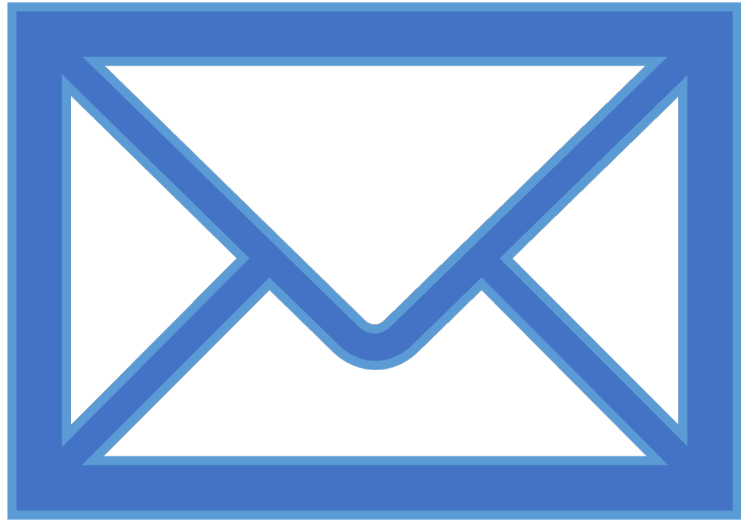


DS Challenge

Toni Almagro



You've got an email!

Dear Toni

As a continuation of the recruiting process we include in this email the instructions for the challenge. Please review the instructions below, and do not hesitate to ask any question about the format of the data files, or the requirements of the exercises.

---

The data is available at <https://amadeus.box.com/s/wh94xpc8tycxl2dvlax06wvs6128vuoy>

You should do all the work in a Python notebook.

- Export the notebook to a public notebook viewer (e.g. [Http://nbviewer.ipython.org](http://nbviewer.ipython.org)).
- Required: include all your code in Github
- Please promptly update the repository with your ongoing work.
- Do not upload the final result when you get it done. Instead, do your work in the public so we can follow your regular updates.



Exercises

# Exercise 1

- **Count the number of lines in Python for each file**





# Exercise 2

## Top 10 arrival airports in the world in 2013 (using the bookings file)

- Arrival airport is the column `arr_port`. It is the IATA code for the airport
- To get the total number of passengers for an airport, you can sum the column `pax`, grouping by `arr_port`. Note that there is negative `pax`. That corresponds to cancelations. So to get the total number of passengers that have actually booked, you should sum including the negatives (that will remove the canceled bookings).
- Print the top 10 arrival airports in the standard output, including the number of passengers.
- **Bonus point:** Get the name of the city or airport corresponding to that airport (programmatically, we suggest to have a look at **GeoBases in Github**)
- **Bonus point:** Solve this problem using `pandas` (instead of any other approach)

## Exercise 3

**Plot the monthly number of searches for flights arriving at Málaga, Madrid or Barcelona**

- For the arriving airport, you can use the Destination column in the searches file.
- Plot a curve for Málaga, another one for Madrid, and another one for Barcelona, in the same figure.
- **Bonus point:** Solving this problem using pandas (instead of any other approach)



# Exercise 4

## Match searches with bookings

- For every search in the searches file, find out whether the search ended up in a booking or not (using the info in the bookings file). For instance, search and booking origin and destination should match.
- For the bookings file, origin and destination are the columns `dep_port` and `arr_port`, respectively.
- Generate a CSV file with the search data, and an additional field, containing 1 if the search ended up in a booking, and 0 otherwise.





## Exercise 5

### Write a Web Service

- Wrap the output of the second exercise in a web service that returns the data in JSON format (instead of printing to the standard output).
- The web service should accept a parameter  $n > 0$ . For the top 10 airports,  $n$  is 10. For the  $X$  top airports,  $n$  is  $X$