

Data Engineering

Trabajo final integrador



Objetivos

- Construir un trabajo integrador que abarque las principales responsabilidades en ingeniería de datos: extracción, almacenamiento y procesamiento de datos



Consigna

Parte 1

Desarrollar un programa en Python que realice:

1. **extracción** de una **API**, como fuente de datos,
2. convierta los datos obtenidos como DataFrames de Pandas
3. y los guarde de forma **cruda**, sin transformaciones o con leves transformaciones, en formato Delta lake.

Deberás usar la librería requests para obtener datos de **2 o más endpoints** de la misma API. **Al menos uno** de los endpoints debe devolver datos temporales, que se actualicen periódicamente (mínimo una vez al día), como por ejemplo: valores meteorológicos, cotizaciones de monedas o acciones de compañías, variaciones de índices económicos, estadísticas deportivas, etc. Los demás endpoints pueden ser datos estáticos o metadatos, como por ejemplo campos que describen a una estación meteorológica (nombre, coordenadas, ciudad, etc.).

Deberás realizar una extracción incremental y una full, según corresponda.

Además tendrás que guardar cada DataFrame en formato Delta lake, cada uno en un directorio específico, como si fuese que estás trabajando en un **data lake**.

- En caso de que estés haciendo una extracción incremental, se recomienda particionar por cada fecha y también por hora (si corresponde).

- En el caso de datos relativamente estáticos, puedes particionar, o no, por algún otro campo, si consideras necesario.

Para más información podés revisar la consigna de la entrega parcial nro 1.

Parte 2

Leer los datos almacenados en la parte 1 y aplicar tareas de procesamiento o transformación de datos con Pandas. Esas tareas de procesamiento pueden ser:

- Eliminación de duplicados
- Eliminación o reemplazo de nulos
- Conversión de tipos de datos de columnas
- Renombrar columnas
- Formatear columnas de tipo fecha.
- Crear nuevas columnas a partir de alguna lógica (Por ejemplo, una columna booleana que indique si una temperatura está por arriba de un límite)
- Cruzar dataframes usando JOINS
- Aplicar agregaciones por medio de GROUP BY y funciones como MAX, MIN, AVG, etc.
- etc.

Deberás realizar al menos 4 tareas de transformación.

El resultado del procesamiento debe ser guardado en uno, o varios, archivos Delta lake en directorio que corresponda.

Para más información podés revisar la consigna de la entrega parcial nro 2.

Formato de presentación:

- Jupyter notebook (archivo .ipynb) o archivo Python (.py)
- Renombrar los archivos a entregar con nombre y apellido, seguido del nombre que consideren necesario. Por ejemplo: GuidoFranco_transformaciones.py.
- **Se recomienda entregar una carpeta comprimida (en formato .zip por ejemplo)**, que contenga los archivos a entregar. La carpeta comprimida debe estar renombrada con su apellido y nombre.

Criterios de evaluación

Se evaluará por medio de una rúbrica de evaluación.

La calificación será numérica, la calificación máxima es de 100.

Para aprobar, se requiere obtener una nota mayor o igual a 70.