# Assessment for DE Internship at DataGrokr

Thank you for your interest in the Data Engineering Internship at DataGrokr. We anticipate the selected candidates to be working in Data Engineering and Cloud-related projects. As such for this given assignment, we'd like to test candidates' skills in those areas. Candidates who are already proficient in SQL, Python, and Flask will have an edge in this assignment but even if you don't know anything about any of these technologies you should be able to do this assignment by following along the instructions and studying the links provided.

Please note that this ability to learn new technologies while following instructions would really help you in your day-to-day activities at DataGrokr.

**What you need to do:**

The objective of the assignment is to test your proficiency in coding and data analysis skills. The assignment has 3 parts.

Section 1: Environment setup and Data Cleaning

Section 2: Data Analysis

Section 3: Exposing results in API

Note: Please follow industry standards while writing the code. Preferred Programming language – Python

## Section 1: Environment setup and data cleaning

Open this link to create a new Collab notebook (You need to sign in to your google account if not signed). Follow the steps to proceed further in your notebook.

Reference for python setup in collab.

**Dataset: Please find the dataset from here.**

Please read the records from the csv file and create a DataFrame(DF) . Clean the DF using the following steps:

- Identify and remove the following columns:
  a. ad_type
  b. title
  c. description
  d. l4
  e. l5
  f. l6
- Investigate and discard all rows that contain Null values in any of these fields:
  a. lon
  b. lat
  c. price_period
  d. bedrooms
  e. surface_total
  f. rooms
  g. price
  h. surface_covered
- Once cleaning is done, separate the DF into 2 Tables. One with property_details and another with property_price_details and load them into SQL database on your local (Ex. SQLite, sqlalchemy). Columns for each should be given as mentioned below.

**Property_Details columns:** id, start_date, end_date, created_on, lat, lon, l1, l2, l3, rooms, bedrooms, bathrooms, surface_total, surface_covered

**Property_Price_Details columns:** id, price, currency, price_period, property_type, operation_type

**Deliverables: Notebook containing data cleaning commands, cleaned dataframes, Code to connect to DB**

## Section 2: Data Analysis

Use the cleaned local database to fetch the following results:

- Retrieve properties that have a price greater than 1 million and are located in "Estados Unidos" (l1).
- Categorize properties based on their surface area as 'Small' if it's less than 50 square meters, 'Medium' if it's between 50 and 100 square meters, and 'Large' if it's greater than 100 square meters:
- List all properties (id) in the "Belgrano" neighborhood (l3) that have the same number of bedrooms and bathrooms as another property in the dataset:
- Calculate the average price per square meter (price / surface_total) for each property type (property_type) in the "Belgrano" neighborhood (l3):
- Identify properties that have a higher price than the average price of properties with the same number of bedrooms and bathrooms.
- Calculate the cumulative price for each property type, ordered by the creation date.
- Identify the 10 locations (l3) with the highest total surface area (sum of surface_total) of properties listed for sale (operation_type = 'Venta'):
- Find the top 5 most expensive properties (based on price) in the "Palermo" neighborhood (l3) that were listed in August 2020:
- Find the top 3 properties with the highest price per square meter (price divided by surface area) within each property type.
- Find the top 3 locations (l1, l2, l3) with the highest average price per square meter (price / surface_total) for properties listed for sale (operation_type = 'Venta') in the year 2020. Exclude locations with fewer than 10 properties listed for sale in 2020 from the results.

**Deliverables: Notebook containing 10 SQL queries.**

## Section 3: Expose the results in API

Set up the flask server:

Flask is a popular framework in Python used for creating REST API. The following resources will be helpful for learning about REST APIs and Flask.  Here are some links to know more about REST APIs:

https://dzone.com/articles/an-introduction-to-restful-apis/

http://www.restapitutorial.com/

• Here are some links to learn more about flask and flask-RESTful:

http://flask.pocoo.org/docs/0.12/

https://flask-restful.readthedocs.io/en/latest/

• There are tons of tutorials online on using Flask to build a REST API. Here are a few examples:

https://medium.com/python-pandemonium/build-simple-restful-api-with python-and-flask-part-1-fae9ff66a706/

- Create a Flask Server in Collab notebook.
    a. The server cell should not keep on running and should exit once the server is in running state.
    b. Check the functionality by running a curl command.
- Please make sure to define the methods in different cells than the server one to maintain clean code.
- Create API endpoints for the 10 questions in Section 2 and print the results as the API response.
- You only need to implement the GET HTTP method. The method names will be like /question-1
- The API methods will be the question number and it should return the results. (Please hardcode the results you received from Sec2) Example of API calls: http://127.0.0.1:6000/question-4
- The API should do some basic error handling. Example: Invalid Question Number.
- The code should be well commented on and formatted as per PEP8 standards.

Bonus:

- Once your API is working as expected, create a ngrok HTTP share in your notebook to create a URL that you can share with us. To know more about it go through: https://ngrok.com/

**Deliverables : Python Code Collab Notebook , Ngrok HTTP share(Bonus)**

## Deliverables:

1. A single collab notebook where you have developed the code for Section 1, Section 2 and Section 3. Along with the respective deliverables mentioned in the section (if any).

- Move the deliverables to folder in your drive with folder name DE_SOLUTION_FirstName_LastName.
- Collab Notebook name: Assignment_Solution_FirstName_LastName.ipynb
- Upload your up-to-date resume to DE_SOLUTION_FirstName_LastName/ folder in your drive with name FirstName_LastName.pdf
- Now email us your google drive folder link of DE_SOLUTION_FirstName_LastName/

NOTE: Make sure to share the google drive link after choosing: Anyone with the link and Viewer Mode

We will run the colab notebook on our end and correct your submissions.

2. Your code will be evaluated not just based on final results but also on code quality. Here are few tips:

- Follow coding standards (PEP-8)
- Appropriate error/exception handling
- Modular function design
- Schema and data validation

3. Your final submission should be sent to dataengineering@datagrokr.com . Your submissions are due to us by end of day 26th Sep 2023 and subject should follow following pattern <Mail Submissions, College name> Data Engineer:

4. Please try to solve this assignment without copying from someone or somewhere. We strictly discourage plagiarism and will have a detailed discussion on your approach to solving this assignment in your next rounds of interview.

Hope you learned something from this assignment. Waiting to hear from you soon. Thanks.

**DataGrokr Hiring Team**