

## Ejercicio Clase 9

Consigna: Por cada ejercicio, escribir el código y agregar una captura de pantalla del resultado obtenido.

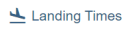
1. Crear una base de datos en Hive llamada northwind\_analytics
2. Crear un script para importar un archivo .parquet de la base northwind que contenga la lista de clientes junto a la cantidad de productos vendidos ordenados de mayor a menor (campos customer\_id, company\_name, productos\_vendidos). Luego ingestar el archivo a HDFS (carpeta /sqoop/ingest/clientes). Pasar la password en un archivo
3. Crear un script para importar un archivo .parquet de la base northwind que contenga la lista de órdenes junto a qué empresa realizó cada pedido (campos order\_id, shipped\_date, company\_name, phone). Luego ingestar el archivo a HDFS (carpeta /sqoop/ingest/envíos). Pasar la password en un archivo
4. Crear un script para importar un archivo .parquet de la base northwind que contenga la lista de detalles de órdenes (campos order\_id, unit\_price, quantity, discount). Luego ingestar el archivo a HDFS (carpeta /sqoop/ingest/order\_details). Pasar la password en un archivo
5. Generar un archivo .py que permita mediante Spark insertar en hive en la db northwind\_analytics en la tabla products\_sold, los datos del punto 1, pero solamente aquellas compañías en las que la cantidad de productos vendidos fue mayor al promedio.
6. Generar un archivo .py que permita mediante Spark insertar en hive en la tabla products\_sent, los datos del punto 2 y 3, de manera tal que se vean las columnas order\_id, shipped\_date, company\_name, phone, unit\_price\_discount (unit\_price with discount), quantity, total\_price (unit\_price\_discount \* quantity). Solo de aquellos pedidos que hayan tenido descuento.
7. Realizar un proceso automático en Airflow que orqueste los pipelines creados en los puntos anteriores. Crear un grupo para la etapa de ingest y otro para la etapa de process. Correrlo y mostrar una captura de pantalla (del DAG y del resultado en la base de datos)

## DAG: ejercicio9

running

Schedule: @daily

Next Run: 2023-05-01, 00:00:00



2023-04-30T00:00:01Z

Runs

25

Run

scheduled\_\_2023-04-30T00:00:00+00:00

Layout

Left &gt; Right

Find Task...

Update

BashOperator

DummyOperator

queued

running

success

failed

up\_for\_retry

up\_for\_reschedule

upstream\_failed

skipped

scheduled

deferred

no\_status

Auto-refresh

