

## Reto Técnico

El siguiente reto técnico está diseñado para evaluar las habilidades del candidato en programación con Python, manejo de bases de datos, desarrollo de Dashboards y redacción de documentación técnica. El objetivo principal es realizar un proceso completo de extracción, transformación, y visualización de datos (ETL), siguiendo un enfoque estructurado y profesional.

### Detalles del reto

#### 1. Duración:

- El candidato tendrá **1 semana** para completar el desafío.

#### 2. Objetivo general:

- Llevar a cabo un proyecto que abarque las etapas principales de un flujo ETL:
  - **Extracción de datos:** Obtener datos desde una base de datos local.
  - **Transformación de datos:** Limpieza de datos, manejo de valores nulos, y reestructuración según las necesidades analíticas.
  - **Visualización de datos:** Creación de Dashboards interactivos que respondan a 6 preguntas clave.
  - **Documentación:** Elaboración de un archivo README.md que explique el flujo de trabajo, las herramientas utilizadas y cómo reproducir los resultados.

#### 3. Habilidades evaluadas:

- **Python:** Escribir código eficiente y organizado para procesos ETL.
- **Manejo de bases de datos:** Conexión a bases de datos locales o en la nube, manipulación de datos estructurados y optimización de consultas.
- **Desarrollo de Dashboards:** Creación de visualizaciones claras y accionables con herramientas como Power BI o Excel.
- **Documentación técnica:** Explicar de forma clara y profesional cada etapa del proceso, incluyendo configuración y uso del proyecto.

#### 4. Entregables:

- Un repositorio de GitHub que contenga:
    - Código fuente.
    - Archivos de datos relevantes.
    - Documentación completa en README.md.
  - Capturas de pantalla o acceso al Dashboard interactivo desarrollado.
-

## **1. Extracción de datos:**

Utilizar Python para establecer una conexión con un archivo de Excel que actúa como base de datos local, empleando las bibliotecas necesarias para leer, manipular y extraer los datos requeridos de manera eficiente. Asegúrese de manejar adecuadamente la ruta del archivo y los formatos de datos para garantizar una integración exitosa.

1. Se utilizará una base de datos local en formato Excel que contiene registros de ventas de los últimos dos años.

La tabla principal incluirá las siguientes columnas:

- **[IdOportunidad]:** Identificador único de la oportunidad (**PK**).
- **[IdEmpresa]:** Identificador de la empresa asociada.
- **[IdPropietario]:** Identificador del propietario responsable de la oportunidad.
- **[Importe]:** Monto de venta de la oportunidad.
- **[TipoDivisaAjuste]:** Tipo de divisa correspondiente al país de la venta.
- **[Zona]:** Región o zona geográfica asignada a la oportunidad.
- **[Participantes]:** Número de personas que participan en la oportunidad.
- **[FechaCierre]:** Fecha efectiva de cierre de la oportunidad.

## **2. Limpieza y transformación de datos**

Una vez extraídos los datos, se llevará a cabo un proceso de limpieza y transformación para garantizar su calidad y usabilidad en el análisis. Este proceso incluye:

### **1. Limpieza de datos:**

- **Manejo de valores nulos:** Identificar celdas con valores faltantes en las columnas clave y asignarles un valor adecuado según el contexto, ya sea un valor por defecto, promedio, o categorización como "Sin datos".
- **Normalización de formatos:** Normalizar los formatos de texto (mayúsculas, minúsculas, eliminación de espacios innecesarios) y ajustar valores numéricos para garantizar coherencia. Por ejemplo, convertir fechas a un formato estándar y unificar el formato de divisas o montos, etc.
- **Eliminación de duplicados:** Identificar y eliminar registros repetidos para evitar distorsiones en el análisis.

### **2. Transformación de datos:**

- Modificar y estructurar los datos para adaptarlos a las necesidades del análisis, como la creación de nuevas columnas derivadas (por ejemplo, clasificación por zonas o rangos de importe).
- Realizar agrupaciones o segmentaciones que permitan responder preguntas específicas, como el rendimiento por zona o el crecimiento por propietario.
- Aplicar cualquier transformación adicional requerida para maximizar el valor analítico de los datos.

### **3. Inserción en otra base de datos**

El siguiente paso consiste en almacenar los datos limpios y transformados en una nueva base de datos para su posterior análisis. Este proceso incluye:

**1. Creación de una nueva base de datos:**

- **Base de datos local:** Crear un nuevo archivo en formato Excel, SQLite o similar que sirva como almacenamiento estructurado de los datos procesados.
- **Base de datos en la nube:** Si es posible, implementar un sistema de almacenamiento más robusto utilizando tecnologías como PostgreSQL o MySQL (en la nube o en un servidor). Esto permitirá un manejo eficiente de consultas y escalabilidad en futuros análisis.

**2. Inserción de datos:**

- Cargar los datos limpios y estructurados en la nueva tabla o archivo.
- Garantizar que la estructura de la tabla respete las reglas de integridad y facilite consultas eficientes.
- Validar que los datos insertados se almacenen correctamente sin pérdidas ni alteraciones.

### **4. Visualización de los datos**

El objetivo es crear un Dashboards interactivo que facilite la interpretación de los datos almacenados, proporcionando insights clave para la toma de decisiones. Los pasos incluyen:

**1. Creación de Dashboards:**

- **Opción 1:** Utilizar Power BI para crear un Dashboard interactivo, integrando filtros y visualizaciones dinámicas.
- **Opción 2:** Usar Excel para construir un Dashboard con tablas dinámicas, gráficos interactivos y segmentaciones que permitan explorar los datos de forma intuitiva.

**2. Preguntas clave resueltas por los gráficos:**

- **¿En qué zona (Zona) se ha dado la mejor venta en el último año?**  
Identificar la región con mayor volumen de ventas en 2024, destacando las tendencias por zona.
- **Top 3 de empresas (IdEmpresa) en crecimiento (%) del 2024 vs 2023.**  
Mostrar las empresas con mayor incremento porcentual en ventas entre los años.
- **Top 3 de asesores (IdDueño\_Oportunidad) en crecimiento (%) del 2024 vs 2023.**  
Identificar los propietarios con mejor desempeño en crecimiento anual.
- **¿Cuál es la región (Zona) con mayor disminución en ventas (Importe) y su porcentaje comparando 2024 vs 2023?**  
Evaluar las zonas que han sufrido una caída significativa en ventas y calcular el porcentaje.

- **¿De la región (Zona) de menos ventas (Importe), quién es el cliente (IdEmpresa) con mayor crecimiento comparando 2024 vs 2023?**

Resaltar clientes que han crecido significativamente en regiones con bajo desempeño.

- **¿Existe una relación entre el número de participantes (Participantes) y la zona (Zona), considerando el importe (Importe) promedio de las oportunidades por cada rango de participantes?**

Explorar si los proyectos con más participantes tienen un impacto en el importe promedio dependiendo de la zona.

- **Análisis abierto:** Realizar análisis adicionales basados en patrones relevantes, impacto de fechas específicas, o correlaciones inesperadas en los datos, etc.

### 3. Visualizaciones recomendadas:

- **Gráficos de barras:** Para top 3 de empresas o propietarios por crecimiento.
- **Gráficos de línea:** Para comparar tendencias anuales de ventas por zona o región.
- **Heatmaps o tablas dinámicas:** Para analizar relaciones entre número de participantes, zona, e importes promedio.
- **Pie charts o gráficos apilados:** Para proporciones de ventas por zona o región.

## 5. Documentación del proyecto

Se debe desarrollar una documentación completa y detallada que sirva como guía para comprender, configurar, y ejecutar el proyecto. Esta documentación debe incluir los siguientes apartados:

### 1. Introducción:

- Breve descripción del problema a resolver.
- Objetivo principal del reto, destacando los resultados esperados.

### 2. Estructura del proyecto:

- Descripción de la organización de carpetas y archivos del proyecto, explicando el propósito de cada uno.
- Ejemplo de estructura:

```

/src
├── extraction.py      # Extracción de datos desde la base de datos
├── transformation.py  # Limpieza y transformación de datos
├── loading.py         # Inserción en la nueva base de datos
└── dashboard.ipynb   # Script para crear visualizaciones y análisis
/data
├── raw_data/         # Datos originales sin procesar
├── processed_data/   # Datos procesados y listos para análisis
└── README.md         # Archivo principal con la documentación

```

3. **Requisitos técnicos:**

- Listado de herramientas, tecnologías y librerías utilizadas (por ejemplo, Python, pandas, matplotlib, Power BI, etc.).
- Especificar versiones relevantes si aplica.

4. **Descripción del código:**

Explicación detallada de las secciones clave del código, como:

- Principales funciones y clases utilizadas, destacando su propósito.
- Flujo general del script o pipeline, describiendo cómo las etapas de extracción, transformación, y carga se integran.

Proporcionar comentarios claros en el código para facilitar su comprensión.

5. **Instrucciones de configuración:**

- Pasos para configurar el entorno de desarrollo (instalación de librerías, creación de la base de datos, etc.).
- Cualquier requerimiento adicional para ejecutar el proyecto (por ejemplo, un archivo.env para variables de entorno).

6. **Guía de uso:**

- Instrucciones paso a paso sobre cómo ejecutar los scripts y reproducir los resultados.
- Detallar cómo visualizar los gráficos y dashboards generados.

## **6. Repositorio**

El proyecto debe ser alojado en un repositorio público de GitHub, que esté bien estructurado y facilite la navegación. Este repositorio debe incluir:

1. **Código y documentación:**

- Todo el código fuente del proyecto, debidamente organizado en directorios específicos.
- Archivo **README.md** con la documentación completa descrita en el apartado anterior.
- Archivos de datos relevantes, tanto originales como procesados, separados en carpetas adecuadas.

2. **Entrega del repositorio:**

- Compartir el enlace al repositorio con los usuarios de GitHub **antoniogJTec** y **Nathansoul** antes de la segunda entrevista.
  - Asegurarse de que el repositorio sea accesible, con permisos para visualizar todos los archivos relevantes.
-

## **Criterios de evaluación**

### **1. Calidad del código:**

- Legibilidad y organización.
- Uso adecuado de funciones, clases y modularidad.

### **2. Base de datos:**

- Correcta extracción y limpieza de datos.
- Inserción correcta en la base de datos secundaria.

### **3. Dashboard y visualización:**

- Claridad y relevancia de los gráficos.
- Interactividad en el caso de Power BI o Excel.

### **4. Documentación:**

- Completa, clara y estructurada.
- Facilita la comprensión del proyecto y la reproducción de resultados.

### **5. Gestión del repositorio:**

- Organización y claridad de carpetas.