Prueba Técnica:

Se solicita la construcción del siguiente flujo de datos que debe ser alojado en un repositorio Github público donde se debe tener en cuenta el uso de estándares de programación, calidad de datos y uso de librerías en el lenguaje de programación Pyspark.

Inicio de la prueba

Trabajas en un equipo de ingeniería de datos que maneja ETLs para múltiples clientes y entornos (develop, qa, main). El equipo quiere una forma robusta y flexible de poder procesar datos que no se procesaron en el pasado en los primeros 6 meses del año.

Implementar una solución de procesamiento de datos parametrizable por fechas usando **OmegaConf**, teniendo como partición la **fecha_proceso** indicada en el csv. Cabe resaltar que se deberá indicar un rango de fechas al momento del lanzamiento del programa.

Tienes que construir un flujo de datos que:

1. Lea un archivo de datos CSV incluido en el correo.
2. Usa la configuración para filtrar por un **rango de fechas configurable** (start_date, end_date) y con ello poder registrar las entregas de producto en una partición respectiva.
3. Genera las salidas pertinentes a las fechas encontradas en el dataset (output_path: data/processed/${**fecha_proceso**}).
4. Utiliza **OmegaConf** para controlar todos los parámetros del flujo desde YAML.
5. Encontrarás diferentes registros de países en el dataset, hacerlo parametrizable para que en la ejecución se pueda llamar en un **rango de fechas y con un solo país**.
6. Encontrarás una columna denominada **unidad** donde CS representa cajas y ST unidades, y para los productos mostrados CS representa 20 unidades. En el dataset de salida se necesita a los productos en una misma unidad.
7. Encontrarás la columna de **tipo_entrega** donde únicamente los valores de ZPRE y ZVE1 representan las entregas de rutina. Mientras que los valores de Z04 y Z05 son para entregas con bonificaciones. Agregar una columna por cada tipo de entrega identificada. El resto de los valores no deben ser considerados en el output.
8. Proponer un estándar correcto en el nombre de las columnas del dataset final.
9. Detectar/eliminar cualquier anomalía encontrada en la data.
10. Puntos extras por columnas adicionales con fundamento
11. Puntos extras por documentar la construcción del flujo de datos gráfica y descriptivamente.

Lo que se revisará:

Repositorio con el código propuesto junto con su resultado.