

# Trabajo Final

## Challenge Data Analytics - Python 🚀

¡Te damos la bienvenida al Challenge de Data Analytics con Python!

En este documento podrás ver todos los detalles del proyecto que deberás realizar para aprobar con el cursado del módulo.

¿Estás listo? ¡Empecemos! 🏁

### Objetivo 📌

Para resolver este challenge, deberás crear un proyecto que consuma datos en la cual podrás seleccionar 3 fuentes distintas para poblar una base de datos SQL.

### Requerimientos funcionales 🔍

Tu proyecto deberá cumplir con una serie de requerimientos funcionales que giran en torno a cuatro ejes centrales: los archivos fuente, el procesamiento de datos, la creación de tablas en la base de datos y la actualización de la base de datos.

Veamos cada uno de ellos en detalle.

### Objetivo 📌

Para resolver este challenge, deberás crear un proyecto que consuma datos en la cual podrás seleccionar 3 fuentes distintas para poblar una base de datos SQL con la información contenida en la Response de nuestro script de Python.

Requerimientos funcionales 🔍

Tu proyecto deberá cumplir con una serie de requerimientos funcionales que giran en torno a cuatro ejes centrales: los archivos fuente, el procesamiento de datos, la creación de tablas en la base de datos y la actualización de la base de datos.

Veamos cada uno de ellos en detalle.

### Archivos fuente:

Los archivos fuentes serán utilizados en tu proyecto para obtener de ellos todo lo necesario para poblar la base de datos. El proyecto deberá:

### Nivel inicial:

1. Obtener el archivo de la fuente que escojamos archivos de fuente utilizando Google Colab y la librería requests, por último debe almacenarse en forma local en formato CSV:

Librerías necesarias:

```
from datetime import datetime
import requests
import pandas as pd
from pandas import json_normalize
import json
```

### API de datos abiertos del clima:

Descripción: Algunas APIs públicas proporcionan datos históricos o en tiempo real sobre el clima, incluidas temperaturas, precipitaciones, vientos, etc.

API: <https://openweathermap.org/api>

2. Solicitud: Debes hacer una solicitud HTTP a la API para obtener los datos en formato JSON y luego convertirlos a CSV utilizando Python.

`BASE_URL = "https://api.openweathermap.org/data/2.5/weather"`

3. Organizar los archivos en rutas siguiendo la siguiente estructura:

`"data_analytics\openweather\tiempodiario_yyyymmdd.csv"`

La fecha de la nomenclatura es la fecha del tiempo tomado.

4. buscar estas ciudades:

```
cityList = ["London", "New York", "Cordoba", "Taipei", "Buenos Aires", "Mexico DF", "Dublin", "Tilfis", "Bogota", "Tokio"]
coordList = ["lat=31&lon=64", "lat=40&lon=-73", "lat=-31&lon=-64", "lat=25&lon=64", "lat=-34&lon=-58", "lat=19&lon=-99", "lat=53&lon=6", "lat=52&lon=0"]
```

5. Realizar los comentarios correspondientes para una correcta comprensión del código. (#)

## Nivel medio:

1. La descarga, procesamiento y actualización de la información en la base de datos se debe poder ejecutar desde un archivo .py
2. El proyecto debe poder deployarse en forma sencilla siguiendo un readme, que al menos contenga las instrucciones para:
  - Utilizarse creando un entorno virtual (venv)
  - Instalar las dependencias necesarias con pip.
  - Configurar la conexión a la base de datos.

Endpoint:

`BASE_URL = "https://api.openweathermap.org/data/2.5/onecall/timemachine?"`

Tomar 5 días de cada ciudad.

Normalizar Tabla teniendo en cuenta el formato y el tipo de dato contenido en cada columna.

3. Las configuraciones necesarias para que el proyecto se ejecute deben poder configurarse desde un archivo config.py
4. Realizar los comentarios correspondientes para su correcta documentación con docstrings(""" """).

## Bases de datos

Se deben dejar disponibles los scripts de creación de las tablas utilizadas.

Conexión a la base de datos:

- Los datos se deben almacenar en una base PostgreSQL.
- La conexión a la base de datos se debe implementar con la librería y ORM SQLAlchemy.
- Se recomienda ver la funcionalidad de pandas dataframe.to\_sql.

## Herramientas para el procesamiento de datos:

En esta etapa del proyecto, se utilizará la librería Pandas para procesar los datos obtenidos y almacenarlos en la base de datos PostgreSQL.

**Explicación:**

1. La librería Pandas se utilizará para leer los datos en formato CSV y procesarlos antes de cargarlos en la base de datos.

**Parte Práctica:**

Ya hemos utilizado la librería Pandas en la parte práctica anterior, donde hemos procesado los datos y los hemos cargado en la base de datos utilizando esta herramienta.