

# ETL Workshop1 Challenge

Isaac Piedrahita (2226259)

Cuarto Semestre de Ingeniería de Datos e Inteligencia Artificial

Javier Alejandro Vergara Zorrilla

ETL (Extract, transform and load)



Ingeniería de datos e inteligencia artificial, Facultad de ingeniería

Universidad Autónoma de Occidente

Santiago de Cali

2024

# Introducción

Este documento detalla el proceso seguido para abordar y resolver el Workshop de "Python Data Engineer", un ejercicio práctico diseñado para **simular** un **escenario real** de entrevista de trabajo. El objetivo principal era demostrar habilidades en la **gestión** y **visualización** de **datos**, partiendo de un archivo CSV con información aleatoriamente generada sobre candidatos que participaron en procesos de selección.

## Metodología

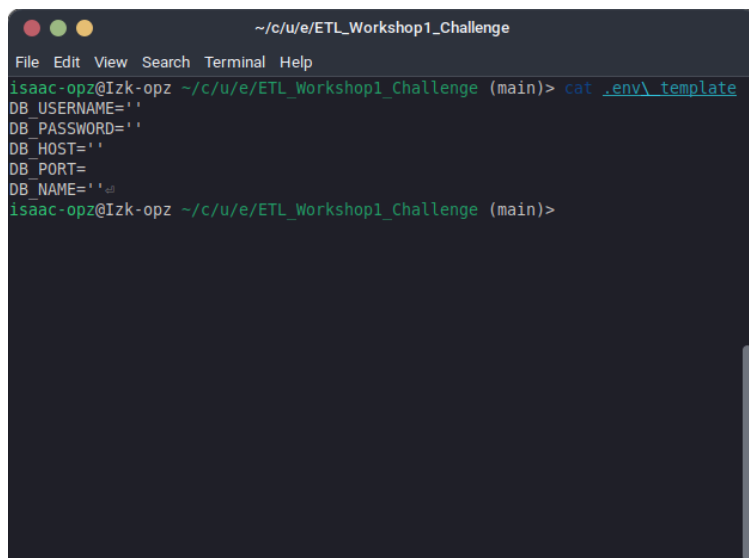
La solución del workshop implicó varios pasos clave, desde la preparación inicial del entorno de desarrollo hasta la migración de datos y la generación de visualizaciones. A continuación, se describen los pasos fundamentales del proceso:

### 1. Preparación del Entorno de Desarrollo

Para el análisis de datos y desarrollo del proyecto, se decidió utilizar **Jupyter Notebook** como nuestro entorno de desarrollo principal, gracias a su capacidad para combinar código, visualizaciones y explicaciones en un solo lugar de manera **interactiva**. **Python** fue el lenguaje de elección para este taller. Además, para manejar las librerías y sus dependencias de forma organizada, se creó un entorno virtual usando **virtualenv**. Esto ayudó a mantener el proyecto ordenado y facilitó la gestión de las versiones de las herramientas necesarias, asegurando que se trabajara en un entorno consistente.

### 2. Manejo de Credenciales de la Base de Datos

Para asegurar la integridad de las **credenciales** de la base de datos, se emplearon archivos `.env` con la ayuda de la librería de **Python**, **dotenv**. Este enfoque permite almacenar de forma segura información sensible como el nombre de usuario, contraseña, host, puerto y nombre de la base de datos, evitando exponer dichos detalles directamente en el código fuente.



```
~/c/u/e/ETL_Workshop1_Challenge
File Edit View Search Terminal Help
isaac-opz@Izk-opz ~/c/u/e/ETL_Workshop1_Challenge (main)> cat .env\template
DB_USERNAME=''
DB_PASSWORD=''
DB_HOST=''
DB_PORT=
DB_NAME=''
isaac-opz@Izk-opz ~/c/u/e/ETL_Workshop1_Challenge (main)>
```

para poder leer las credenciales desde Python, se hizo uso de la librería **OS**

```
db_username = os.getenv("DB_USERNAME")
db_password = os.getenv("DB_PASSWORD")
db_host = os.getenv("DB_HOST")
db_port = os.getenv("DB_PORT")
db_name = os.getenv("DB_NAME")
```

### 3. Configuración de la Base de Datos Relacional

Como solución para el alojamiento de nuestra base de datos, Se optó por **PostgreSQL** como sistema de gestión de base de datos, dada su robustez y capacidades avanzadas para manejar grandes volúmenes de datos. En consecuencia, se procedió a establecer una instancia de base de datos en la nube utilizando la plataforma [Neon](#). Esta decisión estratégica permitirá alojar de manera eficiente los datos referentes a los candidatos contratados, **garantizando** tanto la **escalabilidad** como la **seguridad** en el manejo de la información.

### 4. Migración de Datos a PostgreSQL

La migración de datos desde el archivo CSV hacia la base de datos alojada en la nube se realizó mediante un script de Python (**migration.py**), el cual utilizó **pandas** para la lectura de datos y **SQLAlchemy** para establecer la conexión con la base de datos y ejecutar la migración.

### 5. Análisis y Visualización de Datos

Con los datos ya disponibles en la base de datos PostgreSQL, se procedió a realizar el análisis requerido y a generar las visualizaciones especificadas en el enunciado del workshop. Todo esto en la plataforma **Looker Studio** Estas incluyeron:

- Hires by technology (gráfico de tarta)
- Hires by year (gráfico de barras horizontal)
- Hires by seniority (gráfico de barras)
- Hires by country over years (gráfico de líneas múltiples)

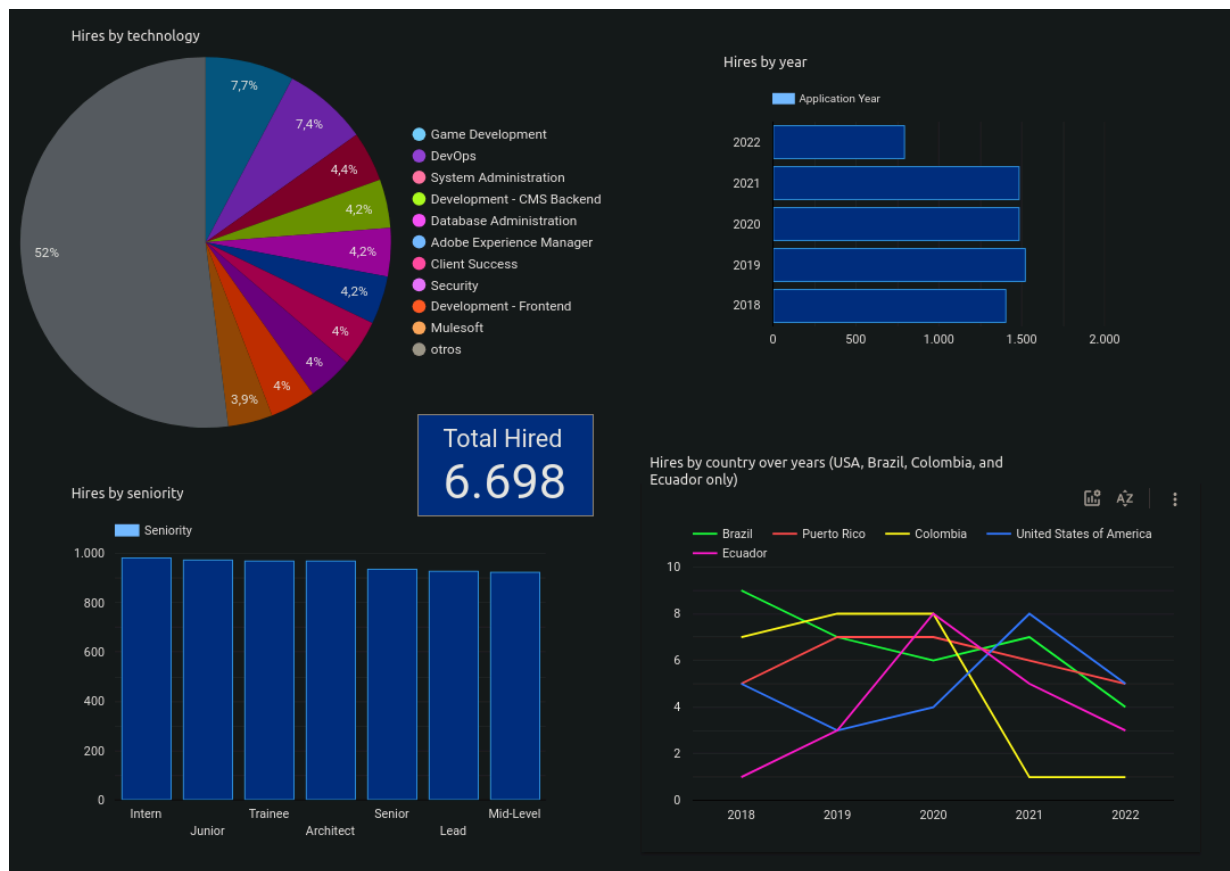
Puede ver el dashboard a detalle en el siguiente [enlace](#)

#### Análisis Dashboard:

-6698 registros de Contratados

- Niveles de seniority de los candidatos varían mucho, se encuentran empleados de nivel de entrada como Interns o Juniors, hasta gente con muchos años de experiencia como Leads o seniors. Aunque aún así todos los niveles de seniority presentan prácticamente la misma cantidad de contratados

- La tecnología en la que los candidatos están especializados es muy variada. Las únicas tecnologías que realmente destacan son **DevOps** y **Game Development**, de resto todas las demás tecnologías presentan valores muy similares en cuanto a candidatos contratados



## Análisis Exploratorio de Datos (EDA)

**Carga y Preparación de Datos:** Se cargaron los datos de los candidatos desde un archivo CSV y se identificaron a aquellos que fueron contratados basándose en las puntuaciones de los desafíos de código y las entrevistas técnicas.

```
import pandas as pd

df = pd.read_csv("raw_data/candidates.csv", delimiter=';', encoding='unicode_escape')
df.head()
```

	First Name	Last Name	Email	Application Date	Country	YOE	Seniority	Technology	Code Challenge Score	Technical Interview Score
0	Bernadette	Langworth	leonard91@yahoo.com	2021-02-26	Norway	2	Intern	Data Engineer	3	3
1	Camryn	Reynolds	zelda56@hotmail.com	2021-09-09	Panama	10	Intern	Data Engineer	2	10
2	Larue	Spinka	okey_schultz41@gmail.com	2020-04-14	Belarus	4	Mid-Level	Client Success	10	9
3	Arch	Spinka	elvera_kulas@yahoo.com	2020-10-01	Eritrea	25	Trainee	QA Manual	7	1
4	Larue	Alterwerth	minnie.gislason@gmail.com	2020-05-20	Myanmar	13	Mid-Level	Social Media Community Management	9	7

We are only interested on HIRED candidates (both scores greater than or equal to 7) so we are gonna create a data frame with candidates who fulfill the requirements

**Filtrado de Candidatos:** Se estableció un criterio de selección, donde se consideraron como contratados aquellos candidatos con puntuaciones en el desafío de código y la entrevista técnica iguales o superiores a 7.

We are only interested on HIRED candidates (both scores greater than or equal to 7) so we are gonna create a data frame with candidates who fulfill the requirements

```
df_contracted = df[(df['Code Challenge Score'] >= 7) & (df['Technical Interview Score'] >= 7)].copy()
```

**Conversión y Manipulación de Fechas:** Se transformó la columna Application Date a tipo datetime y se extrajo el año para facilitar análisis posteriores relacionados con el tiempo.

```
df_contracted['Application Date'] = pd.to_datetime(df_contracted['Application Date'], errors='coerce')
```

```
df_contracted['Application Year'] = df_contracted['Application Date'].dt.year
```

**Análisis de Nulos y Tipos de Datos:** Se verificó que el dataset no contuviera valores nulos y se observó la existencia de datos tanto categóricos (objetos) como numéricos (enteros).

```
df_contracted.isnull().sum()
```

First Name	0
Last Name	0
Email	0
Application Date	0
Country	0
YOE	0
Seniority	0
Technology	0
Code Challenge Score	0
Technical Interview Score	0
Application Year	0
dtype: int64	

**Resumen Estadístico:** Se proporcionó un resumen estadístico que muestra una distribución uniforme de las puntuaciones, y se detallaron los años de experiencia de los candidatos contratados, que varían entre 0 y 30 años, con una media aproximada de 15 años.

```
numeric_summary = df_contracted.describe()
print(numeric_summary)
```

	Application Date	YOE	Code Challenge Score	\
count	6698	6698.000000	6698.000000	
mean	2020-04-10 23:23:40.005972224	15.291281	8.500000	
min	2018-01-01 00:00:00	0.000000	7.000000	
25%	2019-03-07 00:00:00	8.000000	8.000000	
50%	2020-04-09 00:00:00	15.000000	8.000000	
75%	2021-05-26 00:00:00	23.000000	9.000000	
max	2022-07-04 00:00:00	30.000000	10.000000	
std	NaN	8.843949	1.110748	

	Technical Interview Score	Application Year
count	6698.000000	6698.000000
mean	8.479248	2019.810839
min	7.000000	2018.000000
25%	7.000000	2019.000000
50%	8.000000	2020.000000
75%	9.000000	2021.000000
max	10.000000	2022.000000
std	1.126308	1.315268

**Distribución de Candidatos por País:** Los candidatos contratados provienen de una amplia variedad de países, lo que sugiere una diversidad geográfica en la contratación.

```
country_distribution = df_contracted["Country"].value_counts()
print(country_distribution)
```

Country	
Northern Mariana Islands	44
Heard Island and McDonald Islands	41
Sri Lanka	40
Seychelles	40
Niger	40
..	
Armenia	18
Saint Vincent and the Grenadines	16
Maldives	16
Montenegro	15
Guam	15

Name: count, Length: 244, dtype: int64

**Distribución de Candidatos por Tecnología:** Las tecnologías más populares entre los candidatos contratados son el Desarrollo de Juegos y DevOps, con otras tecnologías distribuidas de manera bastante uniforme.

```
technology_distribution = df_contracted["Technology"].value_counts()
print(technology_distribution)
```

```
Technology
Game Development      519
DevOps                 495
System Administration 293
Development - CMS Backend 284
Database Administration 282
Adobe Experience Manager 282
Client Success        271
Security              266
Development - Frontend 266
Mulesoft              260
QA Manual             259
Salesforce            256
Business Analytics / Project Management 255
Data Engineer         255
Development - Backend  255
Business Intelligence  254
Development - FullStack 254
Development - CMS Frontend 251
Security Compliance    250
Design                249
QA Automation         243
Sales                 239
Social Media Community Management 237
Technical Writing      223
Name: count, dtype: int64
```

**Distribución de Candidatos por Año de Aplicación:** Las aplicaciones de los candidatos contratados están distribuidas de manera relativamente uniforme entre 2018 y 2021, con un número menor sorprendentemente en 2022.

```
year_distribution = df_contracted["Application Year"].value_counts()
print(year_distribution)
```

```
Application Year
2019      1524
2020      1485
2021      1485
2018      1409
2022       795
Name: count, dtype: int64
```

**Exportación de Datos:** Finalmente, los datos filtrados de los candidatos contratados se exportaron a un nuevo archivo CSV para realizar el Dashboard.

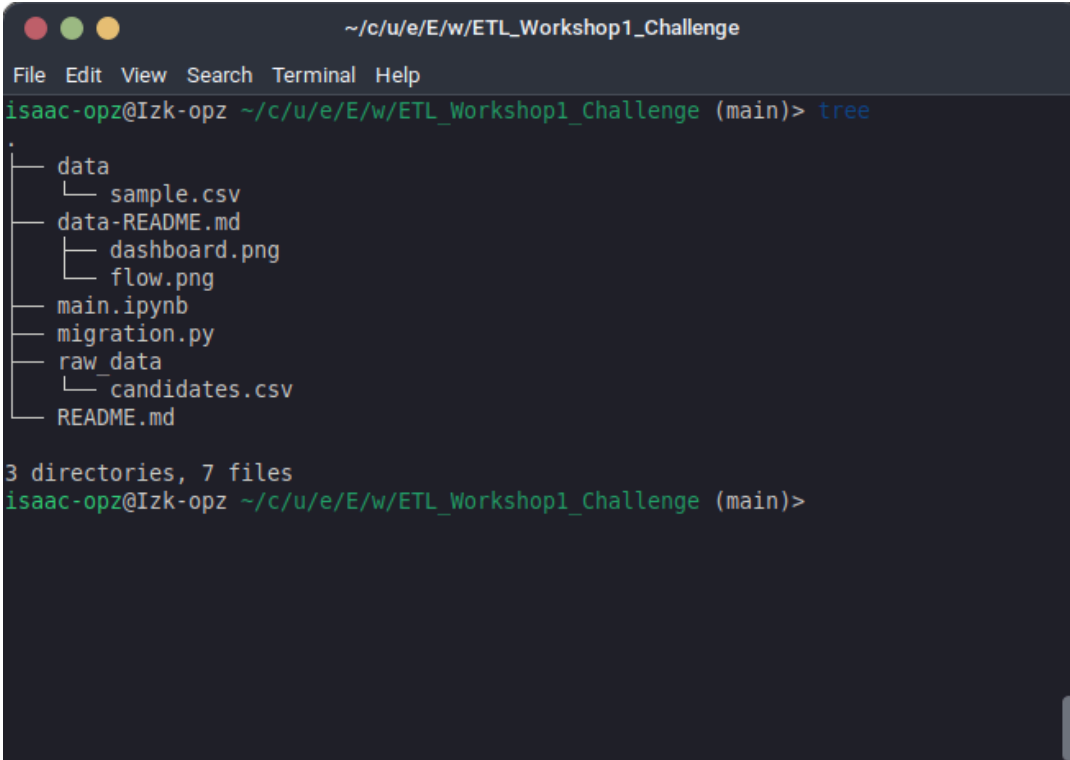
**Nota:** para más detalles sobre el código se recomienda visitar el [repositorio](#) del workshop

## Conclusiones Generales del EDA

El análisis exploratorio realizado proporcionó una comprensión profunda de los perfiles de los candidatos contratados, resaltando aspectos clave como la **experiencia**, **competencias tecnológicas** y **tendencias** a lo largo del tiempo. Se detectó una representación amplia en términos de **diversidad geográfica** y **tecnológica**, y se evidenció que las contrataciones no muestran una preferencia por un año específico.

Este conjunto de hallazgos ofrece una base sólida para decisiones estratégicas relacionadas con el reclutamiento y la planificación de recursos humanos, así como para una posterior modelización o análisis predictivo. Además, el proceso meticuloso y estructurado seguido en este EDA garantiza la reproducibilidad y la claridad en la comunicación de los resultados obtenidos.

## Listado de archivos del repositorio

A screenshot of a terminal window with a dark background. The title bar at the top shows three colored window control buttons (red, yellow, green) on the left and the path ~/c/u/e/E/w/ETL\_Workshop1\_Challenge on the right. Below the title bar is a menu bar with the items File, Edit, View, Search, Terminal, and Help. The terminal content shows a user prompt isaac-opz@Izk-opz followed by the command tree. The output of the tree command lists the directory structure: a root directory containing 'data' (with subfile 'sample.csv'), 'data-README.md' (with subfiles 'dashboard.png' and 'flow.png'), 'main.ipynb', 'migration.py', 'raw\_data' (with subfile 'candidates.csv'), and 'README.md'. Below the tree output, it says '3 directories, 7 files'. The prompt then repeats the tree command. A vertical scrollbar is visible on the right side of the terminal window.

```
~/c/u/e/E/w/ETL_Workshop1_Challenge
File Edit View Search Terminal Help
isaac-opz@Izk-opz ~/c/u/e/E/w/ETL_Workshop1_Challenge (main)> tree
.
├── data
│   └── sample.csv
├── data-README.md
│   ├── dashboard.png
│   └── flow.png
├── main.ipynb
├── migration.py
├── raw_data
│   └── candidates.csv
└── README.md

3 directories, 7 files
isaac-opz@Izk-opz ~/c/u/e/E/w/ETL_Workshop1_Challenge (main)>
```