# Prueba técnica para Ingeniería de Datos

## Planteamiento del reto:

En este reto de ingeniería de datos, queremos que nos demuestres un poco de tus habilidades en el manejo de Pipelines de datos, y cómo integras la captura, procesamiento y consulta de información. Para este caso queremos que crees un pipeline de datos, que permita obtener información de los datos que vas cargando en **Micro Batches**.

En Big data es común encontrar que las cargas de Batch no son suficientes ya que la información se necesita en real time (o near real time), y/o las capacidades de cómputo de tu organización no tienen el suficiente poder para hacer cargas masivas de toda la data. Es entonces cuando es necesario implementar la conocida estrategia "divide y vencerás", y hacemos uso de la ingesta de Micro Batches, es decir, tomar una parte de la data, procesarla, disponerla y repetir el proceso con una nueva fracción de la data faltante. Esto es útil porque no es necesario hacer uso exhaustivo del poder computacional, ya que no toda la data se carga en memoria al mismo tiempo.

## Especificación de la data:

Tenemos un conjunto de cinco archivos en formato TXT, llamado dataset-1, dataset-2, ..., Todos contienen únicamente cinco columnas: ID. Año. Destino. Estrato. Consumo.

## Requerimientos:

1. Descarga la carpeta comprimida que contiene los datos y déjalos en una carpeta exclusiva para este reto.
2. Construye un Pipeline que sea capaz de:
    a. Cargar todos los archivos .TXT (El pipeline no debe contener todo el conjunto de datos, es decir, los cinco archivos al mismo tiempo en memoria en cualquier momento).
    b. Cada Micro Batch debe contener como **máximo 1000 registros**.
    c. Almacena los datos de los archivos .TXT en una base de datos de tu elección (ejemplo: PostgreSQL, MySQL, MongoDB, etc). El diseño de esta base de datos dependerá de ti, crea la tabla o tablas que creas necesarias con el esquema que creas es adecuado, pero ten presente que todos los .TXT deben ir en la misma base de datos.
    d. Calcular la tasa de impuestos al consumo teniendo en cuenta:
        i. Tarifa sobre el consumo para calcular la tasa: tarifa_por_destino.csv
        ii. **Cálculo de la tasa: Tarifa sobre consumo * Consumo = Tasa Calculada**
        iii. Valores mínimos aceptados: minimos.csv

iv. Valores máximos aceptados: maximos.csv
v. **La Tasa Calculada por registro debe modificarse según las reglas de las tablas de mínimos y máximos.**

e. A medida que los datos son cargado, realiza un seguimiento de:
i. Recuento (COUNT) de filas cargadas a la base de datos.
ii. Sumatoria (SUM) de la tasa calculada.

**NOTA**: Se espera que en la ejecución de este pipeline, al menos después de que se cargue cada .TXT (pero idealmente después de la inserción de cada fila), estas estadísticas se actualicen para reflejar los nuevos datos. Las actualizaciones del Pipeline **NO** deben tocar los datos ya cargados, es decir, hacer una consulta para traer los valores esperados para cada actualización no es una buena solución al problema.

**Comprobante de resultados:**

1. Imprime el valor actual de la tasa en ejecución.
2. Realiza una consulta en la base de datos de: recuento total de filas, valor promedio, sumatoria, valor mínimo y valor máximo para el campo calculado "Tasa".

**Algunas reglas y consideraciones del reto:**

- Puedes utilizar cualquier Framework o librería que desees.
- Puedes utilizar cualquier base de datos que desees, bien sea SQL o NoSQL, lo importante es que muestres cómo te conectas a ella, cómo poblas la(s) tabla(s) y como realizas las consultas.
- Puedes hacer uso de alguna interfaz gráfica para administrar/manipular tu base de datos (ejemplo PgAdmin), o puedes hacer uso de línea de comandos.
- Puedes usar cualquier código existente que tengas a disposición.
- No hay una forma definida de resolver esta tarea, queremos ver la forma en la que piensas para resolver un problema así.
- Las estadísticas se pueden almacenar de la forma que desees: en base de datos, en memoria, en un archivo.
- El objetivo es una solución funcional, sin embargo, le damos alto valor al rendimiento.