

Ejercicio Clase 9

1. Crear una base de datos en Hive llamada northwind_analytics

```
hive> create database northwind_analytics;
OK
Time taken: 0.347 seconds
hive> show databases;
OK
default
fl
northwind_analytics
tripdata
Time taken: 0.037 seconds, Fetched: 4 row(s)
hive>
```

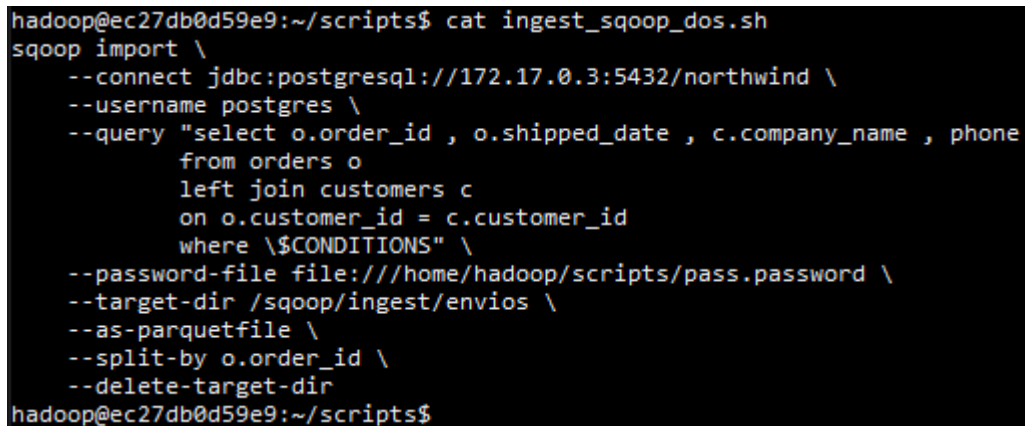
2. Crear un script para importar un archivo .parquet de la base northwind que contenga la lista de clientes junto a la cantidad de productos vendidos ordenados de mayor a menor (campos customer_id, company_name, productos_vendidos). Luego ingestar el archivo a HDFS (carpeta /sqoop/ingest/clientes). Pasar la password en un archivo.

```
sqoop import \
-Dorg.apache.sqoop.splitter.allow_text_splitter=true \
--connect jdbc:postgresql://172.17.0.3:5432/northwind \
--username postgres \
--query "select c.customer_id, c.company_name, SUM(od.product_id) as
productos_vendidos from customers c
left join orders o on c.customer_id = o.customer_id
left join order_details od on o.order_id = od.order_id
group by c.customer_id, c.company_name
having \${CONDITIONS}
order by productos_vendidos desc" \
--password-file file:///home/hadoop/scripts/pass.password \
--target-dir /sqoop/ingest/clientes \
--as-parquetfile \
--split-by c.customer_id \
--delete-target-dir
```

```
hadoop@ec27db0d59e9:~/scripts$ cat > sqoop_ingest.sh
sqoop import \
-Dorg.apache.sqoop.splitter.allow_text_splitter=true \
--connect jdbc:postgresql://172.17.0.3:5432/northwind \
--username postgres \
--query "select c.customer_id, c.company_name, SUM(od.product_id) as productos_vendidos from customers c
left join orders o on c.customer_id = o.customer_id
left join order_details od on o.order_id = od.order_id
group by c.customer_id, c.company_name
having \${CONDITIONS}
order by productos_vendidos desc" \
--password-file file:///home/hadoop/scripts/pass.password \
--target-dir /sqoop/ingest/clientes \
--as-parquetfile \
--split-by c.customer_id \
--delete-target-dir
hadoop@ec27db0d59e9:~/scripts$ cat > sqoop_ingest.sh
```

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.

```
sqoop import \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres \  
--query "select o.order_id , o.shipped_date , c.company_name , phone  
        from orders o  
        left join customers c  
        on o.customer_id = c.customer_id  
        where \${CONDITIONS}" \  
--password-file file:///home/hadoop/scripts/pass.password \  
--target-dir /sqoop/ingest/envios \  
--as-parquetfile \  
--split-by o.order_id \  
--delete-target-dir
```



```
hadoop@ec27db0d59e9:~/scripts$ cat ingest_sqoop_dos.sh  
sqoop import \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres \  
--query "select o.order_id , o.shipped_date , c.company_name , phone  
        from orders o  
        left join customers c  
        on o.customer_id = c.customer_id  
        where \${CONDITIONS}" \  
--password-file file:///home/hadoop/scripts/pass.password \  
--target-dir /sqoop/ingest/envios \  
--as-parquetfile \  
--split-by o.order_id \  
--delete-target-dir  
hadoop@ec27db0d59e9:~/scripts$
```

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(carpetasqoop/ingest/order_details). Pasar la password en un archivo.

```
sqoop import \  
--connect jdbc:postgresql://172.17.0.3:5432/northwind \  
--username postgres \  
--query "select od.order_id , od.unit_price , od.quantity , od.discount  
        from order_details od  
        where \${CONDITIONS}" \  
--password-file file:///home/hadoop/scripts/pass.password \  
--target-dir /sqoop/ingest/orders_details \  
--as-parquetfile \  
--split-by od.order_id \  
--delete-target-dir
```

```

hadoop@ec27db0d59e9:~/scripts$ cat sqoop_ingest_tres.sh
sqoop import \
  --connect jdbc:postgresql://172.17.0.3:5432/northwind \
  --username postgres \
  --query "select od.order_id , od.unit_price , od.quantity , od.discount
          from order_details od
          where \$CONDITIONS" \
  --password-file file:///home/hadoop/scripts/pass.password \
  --target-dir /sqoop/ingest/orders_details \
  --as-parquetfile \
  --split-by od.order_id \
  --delete-target-dir
hadoop@ec27db0d59e9:~/scripts$

```

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 2, pero solamente aquellas compañías en las que la cantidad de productos vendidos fue mayor al promedio.

```

CREATE EXTERNAL TABLE northwind_analytics.products_sold(customer_id
string, company_name string, productos_vendidos int)
ROW FORMAT DELIMITED
FIELDS TERMINATED BY ','
LOCATION '/tables/external/northwind_analytics/products_sold';

```

```

hive> show tables;
OK
products_sold
Time taken: 0.08 seconds, Fetched: 1 row(s)

```

```

1 from pyspark.context import SparkContext
2 from pyspark.sql.session import SparkSession
3 from pyspark.sql import HiveContext
4 sc = SparkContext('local')
5 spark = SparkSession(sc)
6 hc = HiveContext(sc)
7
8 ##leo parquet de HDFS y lo cargo en un dataframe
9 df = spark.read.parquet("hdfs://172.17.0.2:9000/sqoop/ingest/clientes/*.parquet")
10
11 ##creamos una vista del DF
12 df.createOrReplaceTempView("clientes_productos")
13
14 ##filtramos el DF para obtener aquellas compañías en las que la cantidad de productos vendidos fue mayor al promedio.
15 df_products = spark.sql("select customer_id, company_name, cast(productos_vendidos as int) from clientes_productos where productos_vendidos > (select AVG(cast(productos_vendidos as
16 int)) from clientes_productos) order by productos_vendidos desc")
17
18 df_products.createOrReplaceTempView("products_filtrados")
19
20 ##insertamos el DF en la tabla northwind_analytics.products_sold
21 spark.sql("insert into northwind_analytics.products_sold select * from products_filtrados")

```

6. Generar un archivo .py que permita mediante Spark insertar en hive en la tabla products_sent, los datos del punto 3 y 4, 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.

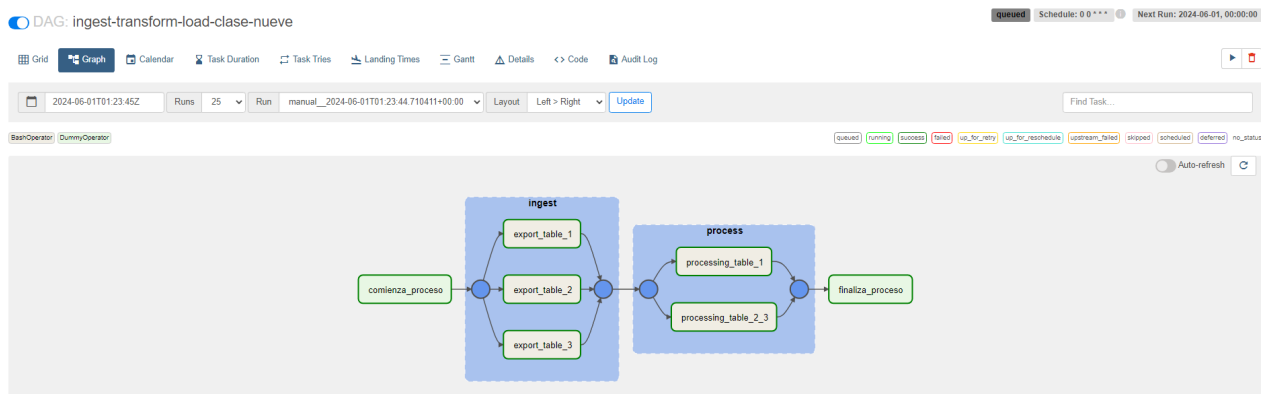
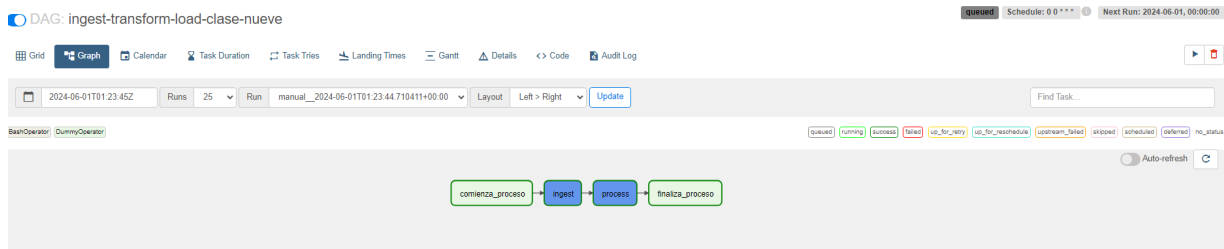
```
CREATE EXTERNAL TABLE northwind_analytics.products_sent(order_id int,  
shipped_date date, company_name string, phone string, unit_price_discount  
double, quantity int, total_price double)  
  
ROW FORMAT DELIMITED  
  
FIELDS TERMINATED BY ','  
  
LOCATION '/tables/external/northwind_analytics/products_sent';
```

```
hive> show tables;  
OK  
products_sent  
products_sold  
Time taken: 0.051 seconds, Fetched: 2 row(s)  
hive>
```

```
1 from pyspark.context import SparkContext  
2 from pyspark.sql.session import SparkSession  
3 from pyspark.sql import HiveContext  
4 from pyspark.sql.functions import from_unixtime, to_date, col  
5 sc = SparkContext('local')  
6 spark = SparkSession(sc)  
7 hc = HiveContext(sc)  
8  
9 ##leo parquet de HDFS y lo cargo en un dataframe  
10 df_envios = spark.read.parquet("hdfs://172.17.0.2:9000/sqoop/ingest/envios/*.parquet")  
11 df_orders = spark.read.parquet("hdfs://172.17.0.2:9000/sqoop/ingest/orders_details/*.parquet")  
12  
13 #pasamos el campo shipped_date a tipo date  
14 df_envios = df_envios.withColumn("shipped_date", to_date(from_unixtime(col("shipped_date")/1000)))  
15  
16 ##creamos una vista del DF  
17 df_envios.createOrReplaceTempView("envios")  
18 df_orders.createOrReplaceTempView("orders")  
19  
20 ##filtramos los df para solo de aquellos pedidos que hayan tenido descuento  
21 df_envios = spark.sql("select order_id as order_id_env, shipped_date, company_name, phone from envios")  
22 df_envios.createOrReplaceTempView("envios_cambio_campo")  
23  
24 df_join = spark.sql("select * from envios_cambio_campo e left join orders o on e.order_id_env = o.order_id where o.discount != 0")  
25  
26 df_join.createOrReplaceTempView("df_joined")  
27  
28 df_descuentos = spark.sql("select order_id, shipped_date, company_name, phone, (unit_price-(unit_price*discount)/100) as unit_price_discount, quantity, round((unit_price-  
29 (unit_price*discount)/100)*quantity,2) as total_price from df_joined")  
30 df_descuentos.createOrReplaceTempView("descuentos_filtrados")  
31  
32 ##insertamos el DF en la tabla northwind_analytics.products_sent  
33 spark.sql("insert into northwind_analytics.products_sent select * from descuentos_filtrados")
```

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)

```
1 from datetime import timedelta
2 from airflow import DAG
3 from airflow.operators.bash import BashOperator
4 from airflow.operators.dummy import DummyOperator
5 from airflow.utils.dates import days_ago
6 from airflow.utils.task_group import TaskGroup
7
8 args = {
9     'owner': 'airflow',
10 }
11
12 with DAG(
13     dag_id='ingest-transform-load-clase-nueve',
14     default_args=args,
15     schedule_interval='0 0 * * *',
16     start_date=days_ago(2),
17     dagrun_timeout=timedelta(minutes=60),
18     tags=['ingest', 'transform'],
19     params={"example_key": "example_value"},
20 ) as dag:
21
22     comienzo_proceso = DummyOperator(
23         task_id='comienzo_proceso',
24     )
25
26     finaliza_proceso = DummyOperator(
27         task_id='finaliza_proceso',
28     )
29
30     with TaskGroup('ingest', tooltip='ingest') as ingest:
31         task_1 = BashOperator(task_id='export_table_1', bash_command='export PATH=({ var.value.SQOOP_HOME }) /bin:$PATH && /usr/bin/sh /home/hadoop/scripts/ingest_sqoop.sh ')
32         task_2 = BashOperator(task_id='export_table_2', bash_command='export PATH=({ var.value.SQOOP_HOME }) /bin:$PATH && /usr/bin/sh /home/hadoop/scripts/ingest_sqoop_dos.sh ')
33         task_3 = BashOperator(task_id='export_table_3', bash_command='export PATH=({ var.value.SQOOP_HOME }) /bin:$PATH && /usr/bin/sh /home/hadoop/scripts/ingest_sqoop_tres.sh ')
34
35     with TaskGroup('process', tooltip='process') as process:
36         task_1 = BashOperator(task_id='processing_table_1', bash_command='ssh hadoop@172.17.0.2 /home/hadoop/spark/bin/spark-submit --files /home/hadoop/hive/conf/hive-site.xml /home/hadoop/scripts/transform_uno.py ')
37         task_2 = BashOperator(task_id='processing_table_2_3', bash_command='ssh hadoop@172.17.0.2 /home/hadoop/spark/bin/spark-submit --files /home/hadoop/hive/conf/hive-site.xml /home/hadoop/scripts/transform_dos.py ')
38
39     comienzo_proceso >> ingest >> process >> finaliza_proceso
40
```



select * FROM products_sold pr | Enter a SQL expression to filter results (use Ctrl+Space)

| | asc customer_id ▼ | asc company_name ▼ | 123 productos_vendidos ▼ |
|----|-------------------|------------------------------|--------------------------|
| 1 | SAVEA | Save-a-lot Markets | 4,568 |
| 2 | ERNSH | Ernst Handel | 3,947 |
| 3 | QUICK | QUICK-Stop | 3,531 |
| 4 | RATTC | Rattlesnake Canyon Grocery | 2,737 |
| 5 | HUNGO | Hungry Owl All-Night Grocers | 2,282 |
| 6 | FRANK | Frankenversand | 2,160 |
| 7 | FOLKO | Folk och få HB | 2,121 |
| 8 | HILAA | HILARION-Abastos | 2,042 |
| 9 | BERGS | Berglunds snabbköp | 2,029 |
| 10 | SUPRD | Suprêmes délices | 1,680 |
| 11 | BONAP | Bon app' | 1,599 |
| 12 | QUEEN | Queen Cozinha | 1,599 |
| 13 | WHITC | White Clover Markets | 1,552 |
| 14 | LEHMS | Lehmans Marktstand | 1,549 |
| 15 | VAFFE | Vaffeljernet | 1,447 |
| 16 | RICSU | Richter Supermarkt | 1,413 |
| 17 | WARTH | Wartian Herkku | 1,409 |
| 18 | BOTTM | Bottom-Dollar Markets | 1,377 |
| 19 | LILAS | LILA-Supermercado | 1,372 |
| 20 | AROUT | Around the Horn | 1,371 |
| 21 | KOENE | Königlich Essen | 1,366 |
| 22 | HANAR | Hanari Carnes | 1,364 |
| 23 | LAMAI | La maison d'Asie | 1,321 |
| 24 | MEREP | Mère Paillarde | 1,273 |

select * FROM products_sold pr | Enter a SQL expression to filter results (use Ctrl+Space)

| | 123 order_id ▼ | shipped_date ▼ | asc company_name ▼ | asc phone ▼ | 123 unit_price_discount ▼ | 123 quantity ▼ | 123 total_price ▼ |
|----|----------------|----------------|--------------------------------|-------------------|---------------------------|----------------|-------------------|
| 1 | 10,456 | 1997-02-28 | Königlich Essen | 0555-09876 | 15.975999999 | 21 | 335.5 |
| 2 | 10,456 | 1997-02-28 | Königlich Essen | 0555-09876 | 7.9879999995 | 40 | 319.52 |
| 3 | 10,459 | 1997-02-28 | Victuailles en stock | 78.32.54.86 | 9.5952003813 | 20 | 191.9 |
| 4 | 10,459 | 1997-02-28 | Victuailles en stock | 78.32.54.86 | 23.9879999995 | 16 | 383.81 |
| 5 | 10,460 | 1997-03-03 | Folk och få HB | 0695-34 67 21 | 6.1844998097 | 4 | 24.74 |
| 6 | 10,460 | 1997-03-03 | Folk och få HB | 0695-34 67 21 | 9.975 | 21 | 209.48 |
| 7 | 10,461 | 1997-03-05 | LILA-Supermercado | (9) 331-6954 | 19.152000761 | 60 | 1,149.12 |
| 8 | 10,461 | 1997-03-05 | LILA-Supermercado | (9) 331-6954 | 20.648250761 | 28 | 578.15 |
| 9 | 10,461 | 1997-03-05 | LILA-Supermercado | (9) 331-6954 | 7.98 | 40 | 319.2 |
| 10 | 10,464 | 1997-03-14 | Furia Bacalhau e Frutos do Mar | (1) 354-2534 | 30.3391996193 | 30 | 910.18 |
| 11 | 10,464 | 1997-03-14 | Furia Bacalhau e Frutos do Mar | (1) 354-2534 | 17.5648003793 | 16 | 281.04 |
| 12 | 10,465 | 1997-03-14 | Vaffeljernet | 86 21 32 43 | 7.5923999047 | 30 | 227.77 |
| 13 | 10,465 | 1997-03-14 | Vaffeljernet | 86 21 32 43 | 98.9009999943 | 18 | 1,780.22 |
| 14 | 10,469 | 1997-03-14 | White Clover Markets | (206) 555-4112 | 15.4767499995 | 2 | 30.95 |
| 15 | 10,469 | 1997-03-14 | White Clover Markets | (206) 555-4112 | 13.8791496181 | 35 | 485.77 |
| 16 | 10,469 | 1997-03-14 | White Clover Markets | (206) 555-4112 | 15.1771998096 | 40 | 607.09 |
| 17 | 10,472 | 1997-03-19 | Seven Seas Imports | (171) 555-1717 | 3.5981999047 | 80 | 287.86 |
| 18 | 10,475 | 1997-04-04 | Suprêmes délices | (071) 23 67 22 20 | 14.3783996177 | 42 | 603.89 |
| 19 | 10,475 | 1997-04-04 | Suprêmes délices | (071) 23 67 22 20 | 13.5796003795 | 60 | 814.78 |
| 20 | 10,475 | 1997-04-04 | Suprêmes délices | (071) 23 67 22 20 | 9.985 | 35 | 349.47 |
| 21 | 10,476 | 1997-03-24 | HILARION-Abastos | (5) 555-1340 | 19.1904007626 | 2 | 38.38 |
| 22 | 10,477 | 1997-03-25 | Princesa Isabel Vinhos | (1) 356-5634 | 14.3639996195 | 20 | 287.28 |
| 23 | 10,477 | 1997-03-25 | Princesa Isabel Vinhos | (1) 356-5634 | 7.98 | 21 | 167.58 |
| 24 | 10,478 | 1997-03-26 | Victuailles en stock | 78.32.54.86 | 24.787599237 | 20 | 495.75 |