Contexto: Has sido contratado como Ingeniero de Datos para una gran organización que gestiona una masiva base de datos de empleados. Tu tarea es analizar y extraer información, procesar datos de manera eficiente y construir flujos de trabajo automatizados utilizando SQL, Python, Spark y Airflow.

Parte 1: Análisis de datos en SQL

1. **Progresión de Carrera de los Empleados:**
    - Para cada empleado, identificar cuántos títulos distintos tuvo y el tiempo promedio transcurrido en cada título.
2. **Tasa de Rotación por Departamento:**
    - Calcular la tasa de rotación para cada departamento (es decir, la proporción de empleados que dejaron el departamento a lo largo del tiempo).
3. **Tendencias Salariales:**
    - Determinar la progresión del salario promedio a lo largo del tiempo, desglosado por título y departamento.
4. **Empleados con Mayor Antigüedad:**
    - Identificar a los 10 empleados con mayor antigüedad, incluyendo su departamento y título más recientes.
5. **Impacto Gerencial:**
    - Calcular el tiempo promedio que los empleados permanecen en un departamento bajo cada gerente e identificar al gerente con la tasa de retención más baja y más alta.
6. **Proporción de Género por Departamento:**
    - Calcular la proporción de género para cada departamento y determinar cuáles son los departamentos con mayor disparidad.

Parte 2: Procesamiento de Datos

1. **Extracción y Limpieza de Datos:**
    - Cargar todas las tablas en DataFrames de Spark.
    - Realizar la limpieza de los datos gestionando valores nulos, asegurando la consistencia de los tipos de datos y manejando registros duplicados si los hubiera.
2. **Análisis de datos:**
    - Calcular los costos anuales de la empresa por el pago de salarios a empleados, desglosado por departamento.
    - Identificar empleados que han cambiado de departamento más de dos veces.
3. **Generación de Reportes:**
    - Generar un informe en formato Parquet con los siguientes campos:
        - *emp_no*
        - *full_name*
        - *current_department*
        - *current_title*
        - *current_salary*
        - *hire_date*
        - *tenure_years* (antigüedad en años)
    - Generar un informe en formato CSV con el resumen de los cambios en la base de empleados entre dos fechas definidas
        - La estructura de este informe deberá ser definida por el candidato
        - Considerar que debe contemplar cambios en diferentes entidades. Por ejemplo:
            1. El empleado *emp_no* cambió de posición/departamento
            2. El empleado *emp_no* tuvo un aumento de salario
            3. El departamento *dept_no* cambió de manager
4. **Automatización del pipeline:**
    - Diseñar un DAG de Airflow que realice los siguientes pasos:
        - Conectarse a la base de datos MySQL.
        - Calcular los cambios que hubo en la base de empleados entre dos fechas definidas y genera un reporte de resumen de los mismos (es el mismo reporte del punto anterior)
        - Almacenar los resultados en un archivo CSV en un bucket definido de s3 .
    - **Programación y Monitoreo del DAG:**
        - Asegúrate de que el DAG se ejecute el primer día de cada mes a las 3:00 AM y genere el reporte de los movimientos/cambios del último mes.
        - Implementar un sistema de registro de errores y alertas (por ejemplo, por correo electrónico) en caso de fallo de una tarea.

Entregables

1. **Código:**
    - Proporcionar un repositorio de Git con:
        - Scripts SQL para las consultas.
        - Scripts en Python y PySpark para la extracción y transformación de datos
        - Código del DAG de Airflow con instrucciones de configuración.
2. **Documentación:**
    - Incluye un `README.md` que explique:
        - Instrucciones de configuración y ejecución.
        - Razonamiento y suposiciones detrás de las decisiones tomadas.
3. **Informes:**
    - Informes en formato CSV y Parquet generados en las secciones de Python y PySpark. (Si el informe completo queda con un tamaño muy grande, puede ser un sample de los mismos)
    - Evidencia de la ejecución del DAG de Airflow.