


# 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 ingresar a la aceleración.

¿Estás list@? ¡Empecemos! 

## Objetivo

Para resolver este challenge, deberás crear un proyecto que consuma datos desde 3 fuentes distintas para popular una base de datos SQL con información cultural sobre bibliotecas, museos y salas de cines argentinos.

## 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 popular la base de datos. El proyecto deberá:

- Obtener los 3 archivos de fuente utilizando la librería [requests](#) y almacenarse en forma local (Ten en cuenta que las urls pueden cambiar en un futuro):
  - [Datos Argentina - Museos](#)
  - [Datos Argentina - Salas de Cine](#)
  - [Datos Argentina - Bibliotecas Populares](#)
- Organizar los archivos en rutas siguiendo la siguiente estructura:  
“categoría\ año-mes\ categoria-día-mes-año.csv”
  - Por ejemplo: “museos\2021-noviembre\ museos-03-11-2021”
  - Si el archivo existe debe reemplazarse. La fecha de la nomenclatura es la fecha de descarga.

### Procesamiento de datos

El procesamiento de datos permitirá a nuestro proyecto transformar los datos de los archivos fuente en la información que va a nutrir la base de datos. Para esto, el proyecto deberá:

- Normalizar toda la información de Museos, Salas de Cine y Bibliotecas Populares, para crear una única tabla que contenga:
  - o cod\_localidad
  - o id\_provincia
  - o id\_departamento
  - o categoría
  - o provincia
  - o localidad
  - o nombre
  - o domicilio
  - o código postal
  - o número de teléfono
  - o mail
  - o web
- Procesar los datos conjuntos para poder generar una tabla con la siguiente información:
  - o Cantidad de registros totales por categoría
  - o Cantidad de registros totales por fuente
  - o Cantidad de registros por provincia y categoría
- Procesar la información de cines para poder crear una tabla que contenga:
  - o Provincia
  - o Cantidad de pantallas
  - o Cantidad de butacas
  - o Cantidad de espacios INCAA

## Creación de tablas en la Base de datos

Para disponibilizar la información obtenida y procesada en los pasos previos, tu proyecto deberá tener una base de datos que cumpla con los siguientes requisitos:

- La base de datos debe ser PostgreSQL
- Se deben crear los scripts .sql para la creación de las tablas.
- Se debe crear un script .py que ejecute los scripts .sql para facilitar el deploy.
- Los datos de la conexión deben poder configurarse fácilmente para facilitar el deploy en un nuevo ambiente de ser necesario.

## Actualización de la base de datos

Luego de normalizar la información y generar las demás tablas, las mismas se deben actualizar en la base de datos. Para eso, es importante tener en cuenta que:

- Todos los registros existentes deben ser reemplazados por la nueva información.
- Dentro de cada tabla debe indicarse en una columna adicional la fecha de carga.

- Los registros para los cuales la fuente no brinda información deben cargarse como nulos.



## Requerimientos técnicos

Tu aplicación deberá cumplir con una serie de requerimientos técnicos que giran en torno a 7 ejes centrales. Veamos cada uno de ellos en detalle.

### Ejecución

La descarga, procesamiento y actualización de la información en la base de datos se debe poder ejecutar desde un archivo .py

### Deploy

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.



### Configuración

Las configuraciones necesarias para que el proyecto se ejecute deben poder configurarse desde un archivo. env, .ini o similar con la librería [Python-decouple](#).

### Logs

El programa debe crear logs oportunos sobre la ejecución del mismo con la librería [Logging](#).

### 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

Utilizar la librería Pandas para procesar todos los datos que sean necesarios.



## Criterios a evaluar

A la hora de evaluar tu challenge, tendremos en cuenta una serie de criterios que nos permitirán analizar con mayor detalle el producto alcanzado. Estos son:

- Implementación de buenas prácticas de codificación y estilo de código (según [PEP8](#)).
- Comentarios oportunos y docstrings descriptivos.
- Manejo de excepciones preciso, no azaroso.
- La estructura del proyecto debe ser limpia y ordenada.
- El código deberá estar modularizado en componentes reutilizables e independientes.