

**Universidad Autónoma de Occidente**

**Workshop 01**

**Subject:** ETL

**Teacher:** Javier Alejandro Vergara Zorrilla

**Name:** Juan Jacobo Delgado Melo

**Code student:** 2230110

**Objective of the Workshop**

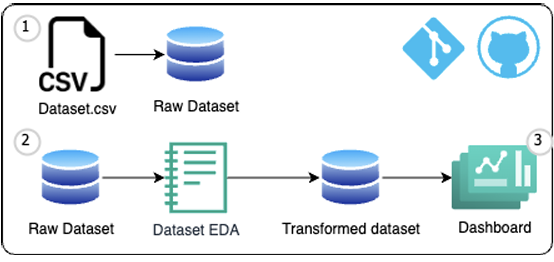
The purpose of this workshop is to implement a complete **ETL (Extract, Transform, Load) pipeline** using a real-world dataset of job candidates. The goal is to extract data from a raw CSV file, transform it through **exploratory data analysis (EDA) and data cleaning**, and load it into a **PostgreSQL database** for further analysis and visualization.

This pipeline allows us to **structure, clean, and analyze** the hiring process by understanding key patterns in hiring trends, technology preferences, and seniority levels. Additionally, the processed data is visualized using **Power BI**, enabling insightful decision-making based on real-world recruitment data.

**Tools used:**

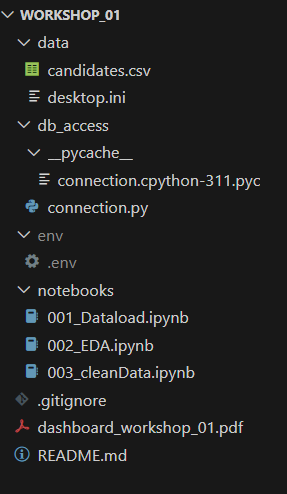
* Python
* Jupyter Notebook
* PostgreSQL
* Power BI

The pipeline with which guidance was provided during the process was as follows:



**Repository Structure**

The repository is structured as follows:



* **data/candidates.csv:** The raw dataset containing job candidate information.
* **db\_access/connection.py**: Handles the database connection setup for PostgreSQL.
* **env/.env:** Stores environment variables (e.g., database credentials) and is excluded from version control.
* **notebooks/**

**001\_Dataload.ipynb**: Loads raw data from the CSV file into the PostgreSQL database.

**002\_EDA.ipynb**: Performs **Exploratory Data Analysis (EDA)** to understand the dataset's structure, trends, and anomalies.

* **003\_cleanData.ipynb:** Cleans and transforms the data, including standardization, feature engineering, and handling missing values.
* **.gitignore**: Excludes sensitive files such as .env from version control.
* **dashboard\_workshop\_01.pdf**: The final Power BI dashboard with visualizations based on the cleaned data.
* **README.md**: A detailed guide on how to set up and run the project.

**Data Loading Process (001\_Dataload.ipynb)**

The data loading phase is the first step of the ETL process, where raw data is extracted from the candidates.csv file and stored in a PostgreSQL database. The key steps executed in 001\_Dataload.ipynb include:

**Setting Up the Database Connection**: Using SQLAlchemy, a connection to PostgreSQL is established by retrieving credentials from the .env file.

**Reading the Raw CSV Data**: The dataset is loaded into a Pandas DataFrame, ensuring that column separators and encodings are correctly interpreted.

**Data Integrity Checks**: The structure of the dataset is verified, checking for missing values, duplicate records, and inconsistencies.

**Loading Data into PostgreSQL**: The raw data is inserted into a table called candidates, replacing any existing data to ensure freshness.

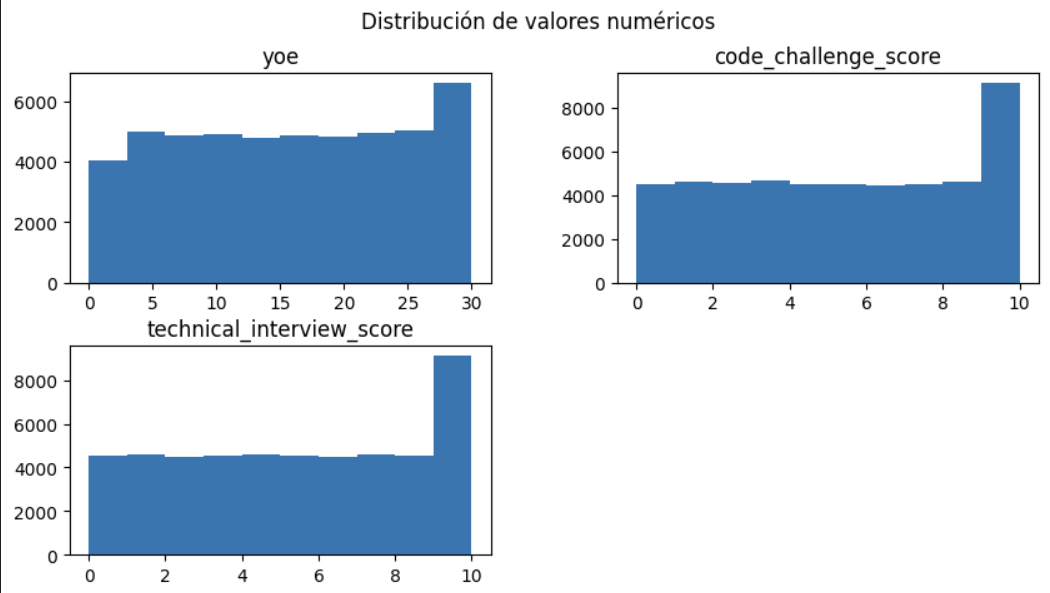
**Validation of Successful Insertion**: A query is run to confirm that the records have been successfully stored in the database.

This notebook serves as the foundation for subsequent data analysis and transformation steps, ensuring that raw data is properly structured and accessible for further processing.

**Exploratory Data Analysis (EDA) 002\_EDA.ipynb**

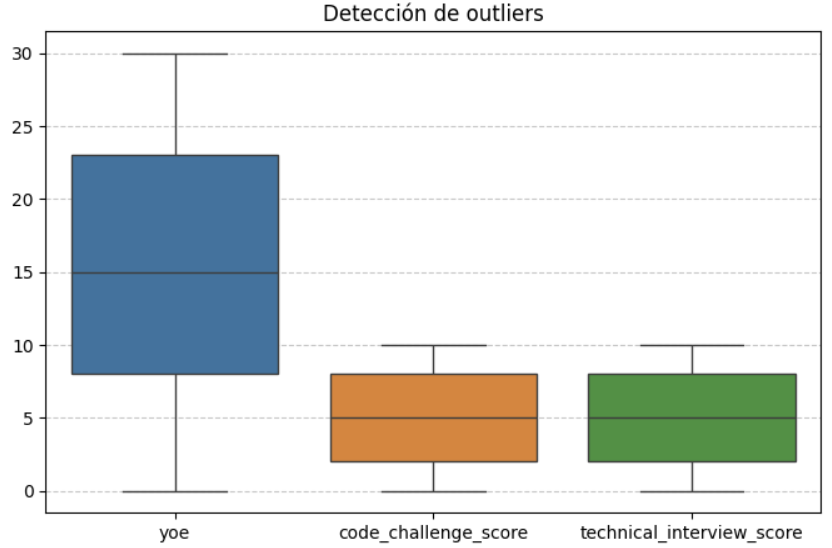
An exploratory data analysis (EDA) was conducted by connecting directly to the PostgreSQL database. Using Python, various statistical and visual techniques were applied to understand the dataset, identify patterns, and detect possible inconsistencies. The analysis was focused on:

**Understanding the Data Distribution**: Histograms were used to examine the spread of key numerical variables such as years of experience, code challenge scores, and technical interview scores.



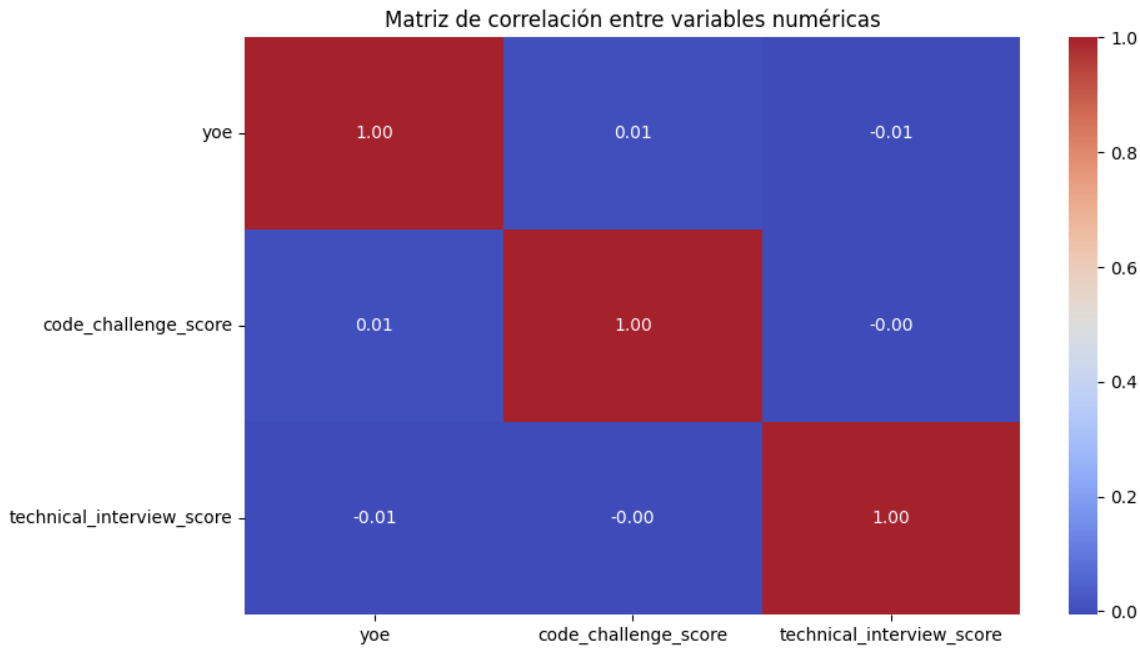
The distribution of **years of experience (yoe)** shows a uniform trend, but with a significant accumulation at 30 years, suggesting that this value could be a maximum limit established in the data and not a true reflection of the candidates' experience. As for the **Code Challenge Score** and **Technical Interview scores,** the distribution is relatively uniform, although a peak is observed at the maximum value of 10, which could indicate that many candidates reached the maximum score, possibly due to the structure of the test or the way it is assessed. These patterns suggest that it would be useful to analyze whether the scores adequately reflect performance and whether there are biases in the evaluation of candidates.

**Outlier Detection**: Box plots were generated to detect any extreme values that might indicate inconsistencies or anomalies.



According to this outlier plot, it can be concluded that no outliers were found, so it is not necessary to eliminate data due to outliers.

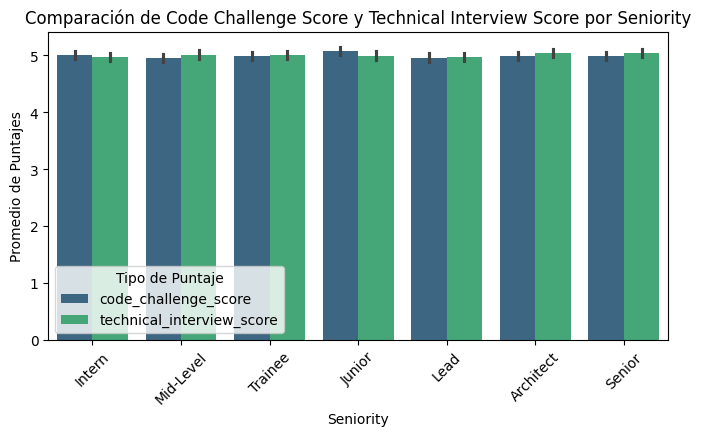
**Correlation Analysis**: A heatmap was created to evaluate the relationships between numerical variables and determine if any strong correlations existed.



The correlation matrix shows that there is no significant relationship between the numerical variables analyzed. **Work experience** (yoe) does not have a strong correlation with the Code Challenge Score (0.01) nor with the **Technical Interview** (-0.01), indicating that **years of experience** do not directly influence performance on these tests. Likewise, the relationship between the **Code Challenge Score** and the **Technical Interview** is virtually nonexistent (-0.00), suggesting that candidates who score well on one test do not necessarily perform well on the other. These results indicate that the assessments may be measuring different skills and that seniority or previous experience does not guarantee better performance on the technical tests.

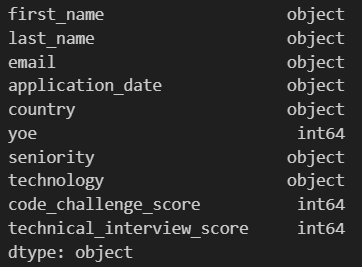
**Seniority Level Analysis**: A comparative bar chart was plotted to analyze how different seniority levels performed in code challenges and technical interviews.

**Do candidates with higher levels of seniority score higher on tests and interviews?**



The graph shows that the average Code Challenge Score and Technical Interview Score is very similar at all seniority levels, with no significant differences. This indicates that work experience does not directly influence performance on these tests, suggesting that the tests evaluate specific technical skills rather than career path.

**Column data types:**



**Data Cleaning (003\_cleanData.ipynb)**

The data cleaning and transformation phase focused on ensuring data quality and preparing the dataset for analysis and visualization. The key steps performed in 003\_cleanData.ipynb include:

**Renaming Columns**: Standardized column names by replacing spaces with underscores for consistency and ease of access.

**Date Transformation**: Converted the column **application\_date** from string format into an integer representation of the year, month, and day.

**Creation of the hired Column**: Introduced a new column indicating whether a candidate was hired or not. A candidate is considered hired if both their **code\_challenge\_score** and **technical\_interview\_score** is greater than or equal to 7.

**Handling Missing Values**: Identified and appropriately handled missing values in the dataset.

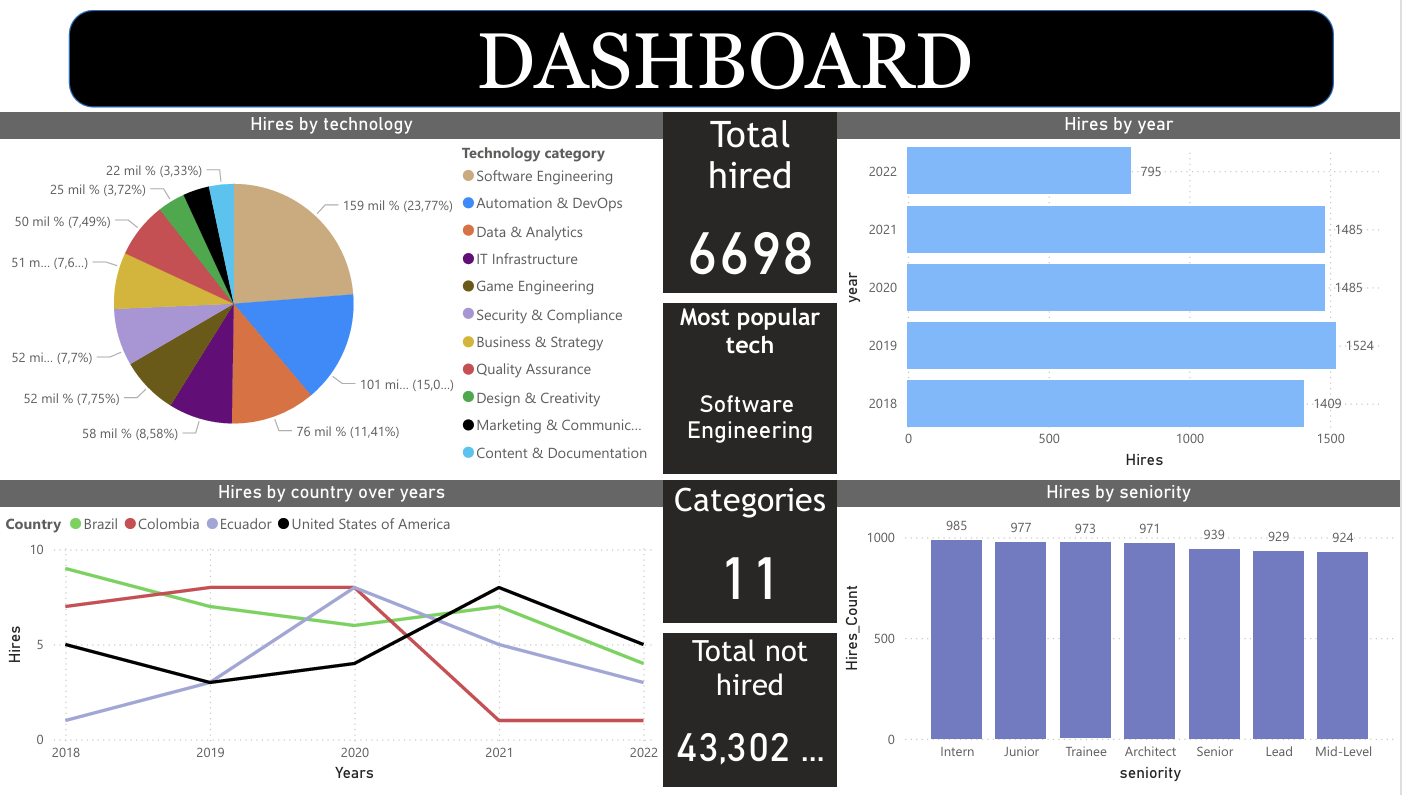
**Technology Grouping:** Mapped the technology column into broader categories to enhance the clarity of technology trends.

**Normalization**: Scaled numerical features to a common range to ensure uniformity and comparability across all data points.

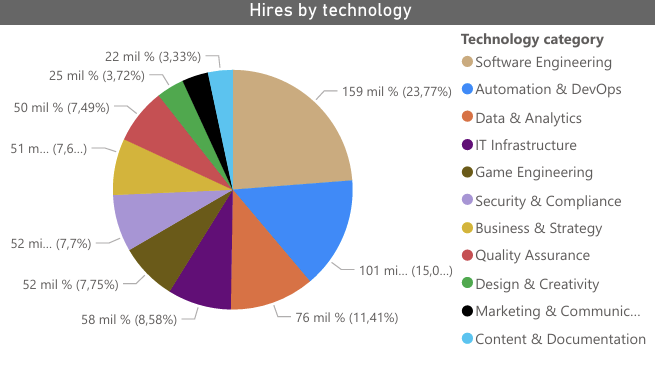
**Storing the Cleaned Data**: Loaded the transformed dataset into a new PostgreSQL table **candidates\_eda** for use in Power BI dashboards.

**Dashboard and Conclusions**

The next dashboard was created from the data cleaning database:

****

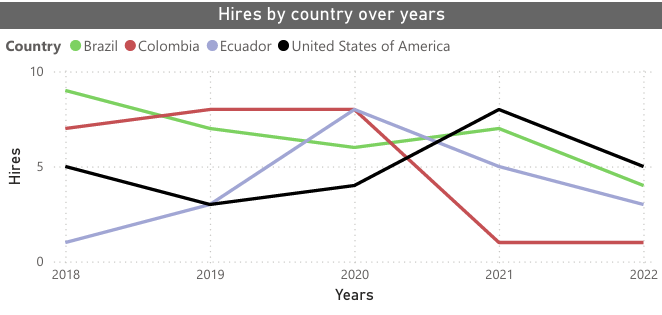
**Now each graph is described with its corresponding analysis:**



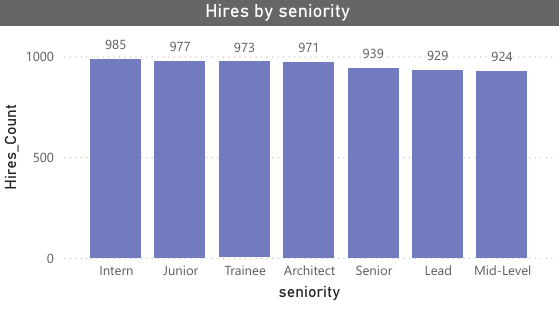
The "Hires by Seniority" bar chart shows a fairly balanced distribution of hires across all seniority levels. Interns (985 hires) had the highest number of hires, followed closely by Junior, Trainee, and Architect positions, indicating a focus on early career hiring. Senior, Lead, and Mid-Level positions had slightly fewer hires, but remained consistent, indicating steady demand across all experience levels.



The "Hires by Year" horizontal bar graph shows the number of hires in each year. The data shows a relatively consistent hiring trend from 2018 to 2021, with each year capturing between 1,400 and 1,524 hires. However, the significant drop in 2022 (795 hires) is due to the fact that the dataset only captures data through June 4, 2022, and does not indicate an actual decline in hiring. This chart is useful for assessing year-over-year hiring trends, and may be useful for understanding hiring patterns or external factors affecting hiring in different years.



The multi-line "Hires by Country Over Years" chart shows hiring trends for Brazil, Colombia, Ecuador, and the United States from 2018 to 2022. The United States (black line) shows a steady increase in hires, becoming the leading country by 2021. Colombia (red line) shows growth initially, but experiences a sharp decline after 2020 and remains low in hires in 2021 and 2022. Brazil (green line) had a consistent presence but showed a slight downward trend over time. Ecuador (blue line) showed a gradual increase in hiring through 2020, but experienced a decline in subsequent years. This visualization highlights shifting hiring patterns across countries that may be influenced by market trends, company strategies, or external economic factors.



The "Hires by Seniority" bar chart shows a fairly balanced distribution of hires across all seniority levels. Interns (985 hires) had the highest count, followed closely by Junior, Trainee, and Architect roles, indicating a focus on early-career hiring. Senior, Lead, and Mid-Level positions had slightly fewer hires but remained consistent, suggesting steady demand across all experience levels.