PROYECTO INTEGRADOR

1) HDFS

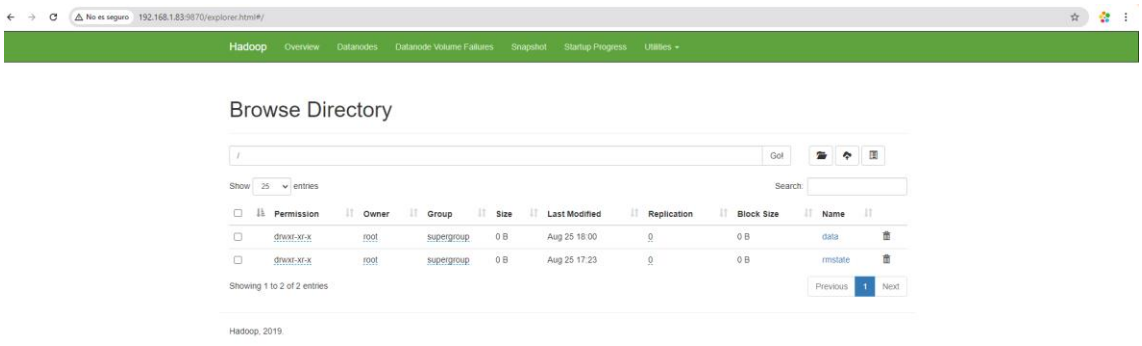
Paso 1: Ingresamos al contendor

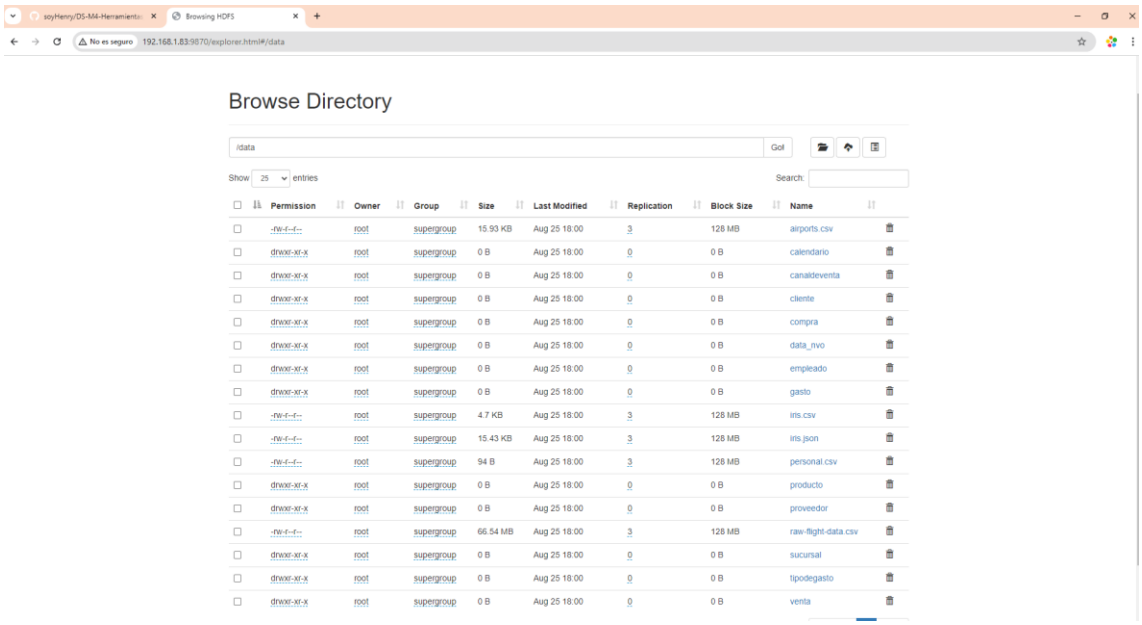
```
ubuntu@servidor_ubuntu: ~/herramientas_big_data
ubuntu@servidor_ubuntu:~/herramientas_big_data$ sudo docker exec -it namenode bash
root@2b65ee097865:/# hdfs dfs -put /home/Datasets/* /data
2024-08-25 23:00:02,985 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,272 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,340 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,410 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,473 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,562 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,617 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,694 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,723 WARN hdfs.DataStreamer: Caught exception
java.lang.InterruptedException
        at java.lang.Object.wait(Native Method)
        at java.lang.Thread.join(Thread.java:1252)
        at java.lang.Thread.join(Thread.java:1326)
        at org.apache.hadoop.hdfs.DataStreamer.closeResponder(DataStreamer.java:986)
        at org.apache.hadoop.hdfs.DataStreamer.closeInternal(DataStreamer.java:847)
        at org.apache.hadoop.hdfs.DataStreamer.run(DataStreamer.java:843)
2024-08-25 23:00:03,811 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,894 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
2024-08-25 23:00:03,956 INFO sasl.SaslDataTransferClient: SASL encryption trust check: localHost
Trusted = false, remoteHostTrusted = false
```

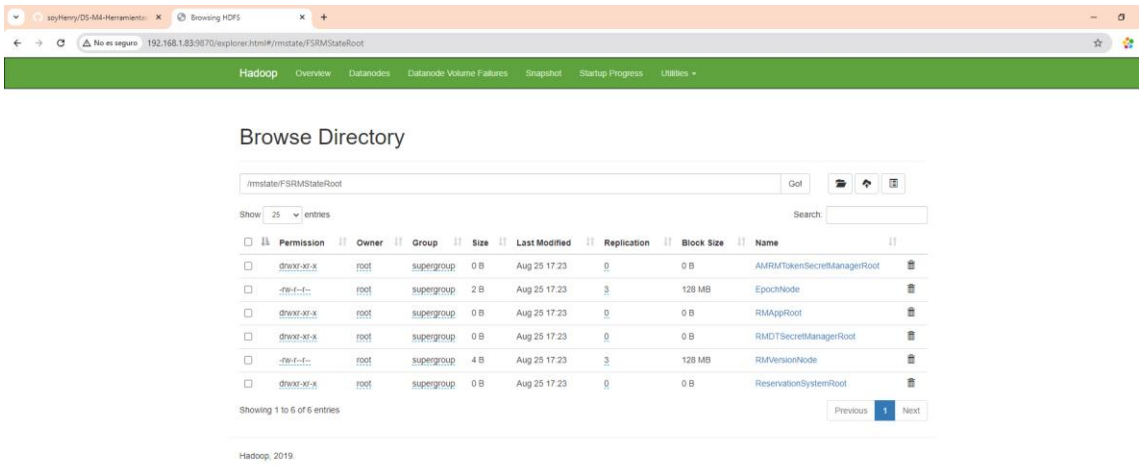
Paso 2: Se copia los Datasets en el contenedor

```
ubuntu@servidor_ubuntu:~/herramientas_big_data$ sudo docker cp ./Datasets nameno
de:/home/Datasets
ubuntu@servidor_ubuntu:~/herramientas_big_data$ ls
Datasets                          docker-compose-kafka.yml
Generacion_Ventas.ipynb           docker-compose-v1.yml
Mongo                             docker-compose-v2.yml
Parquet                           docker-compose-v3.yml
Paso00.sh                         docker-compose-v4.yml
Paso01.sh                         docker-compose.yml
Paso02.hql                        ejemploNeo4J.txt
Paso02_ConConsultas.hql           hadoop-hive.env
Paso03.hql                        hadoop.env
Paso04.hql                        hbase-distributed-local.env
Paso04_ConConsulta.hql            iris.hql
Paso05.py                         pruebaPySpark.py
Paso06_GeneracionVentasNuevasPorDia.py  pruebaScala.scala
Paso06_IncrementalVentas.py       pyspark-ETL.ipynb
README.md
ubuntu@servidor_ubuntu:~/herramientas_big_data$ sudo docker cp Paso02.hql hive-s
```

Paso 3: Utilizamos el puerto para observar los datos





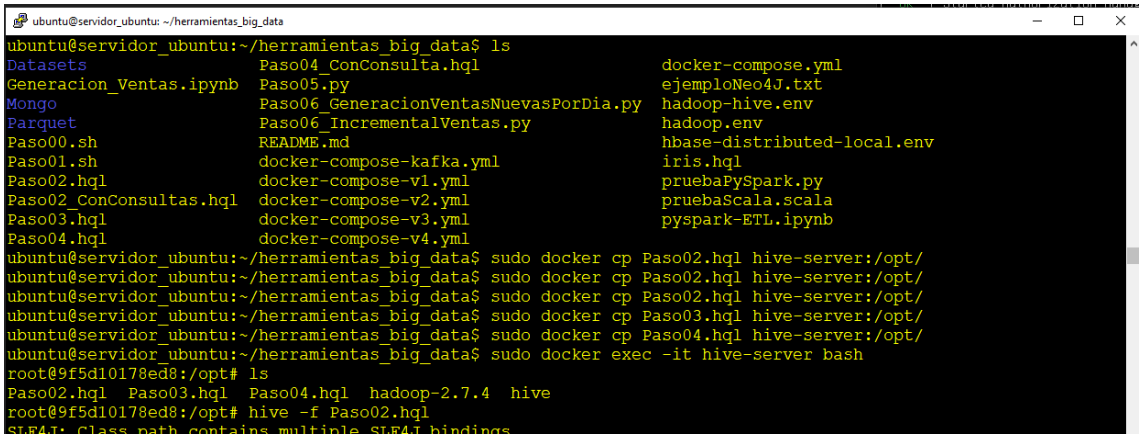


2) Hive

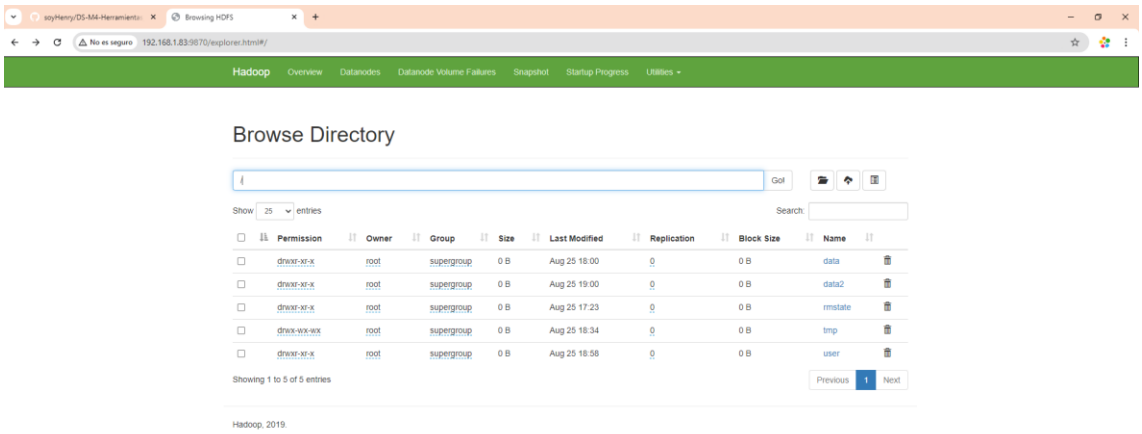
Paso 1: Utilizamos el entorno sudo docker-compose -f docker-compose-v2.yml up -d

Paso 2: Copiamos los archivos hacia el contenedor

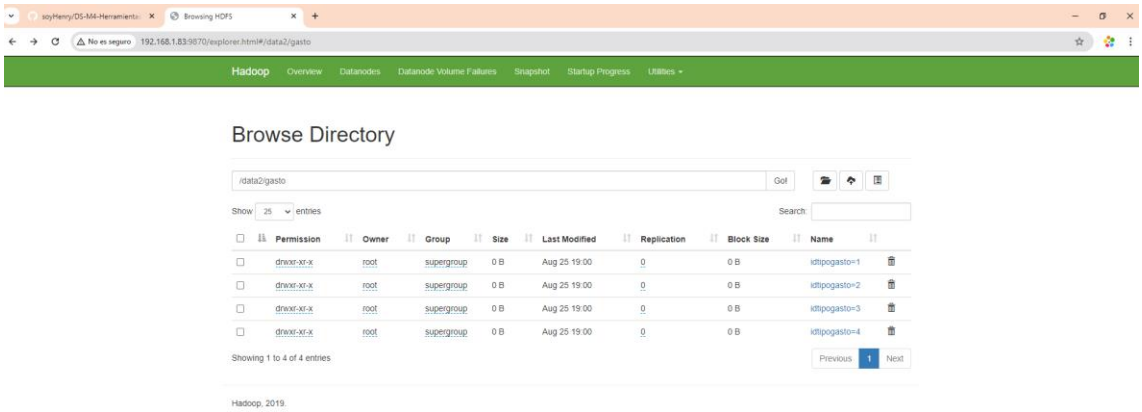
Paso 3: Ingresamos al contenedor sudo docker exec -it hive-server bash, confirmamos los archivos que se encuentran dentro del contenedor y luego ejecutamos los archivos Paso2, Paso3 y Paso 4, los cuales van a comprimir los archivos.



Paso 2: Mostramos la carpeta data2 que se han generado con los archivos comprimidos.



Se muestra que la Tabla gato se ha particionado por tipo de gasto.



3) Formatos de Almacenamiento

Paso 1: creamos un nuevo archivo con la partición de las compras Particion_compra.hql

Paso 2: como el winzip llevamos el archivo dentro de la carperta “herramientas_big_data”

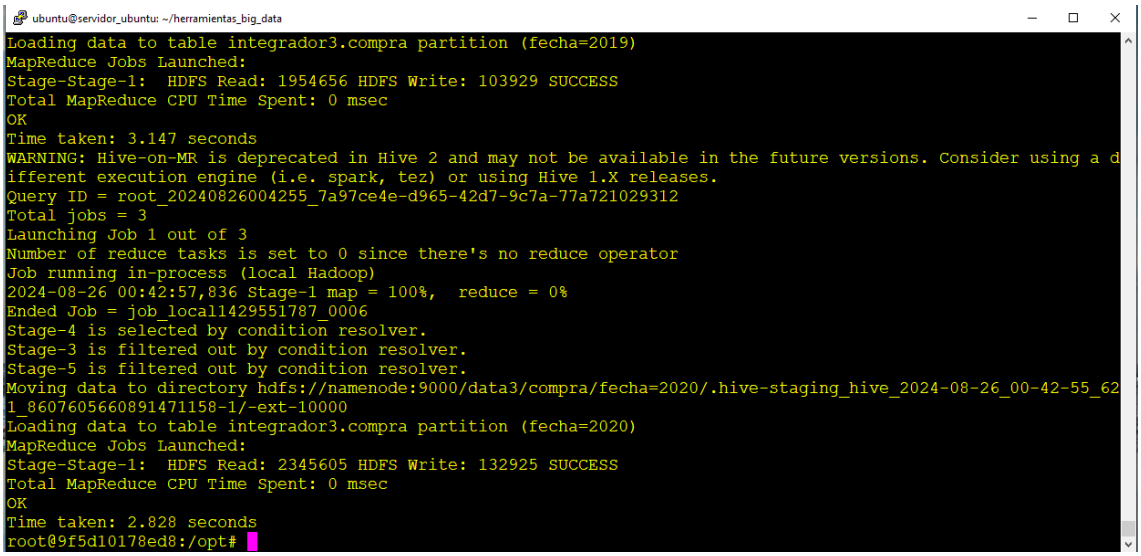
Paso 3: Luego llevamos el archivo dentro del contenedor

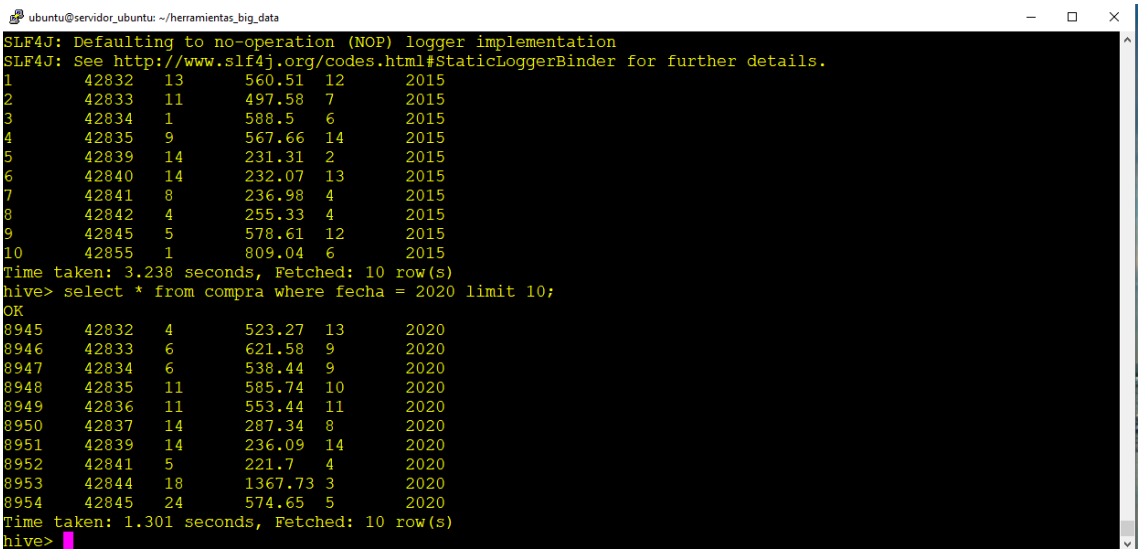
Sudo Docker cp Partición_compra.hql hive-server:/opt/

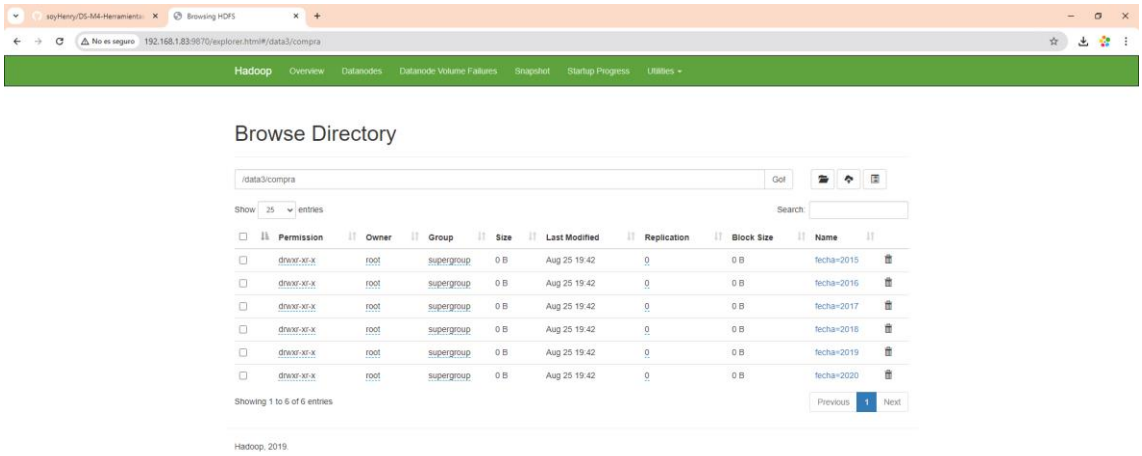
Paso 4: Ingresamos al contenedor sudo docker exec -it hive-server bash

Paso 5: Dentro del contenedor ejecutamos el archivo

root@9f5d10178ed8:/opt# hive -f Particion_compra.hql







4) SQL

Paso 1: Ingresamos al contenedor :~/herramientas_big_data$ sudo docker exec -it hive-server bash

Paso 2: dentro del contenedor ingresamos a root@9f5d10178ed8:/opt# hive

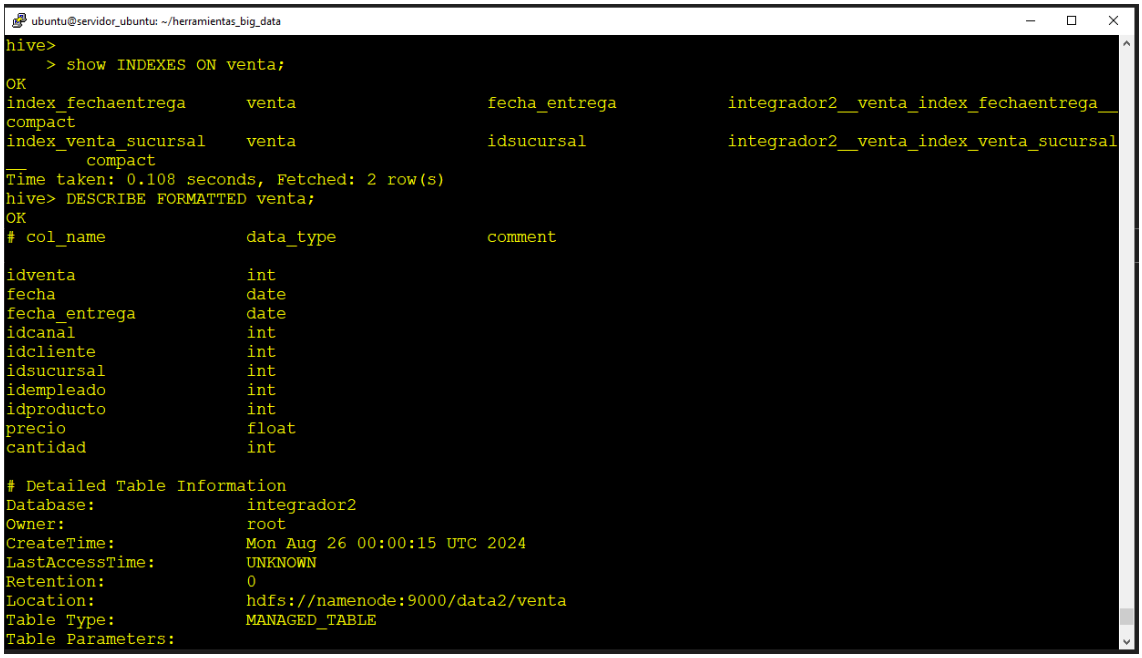
Paso 3: dentro de hive> show databases; observamos:

```
hive> show databases;
OK
default
integrador
integrador2
integrador3
Time taken: 1.283 seconds, Fetched: 4 row(s)
hive> use integrador2;
OK
Time taken: 0.049 seconds
```

Paso 3: usamos el database integrador2 para crear el índice en la tabla venta:

```
hive> use integrador2;
OK
Time taken: 0.049 seconds
hive> CREATE INDEX index_fechaentrega ON TABLE venta(Fecha_Entrega)
    >  AS 'org.apache.hadoop.hive.ql.index.compact.CompactIndexHandler'
    >  WITH DEFERRED REBUILD;
OK
Time taken: 0.389 seconds
```

Paso 4: observamos los índices de la tabla venta y la descripción de la misma.


ubuntu@servidor_ubuntu: ~/herramientas_big_data
hive>
> show INDEXES ON venta;
OK
index_fechaentrega venta fecha_entrega integrador2__venta_index_fechaentrega__
compact
index_venta_sucursal venta idsucursal integrador2__venta_index_venta_sucursal
__ compact
Time taken: 0.108 seconds, Fetched: 2 row(s)
hive> DESCRIBE FORMATTED venta;
OK
col_name data_type comment
idventa int
fecha date
fecha_entrega date
idcanal int
idcliente int
idsucursal int
idempleado int
idproducto int
precio float
cantidad int
Detailed Table Information
Database: integrador2
Owner: root
CreateTime: Mon Aug 26 00:00:15 UTC 2024
LastAccessTime: UNKNOWN
Retention: 0
Location: hdfs://namenode:9000/data2/venta
Table Type: MANAGED_TABLE
Table Parameters: