University of East London School of Architecture, Computing and Engineering (ACE)
Author: Dr Amin Karami (a.karami@uel.ac.uk) 19-20

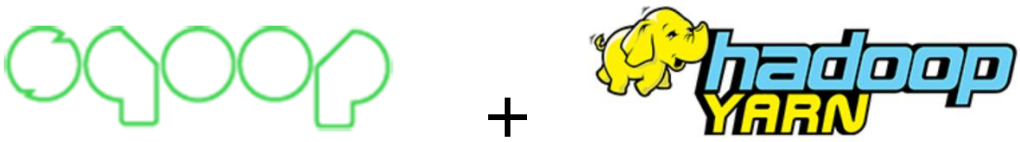
Tutorial 3: Apache Sqoop with Cloudera

CN7022 - Big Data Analytics

Dr Amin Karami (a.karami@uel.ac.uk)

LEARNING OUTCOMES: After completing this tutorial, you should:

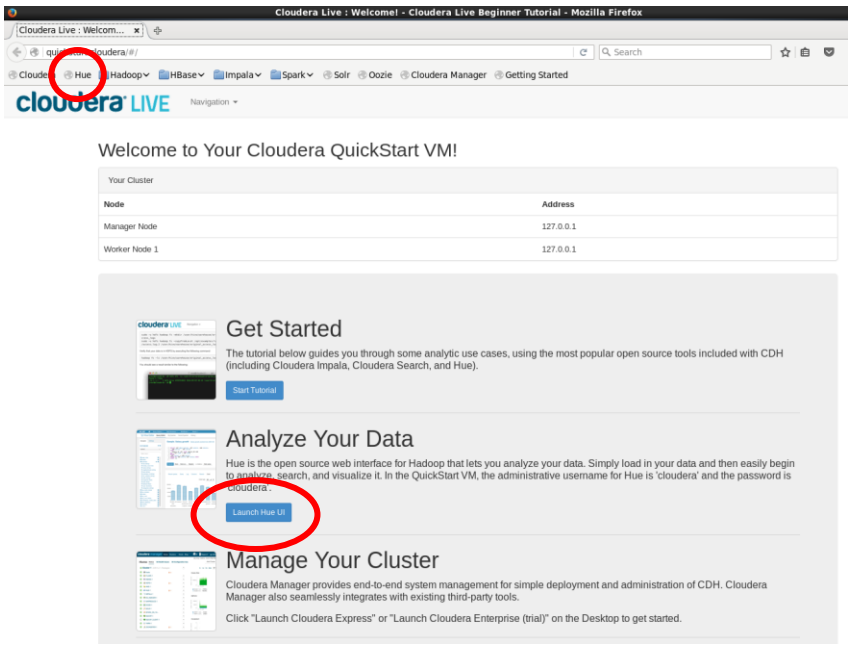
- Have gotten a hands-on experience in deploying Apache Sqoop with Cloudera
- Learn to transfer data from a relational database into Hadoop using Sqoop
- Learn hands-on practices on Sqoop commands



Phase 1: Getting Ready for Hue Cloudera

Hue is an open-source SQL Cloud Editor, licensed under the Apache v2 license, to dynamically interact and visualize big sized data.

Launch the “cloudera-quickstart-vm” from VMWare workstation. It takes a couple of minutes to load up. Then, find and click on **Hue** icon in the browser from one of the two ways as shown in the following screenshot.

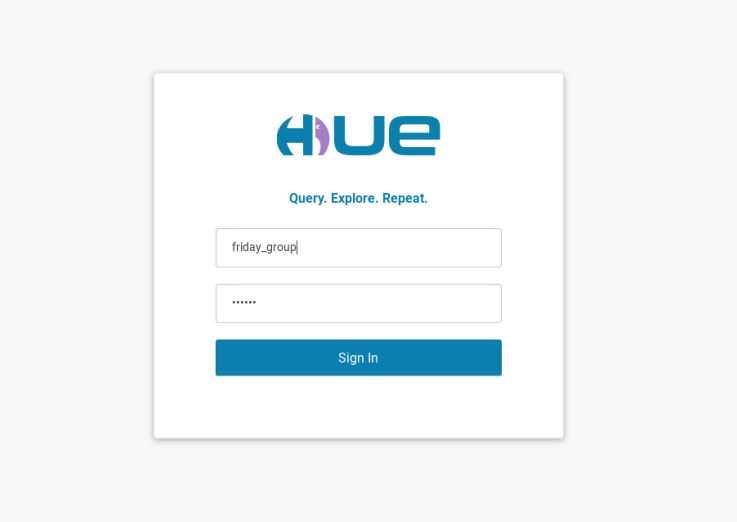


Then, enter your username/password:

Username: **friday_group** Password: **friday [default]**

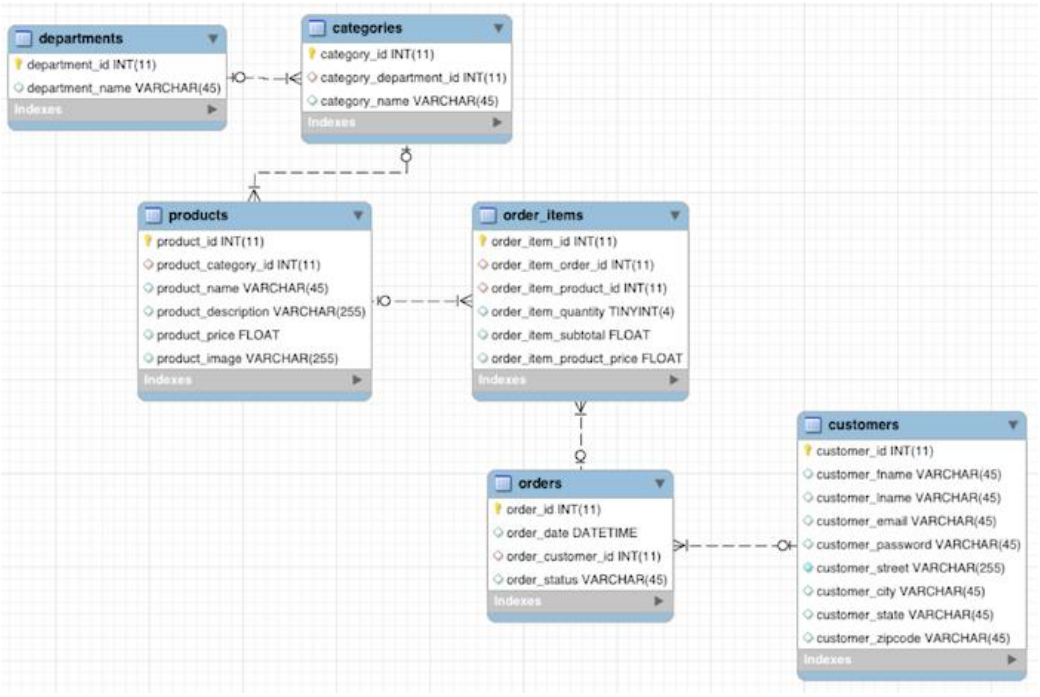
Username: **tuesday_group** Password: **tuesday**

Username: **wednesday_group** Password: **wednesday**



Phase 2: Working with Sqoop

The flowchart of `retail_db` is in the following picture. We are going to import it into the Hadoop with Sqoop. This database already exists in the Cloudera. Let us check the databse. Source: https://www.cloudera.com/developers/get-started-with-hadoop-tutorial/exercise-1.html



a) Open `cmd`. Login to MySQL Database Server (password: `cloudera`).

```
mysql -u root -p
```

b) Show all the databases and the tables.

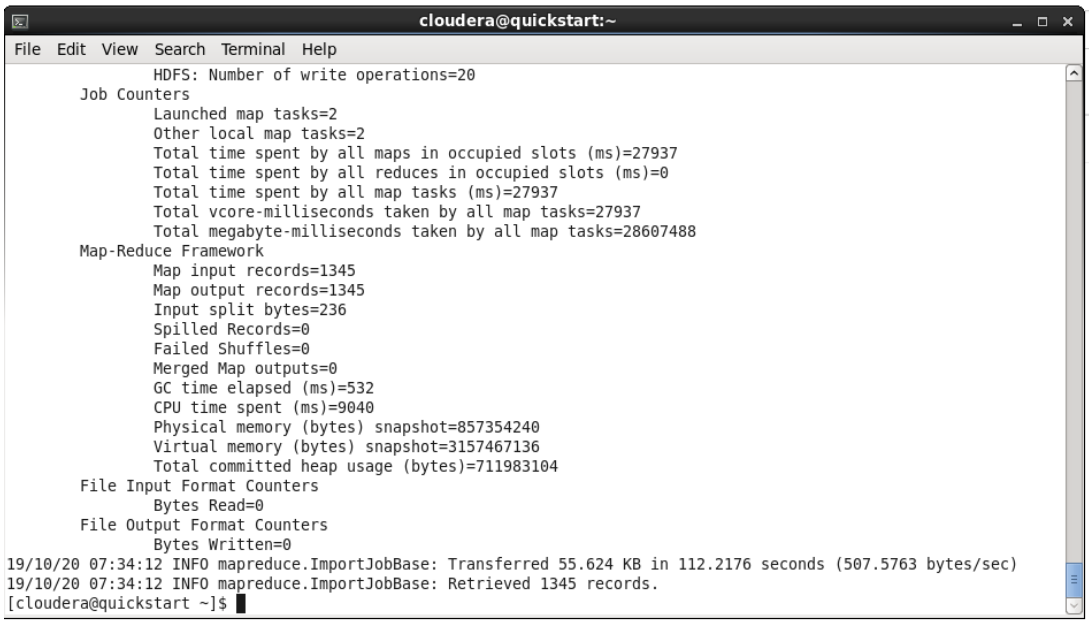
```
show databases;
use retail_db;
show tables;
quit
```

```
+--------------------+
| Tables_in_retail_db |
+--------------------+
| categories         |
| customers          |
| departments        |
| order_items        |
| orders             |
| products           |
+--------------------+
6 rows in set (0.00 sec)
```

c) Open a new terminal and run the following Sqoop command line by line. **It would take a while to be completed, because Sqoop launches MapReduce job.**

```
[cloudera@quickstart ~]$ sqoop import-all-tables \
-m 2 \
--connect jdbc:mysql://localhost:3306/retail_db \
--username=root \
--password=cloudera \
--compression-codec=snappy \
--as-parquetfile \
--warehouse-dir=/user/hive/warehouse \
--hive-overwrite \
--hive-import
```

After about 10 minutes, you have to see the following screenshot as the confirmation:



d) **Verification**: When this command is complete, confirm that your data files exist in HDFS. These commands will show the directories and the files inside them that make up your tables:

- `hdfs dfs -ls /user/hive/warehouse/`
- `hdfs dfs -ls /user/hive/warehouse/categories/`

```
[cloudera@quickstart ~]$ hdfs dfs -ls /user/hive/warehouse/
Found 6 items
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 14:57 /user/hive/warehouse/categories
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 14:59 /user/hive/warehouse/customers
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 15:02 /user/hive/warehouse/departments
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 15:04 /user/hive/warehouse/order_items
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 15:06 /user/hive/warehouse/orders
drwxrwxrwx   - cloudera supergroup          0 2019-10-20 15:08 /user/hive/warehouse/products
[cloudera@quickstart ~]$ hdfs dfs -ls /user/hive/warehouse/categories
Found 4 items
drwxr-xr-x   - cloudera supergroup          0 2019-10-20 14:56 /user/hive/warehouse/categories/.metadata
drwxr-xr-x   - cloudera supergroup          0 2019-10-20 14:57 /user/hive/warehouse/categories/.signals
-rw-r--r--   1 cloudera supergroup       1491 2019-10-20 14:57 /user/hive/warehouse/categories/0e01f6e7-4f7a-4b66-9225-34c2bbe9bfd6.parquet
-rw-r--r--   1 cloudera supergroup       1520 2019-10-20 14:57 /user/hive/warehouse/categories/c141345a-2a78-48cc-8801-058e7e53d102.parquet
```

Note: The number of `.parquet` files shown will be equal to what was passed to Sqoop with the `-m` parameter. This is the number of 'mappers' that Sqoop will use in its MapReduce jobs. It could also be thought of as the number of simultaneous connections to your database, or the number of disks/Data Nodes you want to spread the data across. So on a single-node you will just see one, but larger clusters will have a greater number of files.

OR, complete the verification step visually (localhost:50070):

/user/hive/warehouse Go!

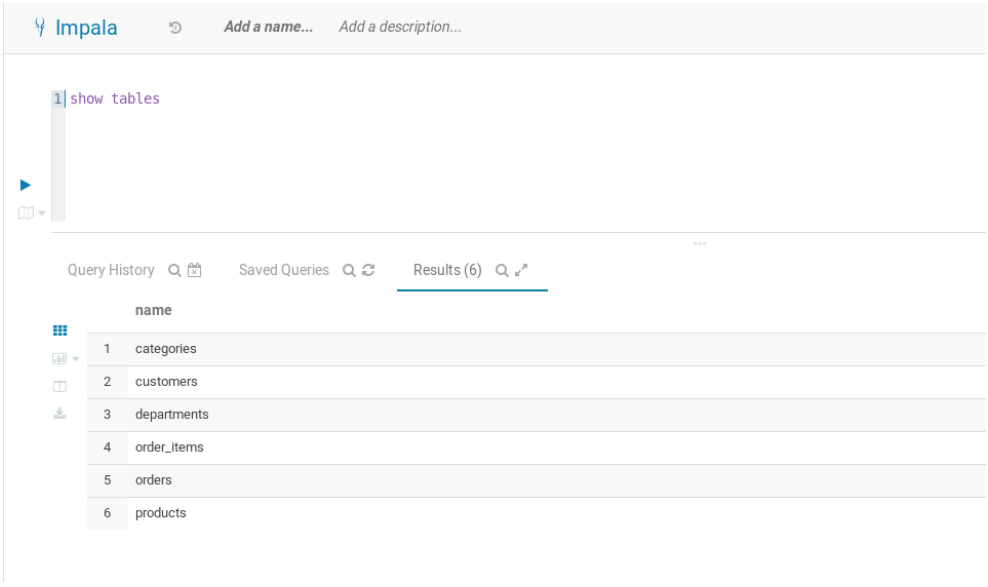
Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 14:57:39 -0700 2019	0	0 B	categories
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 14:59:25 -0700 2019	0	0 B	customers
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 15:02:03 -0700 2019	0	0 B	departments
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 15:04:20 -0700 2019	0	0 B	order_items
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 15:06:39 -0700 2019	0	0 B	orders
drwxrwxrwx	cloudera	supergroup	0 B	Sun Oct 20 15:08:46 -0700 2019	0	0 B	products

Phase 3: Working with Impala

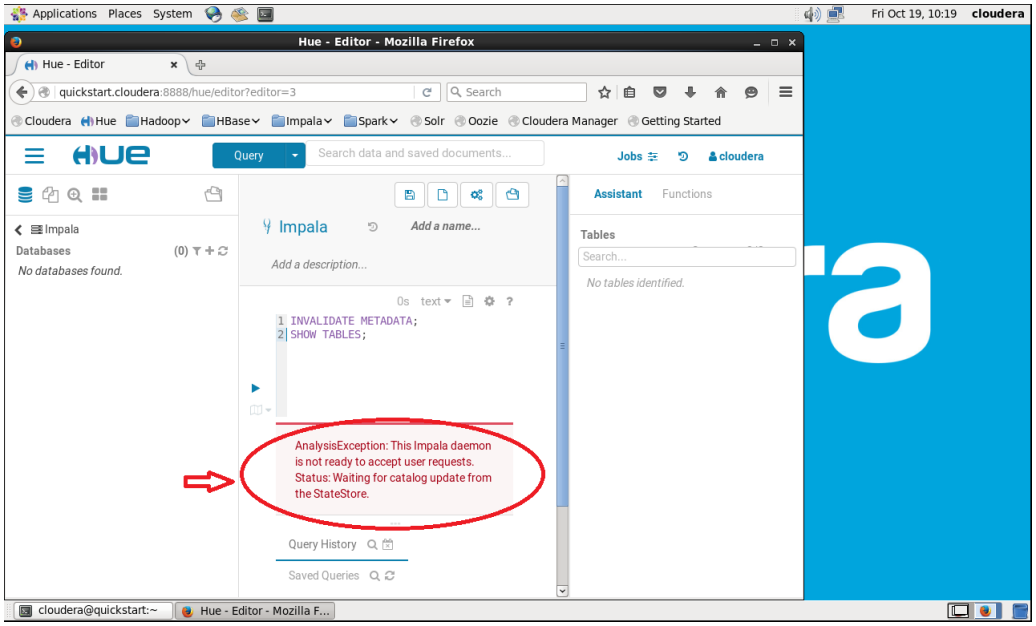
We are going to use Hue's Impala to query our tables. *Impala is a query engine that runs on Hadoop*. Once you are inside the Hue editor, click on **Query → Editor → Impala** to launch the Impala Editor.

To save time during queries, Impala does not poll constantly for metadata changes. So the first thing we must do is tell Impala that its metadata is out of date. Then we should see our tables show up, ready to be queried:

```
invalidate metadata;
show tables;
```



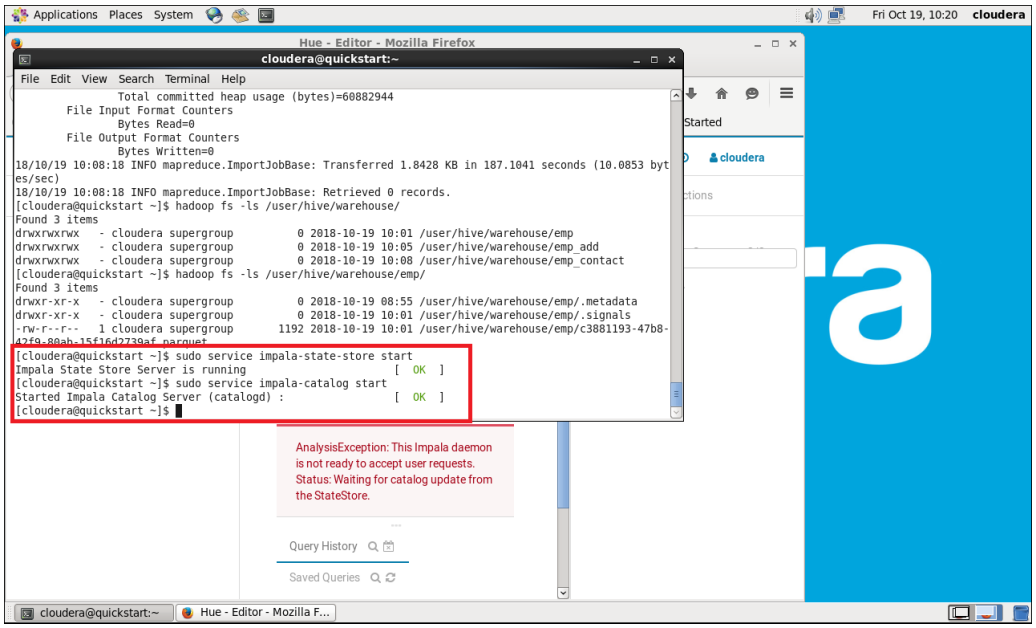
[Exceptional Error:] You may get the following error when you run the commands. This is because both services `impala-state-store` and `impala-catalog` are down and need to be restarted.



To fix the error**, back to the terminal and type the following commands:**

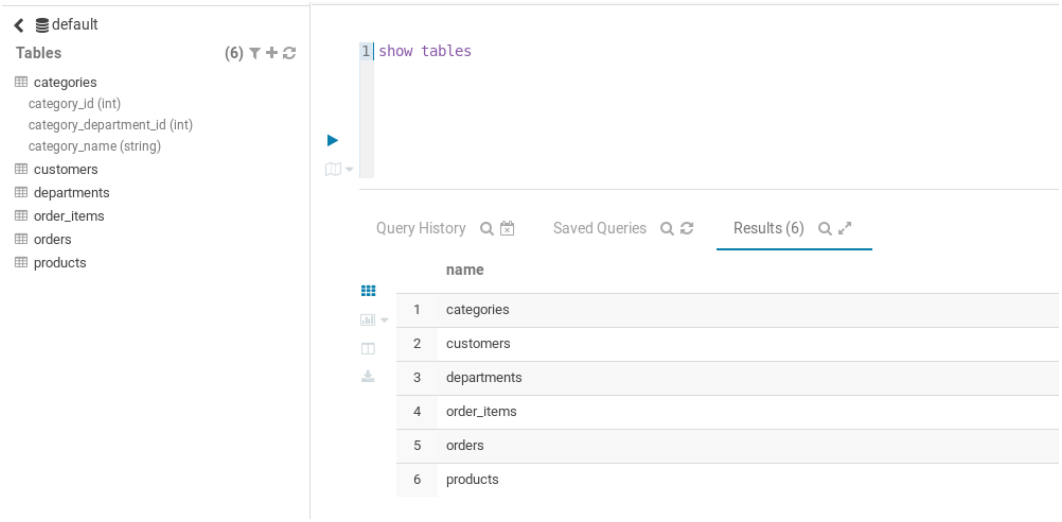
```
sudo service impala-state-store start

sudo service impala-catalog start
```



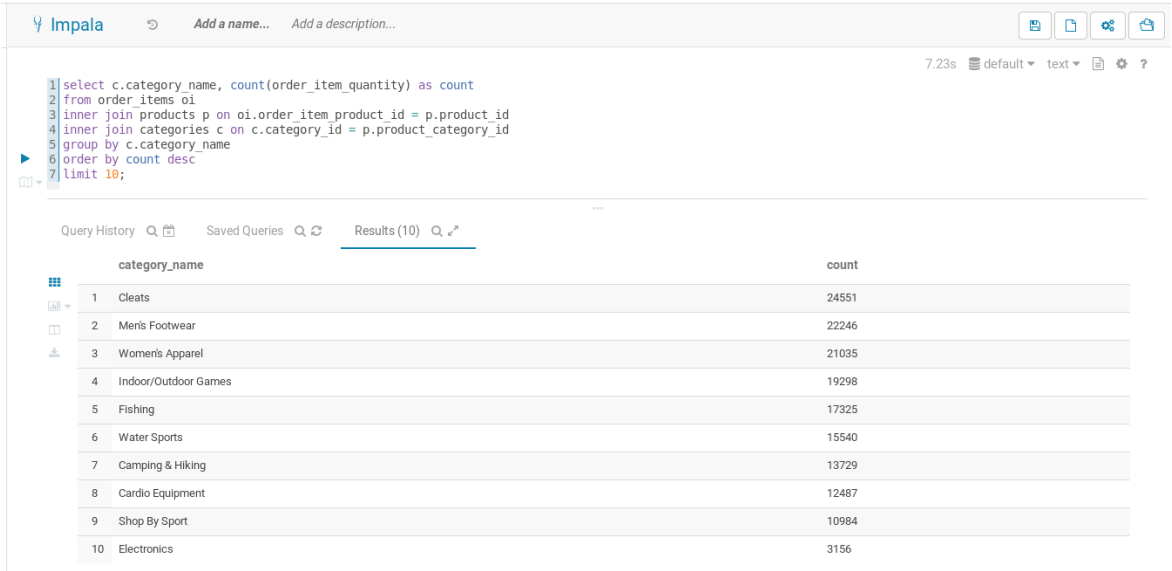
Now click on Hue icon again to refresh the page and then run the query. You can now see the new tables in the result menu.

e) Now, your transaction data is readily available for structured queries in CDH, *it's time to address a few questions. You can make a few basic/advanced SQL queries to get familiar with the retail_db database*. Also, you can see the tables' properties visually in the Hue platform, such as:



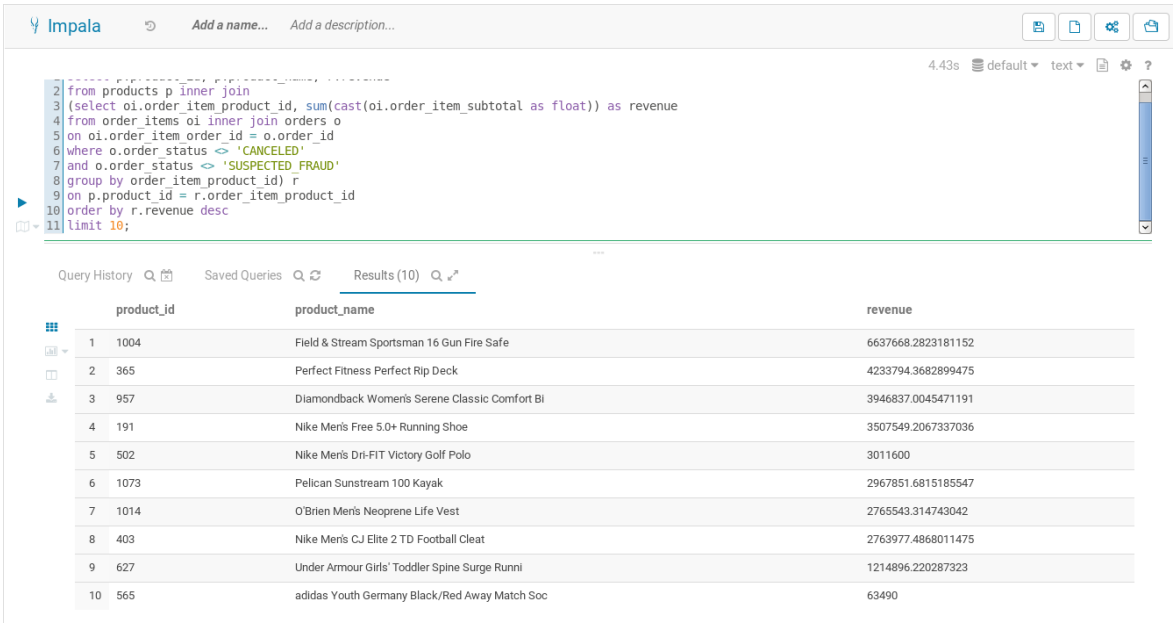
f) SQL query 1: calculate the total revenue per product and showing the top 10 revenue generating products:

```
select c.category_name, count(order_item_quantity) as count
from order_items oi
inner join products p on oi.order_item_product_id = p.product_id
inner join categories c on c.category_id = p.product_category_id
group by c.category_name
order by count desc
limit 10;
```



g) SQL query 2: get the top 10 revenue generating products:

```
select p.product_id, p.product_name, r.revenue
from products p inner join
(select oi.order_item_product_id,
sum(cast(oi.order_item_subtotal as float)) as revenue
from order_items oi inner join orders o
on oi.order_item_order_id = o.order_id
where o.order_status <> 'CANCELED'
and o.order_status <> 'SUSPECTED_FRAUD'
group by order_item_product_id) r
on p.product_id = r.order_item_product_id
order by r.revenue desc
limit 10;
```



	product_id	product_name	revenue
1	1004	Field & Stream Sportsman 16 Gun Fire Safe	6637668.2823181152
2	365	Perfect Fitness Perfect Rip Deck	4233794.3682899475
3	957	Diamondback Women's Serene Classic Comfort Bi	3946837.0045471191
4	191	Nike Men's Free 5.0+ Running Shoe	3507549.2067337036
5	502	Nike Men's Dri-FIT Victory Golf Polo	3011600
6	1073	Pelican Sunstream 100 Kayak	2967851.6815185547
7	1014	O'Brien Men's Neoprene Life Vest	2765543.314743042
8	403	Nike Men's CJ Elite 2 TD Football Cleat	2763977.4868011475
9	627	Under Armour Girls' Toddler Spine Surge Runni	1214896.220287323
10	565	adidas Youth Germany Black/Red Away Match Soc	63490