possession
B365H	numeric	Bookmaker or betting site odds in decimal format
B365D	numeric	Bookmaker or betting site odds in decimal format

Player

Attribute	Type	Description
id	integer	table identifier
player_api_id	integer	unique id
player_fifa_api_id	integer	unique id
player_name	text	name of the player
birthday	text	player's birthday
weight	integer	weight of the player
height	integer	height of the player

Player_Attributes

Attribute	Type	Description
id	integer	table identifier
player_api_id	integer	foreign key for player table
player_fifa_api_id	integer	foreign key for player table
date	text	date of the update
overall_rating	integer	fifa rating
potential	integer	fifa rating
preferred_foot	integer	fifa rating

Team

Attribute	Type	Description
id	integer	table identifier
team_api_id	integer	unique id
team_fifa_api_id	integer	unique id
team_long_name	text	team long name
team_short_name	text	team short name

Team_Attributes

Attribute	Type	Description
id	integer	table identifier
team_api_id	integer	foreign key for team table
team_fifa_api_id	integer	foreign key for team table
date	text	date of session
buildupplayspeed	integer	rank from 1 to 100
buildupplayspeedclass	text	classification for build up speed
buildupplaydribbling	integer	rank from 1 to 100
buildupplaydribblingclass	text	classification for build up dribbling

I first originally decided to create the schema (another word for database in MySQL) from scratch and construct the tables manually on an application provided by MySQL called Workbench. In Workbench, I was able to create the tables and include columns and their datatypes. Plus, I was able to select options for those columns, like if a column could be used as a primary key which is a field in a table that identifies each row in a unique manner in a database table and must contain unique values, if the column should not be allowed to have values that are deemed NULL (meaning a column that has no value), if a column can be unsigned (where values can only be positive numbers), or if a column is unique (where there cannot be duplicate values).

I was able to construct the tables with the right columns, according to the SQLite database. And then add in some data from using the SQLite file and importing that data into the MySQL server by using an application I found called SQLite to MySQL. I added data into the tables in (except for the table `sqlite_sequence` since that just listed the number of rows each table had) and was able to add all the rows to each table.

For analyzing the data, I did it by using both MySQL Workbench and Windows Command Prompt. In the Command Prompt, I signed in with MySQL credentials and was able to get into MySQL where I could interact with the databases and their tables.

This screenshot shows the tables listed in the 'soccer' schema.

```
mysql> SHOW TABLES FROM soccer;
+-----+
| Tables_in_soccer |
+-----+
| country          |
| league           |
| match            |
| player           |
| player_attributes|
| team             |
| team_attributes  |
+-----+
7 rows in set (0.00 sec)
```

This screenshot lists the data from the table 'country' from the soccer schema.

```
mysql> SELECT * FROM soccer.country;
```

id	name
1	Belgium
1729	England
4769	France
7809	Germany
10257	Italy
13274	Netherlands
15722	Poland
17642	Portugal
19694	Scotland
21518	Spain
24558	Switzerland

```
11 rows in set (0.02 sec)
```

This screenshot lists the data from the table 'league' from the soccer schema.

```
mysql> SELECT * FROM soccer.league;
```

id	country_id	name
1	1	Belgium Jupiler League
1729	1729	England Premier League
4769	4769	France Ligue 1
7809	7809	Germany 1. Bundesliga
10257	10257	Italy Serie A
13274	13274	Netherlands Eredivisie
15722	15722	Poland Ekstraklasa
17642	17642	Portugal Liga ZON Sagres
19694	19694	Scotland Premier League
21518	21518	Spain LIGA BBVA
24558	24558	Switzerland Super League

```
11 rows in set (0.04 sec)
```

This screenshot lists some data from the table 'team' from the soccer schema.

```
mysql> SELECT * FROM soccer.team;
```

id	team_api_id	team_fifa_api_id	team_long_name	team_short_name
1	9987	673	KRC Genk	GEN
2	9993	675	Beerschot AC	BAC
3	10000	15005	SV Zulte-Waregem	ZUL
4	9994	2007	Sporting Lokeren	LOK
5	9984	1750	KSV Cercle Brugge	CEB
6	8635	229	RSC Anderlecht	AND
7	9991	674	KAA Gent	GEN
8	9998	1747	RAEC Mons	MON
10	9985	232	Standard de Liège	STL
11	8203	110724	KV Mechelen	MEC
12	8342	231	Club Brugge KV	CLB
13	9999	546	KSV Roeselare	ROS
14	8571	100081	KV Kortrijk	KOR
16	9996	111560	Royal Excel Mouscron	MOU
17	10001	681	KVC Westerlo	WES
18	9986	670	Sporting Charleroi	CHA
614	9997	680	Sint-Truidense VV	STT
1034	9989	239	Lierse SK	LIE
1042	6351	2013	KAS Eupen	EUP
1513	1773	100087	Oud-Heverlee Leuven	O-H
2004	8475	110913	Waasland-Beveren	WAA
2476	8573	682	KV Oostende	OOS
2510	274581	111560	Royal Excel Mouscron	MOP
3457	10260	11	Manchester United	MUN
3458	10261	13	Newcastle United	NEW
3459	9825	1	Arsenal	ARS
3460	8659	109	West Bromwich Albion	WBA
3461	8472	106	Sunderland	SUN
3462	8650	9	Liverpool	LIV
3463	8654	19	West Ham United	WHU
3464	8528	1917	Wigan Athletic	WIG
3465	10252	2	Aston Villa	AVL
3466	8456	10	Manchester City	MCI
3467	8668	7	Everton	EVE
3468	8655	3	Blackburn Rovers	BLB

This screenshot lists the table 'league' in alphabetical order according to the column 'name' (I know all of the data is already set in order but even if the id and country_id values weren't in numerical order, the table would be ordered to list the values in 'league' in alphabetical order according to the column 'name')

```
mysql> SELECT * FROM league ORDER BY name;
```

id	country_id	name
1	1	Belgium Jupiler League
1729	1729	England Premier League
4769	4769	France Ligue 1
7809	7809	Germany 1. Bundesliga
10257	10257	Italy Serie A
13274	13274	Netherlands Eredivisie
15722	15722	Poland Ekstraklasa
17642	17642	Portugal Liga ZON Sagres
19694	19694	Scotland Premier League
21518	21518	Spain LIGA BBVA
24558	24558	Switzerland Super League

11 rows in set (0.00 sec)

This screenshot lists out how many rows the soccer schema table 'player' has.

```
mysql> SELECT COUNT(*) FROM soccer.player;
```

COUNT(*)
11059

1 row in set (0.01 sec)

This screenshot lists out how many rows the soccer schema table 'player_attributes' has.

```
mysql> SELECT COUNT(*) FROM soccer.player_attributes;
+-----+
| COUNT(*) |
+-----+
|    183978 |
+-----+
1 row in set (0.13 sec)
```

This screenshot lists out how many rows the soccer schema table 'match' has.

```
mysql> SELECT COUNT(*) FROM soccer.match;
+-----+
| COUNT(*) |
+-----+
|    25979 |
+-----+
1 row in set (0.12 sec)
```

This screenshot lists out how many rows the soccer schema table 'team_attributes' has.

```
mysql> SELECT COUNT(*) FROM soccer.team_attributes;
+-----+
| COUNT(*) |
+-----+
|    1458 |
+-----+
1 row in set (0.01 sec)
```

This screenshot find the name "England Premier League" from the column name in the table 'league'.

```
mysql> SELECT * FROM soccer.league WHERE name = 'England Premier League';
+-----+-----+-----+
| id | country_id | name |
+-----+-----+-----+
| 1729 | 1729 | England Premier League |
+-----+-----+-----+
1 row in set (0.00 sec)
```

This screenshot lists out several soccer team's name and their profile from the 'team' table.

```
mysql> SELECT * FROM soccer.team WHERE team_long_name = 'Manchester United';
+-----+-----+-----+-----+-----+
| id | team_api_id | team_fifa_api_id | team_long_name | team_short_name |
+-----+-----+-----+-----+-----+
| 3457 | 10260 | 11 | Manchester United | MUN |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM soccer.team WHERE team_long_name = 'Liverpool';
+-----+-----+-----+-----+-----+
| id | team_api_id | team_fifa_api_id | team_long_name | team_short_name |
+-----+-----+-----+-----+-----+
| 3462 | 8650 | 9 | Liverpool | LIV |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM soccer.team WHERE team_long_name = 'Paris Saint-Germain';
+-----+-----+-----+-----+-----+
| id | team_api_id | team_fifa_api_id | team_long_name | team_short_name |
+-----+-----+-----+-----+-----+
| 9548 | 9847 | 73 | Paris Saint-Germain | PSG |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM soccer.team WHERE team_long_name = 'Aston Villa';
+-----+-----+-----+-----+-----+
| id | team_api_id | team_fifa_api_id | team_long_name | team_short_name |
+-----+-----+-----+-----+-----+
| 3465 | 10252 | 2 | Aston Villa | AVL |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> SELECT * FROM soccer.team WHERE team_long_name = 'Vancouver Whitecaps';
Empty set (0.00 sec)
```

This screenshot lists out several soccer player's name and their profile from the soccer.player table.

```
mysql> SELECT * FROM soccer.player WHERE player_name = 'Lionel Messi';
+-----+-----+-----+-----+-----+-----+-----+
| id | player_api_id | player_name | player_fifa_api_id | birthday | height | weight |
+-----+-----+-----+-----+-----+-----+-----+
| 6176 | 30981 | Lionel Messi | 158023 | 1987-06-24 00:00:00 | 170.18 | 159 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

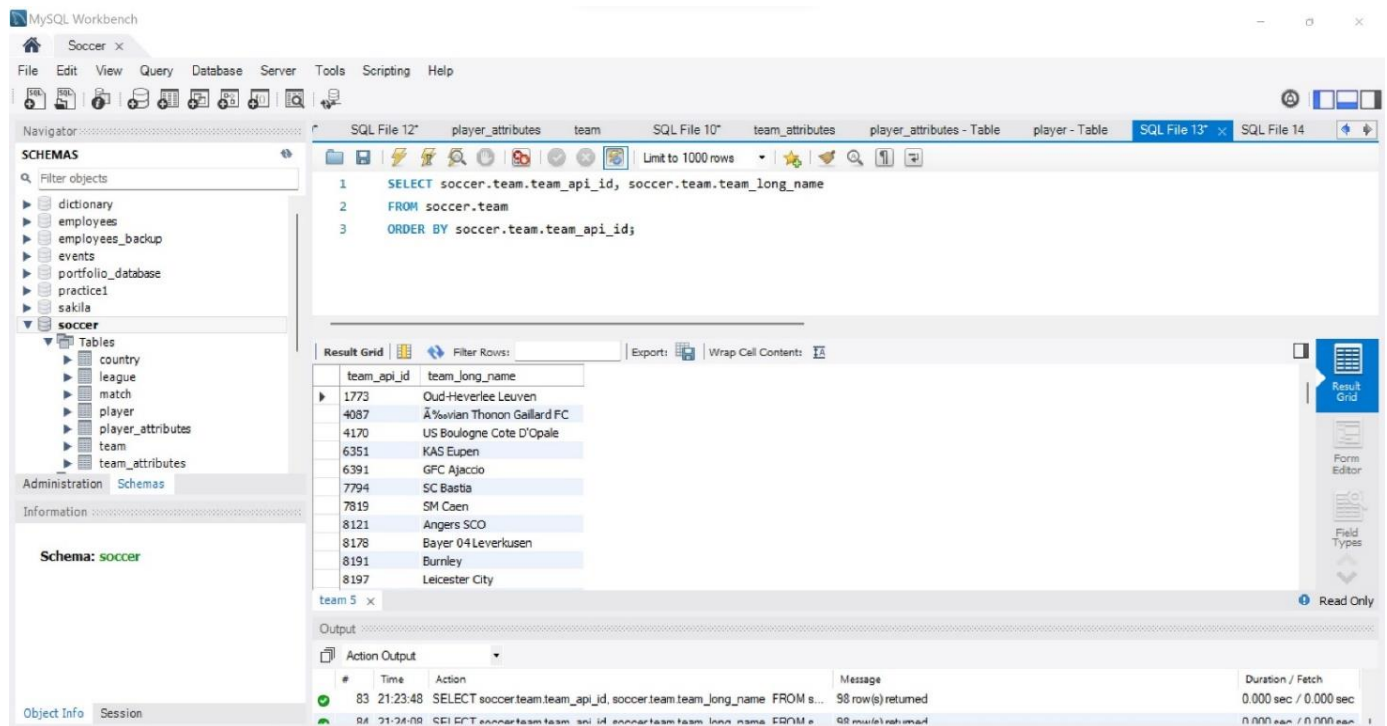
mysql> SELECT * FROM soccer.player WHERE player_name = 'Cristiano Ronaldo';
+-----+-----+-----+-----+-----+-----+-----+
| id | player_api_id | player_name | player_fifa_api_id | birthday | height | weight |
+-----+-----+-----+-----+-----+-----+-----+
| 1995 | 30893 | Cristiano Ronaldo | 20801 | 1985-02-05 00:00:00 | 185.42 | 176 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

mysql> SELECT * FROM soccer.player WHERE player_name = 'Fernando Torres';
+-----+-----+-----+-----+-----+-----+-----+
| id | player_api_id | player_name | player_fifa_api_id | birthday | height | weight |
+-----+-----+-----+-----+-----+-----+-----+
| 3335 | 30853 | Fernando Torres | 49369 | 1984-03-20 00:00:00 | 185.42 | 172 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

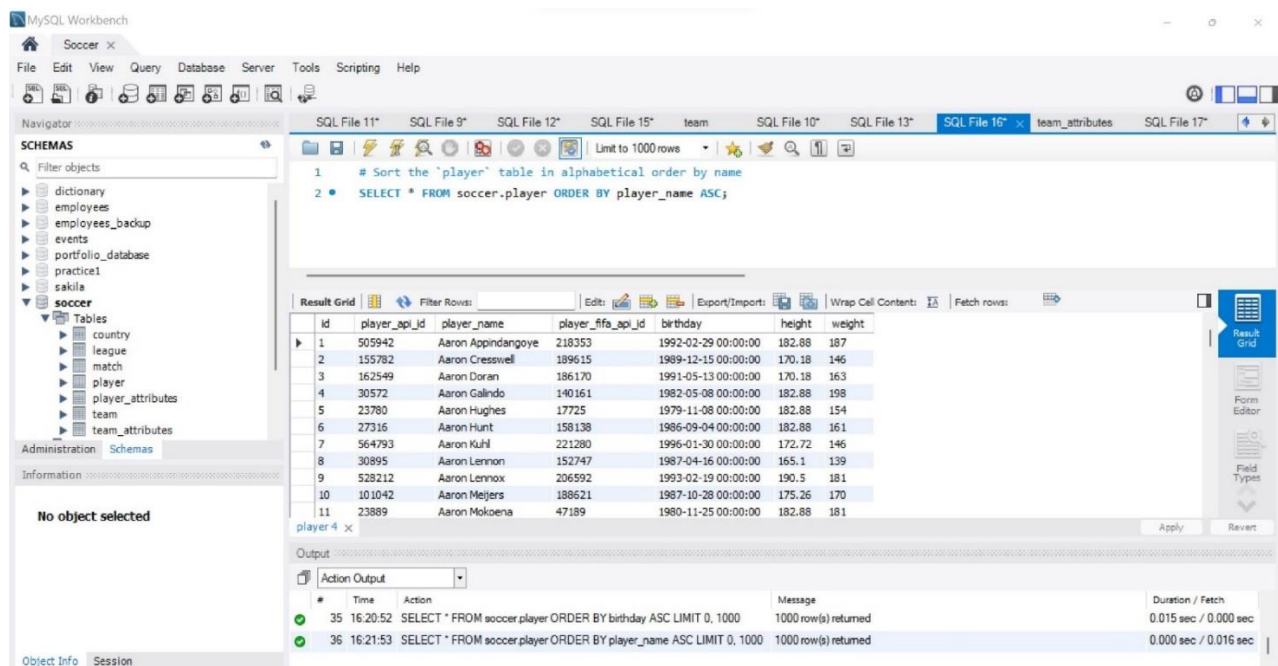
mysql> SELECT * FROM soccer.player WHERE player_name = 'David Villa';
+-----+-----+-----+-----+-----+-----+-----+
| id | player_api_id | player_name | player_fifa_api_id | birthday | height | weight |
+-----+-----+-----+-----+-----+-----+-----+
| 2430 | 30909 | David Villa | 113422 | 1981-12-03 00:00:00 | 175.26 | 152 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)

mysql> SELECT * FROM soccer.player WHERE player_name = 'David Beckham';
+-----+-----+-----+-----+-----+-----+-----+
| id | player_api_id | player_name | player_fifa_api_id | birthday | height | weight |
+-----+-----+-----+-----+-----+-----+-----+
| 2324 | 30612 | David Beckham | 250 | 1975-05-02 00:00:00 | 182.88 | 165 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.01 sec)
```

This screenshot lists out two columns, team_api_id and team_long_name, from the table 'team' and the results are ordered by the column team_api_id.



This screenshot lists the 'player' table sorted in alphabetical order by name



This table lists the 'player' table being sorted by the 10 oldest players in that table.

MySQL Workbench

Soccer x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dictionary
employees
employees_backup
events
portfolio_database
practice1
sakila
soccer

Tables

country
league
match
player
player_attributes
team
team_attributes

Administration Schemas

Information

No object selected

Object Info Session

SQL File 11* SQL File 9* SQL File 12* SQL File 15* team SQL File 10* SQL File 13* SQL File 16* x team_attributes SQL File 17*

Limit to 1000 rows

1 # Sort the 'player' table by finding the oldest 10 players
2 • SELECT * FROM soccer.player ORDER BY birthday ASC LIMIT 10;

Result Grid

	id	player_api_id	player_name	player_fifa_api_id	birthday	height	weight
▶	290	39425	Alberto Fontana	2431	1967-01-23 00:00:00	185.42	161
	8296	26099	Paolo Maldini	1109	1968-06-26 00:00:00	185.42	187
	9007	26576	Rob van Dijk	15708	1969-01-15 00:00:00	187.96	183
	6245	27666	Luca Bucci	12514	1969-03-13 00:00:00	180.34	174
	2495	23605	Dean Windass	50474	1969-04-01 00:00:00	177.8	183
	3456	41243	Francesco Antonioli	137129	1969-09-14 00:00:00	187.96	181
	7400	27346	Michael Tarnat	722	1969-10-27 00:00:00	185.42	192
	4832	30648	Jens Lehmann	805	1969-11-10 00:00:00	190.5	192
	4126	26003	Hans Vonk	7947	1970-01-30 00:00:00	195.58	183
	2432	23998	David Weir	50637	1970-05-10 00:00:00	190.5	190

player 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
36	16:21:53	SELECT * FROM soccer.player ORDER BY player_name ASC LIMIT 0, 1000	1000 row(s) returned	0.000 sec / 0.016 sec
37	16:23:59	SELECT * FROM soccer.player ORDER BY birthday ASC LIMIT 10	10 row(s) returned	0.000 sec / 0.000 sec

This table lists the 'player' table being sorted by the 10 youngest players in that table.

MySQL Workbench

Soccer x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

dictionary
employees
employees_backup
events
portfolio_database
practice1
sakila
soccer

Tables

country
league
match
player
player_attributes
team
team_attributes

Administration Schemas

Information

No object selected

Object Info Session

SQL File 11* SQL File 9* SQL File 12* SQL File 15* team SQL File 10* SQL File 13* SQL File 16* x team_attributes SQL File 17*

Limit to 1000 rows

1 # Sort the 'player' table by finding the youngest 10 players
2 • SELECT * FROM soccer.player ORDER BY birthday DESC LIMIT 10;

Result Grid

	id	player_api_id	player_name	player_fifa_api_id	birthday	height	weight
▶	5183	682552	Jonathan Leko	231408	1999-04-24 00:00:00	182.88	141
	3827	618878	Gianluigi Donnarumma	230621	1999-02-25 00:00:00	195.58	198
	272	698273	Alban Lafont	231691	1999-01-23 00:00:00	193.04	170
	5954	701154	Kylan Mbappe Lottin	231747	1998-12-20 00:00:00	177.8	148
	8840	575789	Reece Oxford	225908	1998-12-16 00:00:00	190.5	157
	4143	671518	Harry Souttar	229901	1998-10-22 00:00:00	198.12	176
	1771	688295	Christian Pulisic	227796	1998-09-18 00:00:00	172.72	152
	5159	710807	Jonathan Montiel	231873	1998-09-03 00:00:00	172.72	148
	10372	742405	Tom Davies	230005	1998-06-30 00:00:00	175.26	172
	9212	699362	Ronael Pierre-Gabriel	231685	1998-06-13 00:00:00	175.26	150

player 6 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
37	16:23:59	SELECT * FROM soccer.player ORDER BY birthday ASC LIMIT 10	10 row(s) returned	0.000 sec / 0.000 sec
38	16:24:20	SELECT * FROM soccer.player ORDER BY birthday DESC LIMIT 10	10 row(s) returned	0.000 sec / 0.000 sec

This table lists the 'player' table being sorted by the 10 tallest players in that table.

The screenshot shows the MySQL Workbench interface with a SQL query in the editor:

```
1 # Sort the `player` table by finding the tallest 10 players
2 SELECT * FROM soccer.player GROUP BY height ORDER BY height DESC LIMIT 10;
```

The Result Grid displays the following data:

id	player_api_id	player_name	player_fifa_api_id	birthday	height	weight
5908	148325	Kristof van Hout	185306	1987-02-09 00:00:00	208.28	243
1301	150209	Bogdan Milic	189967	1987-11-24 00:00:00	203.2	216
35	564712	Abdoul Ba	225050	1994-02-08 00:00:00	200.66	212
39	191784	Abdoulaye Ba	204826	1991-01-01 00:00:00	198.12	174
217	297556	Ahmed Hegazy	210648	1991-01-25 00:00:00	195.58	183
70	27277	Abou Diaby	163423	1986-05-11 00:00:00	193.04	168
9	528212	Aaron Lennox	206592	1993-02-19 00:00:00	190.5	181
13	163222	Aaron Muirhead	213568	1990-08-30 00:00:00	187.96	168
22	245653	Abdelfettah Boukhriess	202425	1986-10-22 00:00:00	185.42	161
1	505942	Aaron Appindangoye	218353	1992-02-29 00:00:00	182.88	187

The Output pane shows the execution results:

#	Time	Action	Message	Duration / Fetch
39	16:25:06	SELECT * FROM soccer.player ORDER BY height DESC LIMIT 10	10 row(s) returned	0.000 sec / 0.000 sec
40	16:25:42	SELECT * FROM soccer.player GROUP BY height ORDER BY height DES...	10 row(s) returned	0.015 sec / 0.000 sec

This screenshot shows the statement that combines two tables, player and player_attributes, joining together since they both have a column that's named player_api_id, and displays the player_api_id, player_name, and their overall_rating.

The screenshot shows the MySQL Workbench interface with a SQL query in the editor:

```
1 SELECT soccer.player.player_api_id, soccer.player.player_name, soccer.player_attributes.overall_rating
2 FROM soccer.player
3 INNER JOIN soccer.player_attributes ON soccer.player.player_api_id = soccer.player_attributes.player_api_id;
```

The Result Grid displays the following data:

player_api_id	player_name	overall_rating
505942	Aaron Appindangoye	67
505942	Aaron Appindangoye	67
505942	Aaron Appindangoye	62
505942	Aaron Appindangoye	61
505942	Aaron Appindangoye	61
155782	Aaron Cresswell	74
155782	Aaron Cresswell	74
155782	Aaron Cresswell	73
155782	Aaron Cresswell	73
155782	Aaron Cresswell	73
155782	Aaron Cresswell	73
155782	Aaron Cresswell	74
155782	Aaron Cresswell	73
155782	Aaron Cresswell	74

This screenshot shows the statement and the resulting grid that comes from trying to find the soccer player with the highest maximum overall rating from the database, combining the two tables player and player_attributes, since they both have a column with the same name player_api_id.

MySQL Workbench

Soccer x

File Edit View Query Database Server Tools Scripting Help

SQL File 11* SQL File 9* SQL File 12* x SQL File 15* team SQL File 10* SQL File 13* SQL File 16* team_attributes SQL File 17*

Limit to 1000 rows

```

1 # Find the soccer player with the highest max rating
2 • SELECT soccer.player.player_api_id, soccer.player.player_name, max(soccer.player_attributes.overall_rating) as max_rating
3 FROM soccer.player
4 INNER JOIN soccer.player_attributes ON soccer.player.player_api_id = soccer.player_attributes.player_api_id;

```

Result Grid

player_api_id	player_name	max_rating
505942	Aaron Appindangoye	94

Result 4 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
40	16:25:42	SELECT * FROM soccer.player GROUP BY height ORDER BY height DES...	10 row(s) returned	0.015 sec / 0.000 sec
41	16:37:21	SELECT soccer.player.player_api_id, soccer.player.player_name, max(socc...	1 row(s) returned	0.547 sec / 0.000 sec

Object Info Session

This screenshot shows the statement and the resulting grid that comes from trying to find the teams that have the highest defensive aggression rating and defensive pressure rating recorded in their games

MySQL Workbench

Soccer x

File Edit View Query Database Server Tools Scripting Help

SQL File 11* SQL File 9* SQL File 12* SQL File 15* SQL File 10* SQL File 16* team_attributes team SQL File 17* SQL File 13* x

Limit to 1000 rows

```

1 # Find the teams who have highest defensive aggression rating recorded in their games
2 • SELECT soccer.team.team_api_id AS team_id,
3 soccer.team.team_long_name AS team_name,
4 soccer.team_attributes.defencePressure AS pressure_rating,
5 soccer.team_attributes.defenceAggression AS aggression_rating
6 FROM soccer.team
7 JOIN soccer.team_attributes
8 ON soccer.team.team_api_id = soccer.team_attributes.team_api_id
9 ORDER BY aggression_rating DESC;

```

Result Grid

team_id	team_name	pressure_rating	aggression_rating
8178	Bayer 04 Leverkusen	61	72
9789	Borussia Dortmund	65	71
9993	Beerschot AC	70	70
9985	Standard de Liège	70	70
8203	KV Mechelen	60	70

Result 5 x

Output

Action Output

#	Time	Action	Message	Duration / Fetch
47	16:54:16	SELECT soccer.team.team_api_id AS team_id, soccer.team.team_long_na...	528 row(s) returned	0.000 sec / 0.000 sec
48	16:59:51	SELECT soccer.team.team_api_id AS team_id, soccer.team.team_long_na...	528 row(s) returned	0.000 sec / 0.000 sec

Object Info Session