

Project 7: Fédération Internationale de Football Association

Corrections

- Mar 7: for q2, there is a tie for highest wage. Break the tie in favor of the first player, and only output that player's name.
- Mar 11: for q20, the tests have been modified to accept either of two correct answers: Chilian Primera División OR Campeonato Brasileiro Série A

Intro

Let's Play Fifa18, Python style! In this project, you will get more practice with lists and start using dictionaries. Start by downloading `test.py` and `Fifa18.csv`. This dataset is too large to preview on GitHub (>17K rows), but you can view the [raw version](#) or using a program such as [Excel](#). You can also preview the first 100 rows [here](#). For this project, you'll create a new `main.ipynb` and answer questions in the usual format.

The Data

Try to familiarize yourself with the data before starting the analysis. We have players belonging to a wide range of nationalities and clubs in Fifa18. As you can see the numeric data includes their weekly wages (in Euros) (Yes, wages are per week!), net worth of the player (in Euros) and the performance rating (score out of 100). For instance, the player named "Neymar" is associated with Brazil, is signed up by club "Paris Saint-Germain", and is paid a weekly wage of 280000 Euros.

To ingest the data to your notebook, paste the following in an early cell:

```
import csv
fifa_file = open('Fifa18.csv', encoding='utf-8')
file_reader = csv.reader(fifa_file)
player_data = list(file_reader)
header = player_data[0]
player_data = player_data[1:]
for row in player_data:
    for idx in [2,6,7,8]:
        row[idx] = float(row[idx])
```

Consider peeking at the first few rows:

```
print(header)
for row in player_data[:5]:
    print(row)
```

It's up to you to write any functions that will make it more convenient to access this data.

Let's Start!

Question 1: what is the name of the oldest player?

Question 2: what is the name of the highest-paid player?

If two players are paid the same, break the tie in favor of whoever appears first in the dataset.

Question 3: what is the name of the highest net-worth player?

Question 4: what club is that player in?

Complete the following function in your notebook:

```
def get_column(col_idx):
    pass # replace this
```

The function extracts an entire column from `player_data` to a list, which it returns. For example, imagine `player_data` contained this:

```
[
    ["a", "b", "c"],
    ["d", "e", "f"],
    ["g", "h", "i"]
]
```

Then column 0 is `["a", "d", "g"]`, column 1 is `["b", "e", "h"]`, and column 2 is `["c", "f", "i"]`. A call to `get_column(1)` should therefore return `["b", "e", "h"]`, and so on.

Question 5: what are the first five nationalities listed in the dataset?

Use `get_column`, then take a slice from the list that is returned to you.

Question 6: which five names are alphabetically first in the dataset?

By alphabetically, we mean according to Python (e.g., it is true that `"B" < "a"`), so don't use the `lower` method or anything.

Don't deduplicate names in this output in the case that multiple players have the same name.

Question 7: what is the average net worth?

Question 8: what is the average age?

Define a function `player_count` that takes a parameter, `country`, and counts the number of players belonging to that country. This function will be useful for the questions that follow.

Question 9: how many players have Portugal as their nationality?

Question 10: how many players have Brazil as their nationality?

Question 11: which country has the most players participating in FIFA18?

The `player_count` function can be useful here.

Hint 1: You will first need the list of countries participating in FIFA18.

Hint 2: Make sure you aren't calling `player_count` more times than necessary. If you're not careful, the code will be very slow to execute!

Define a function `player_to_dict` that takes a parameter, `player_id`, and returns a dict containing all the information about the player that matches. Find the player row by matching `player_id` to the `Id` column in the data.

Question 12: what are the stats for the player with `Id` equal to 20801?

Use your `player_to_dict` function. The output should be a dictionary like this:

```
{'Id': '20801',  
 'name': 'Cristiano Ronaldo',  
 'Age': 32.0,  
 'nationality': 'Portugal',  
 'club': 'Real Madrid CF',  
 'league': 'Spanish Primera División',  
 'euro_wage': 565000.0,  
 'networth': 95500000.0,  
 'score_of_100': 94.0}
```

Question 13: what are the stats for the player with `Id` equal to 190871?

Question 14: what are the stats for the player with `Id` equal to 158023?

Question 15: what are the stats for the player with `Id` equal to 192985?

Question 16: how many players are there per nationality?

Answer in the form of a dictionary where the key is the nationality and the value is the number of players for that nationality.

```
{'Portugal': 355,  
 'Argentina': 948,  
 'Brazil': 800,  
 'Uruguay': 150,  
 'Germany': 1132,  
 'Poland': 332,  
 ...  
}
```

Question 17: how many players are there per league?

Question 18: what is the average player wage per league?

Question 19: what is the average player age per league?

Question 20: which league has the highest average age?