1. Busca pelos dados

Plataforma para a coleta do dataset:

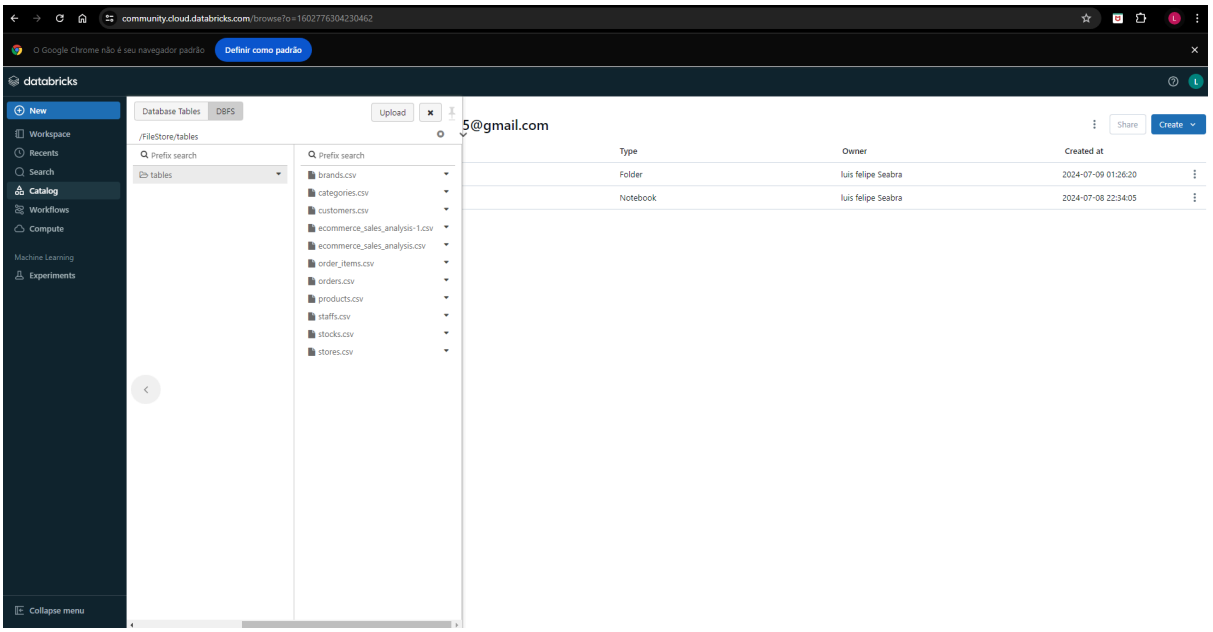
- Kaggle (https://www.kaggle.com/datasets)

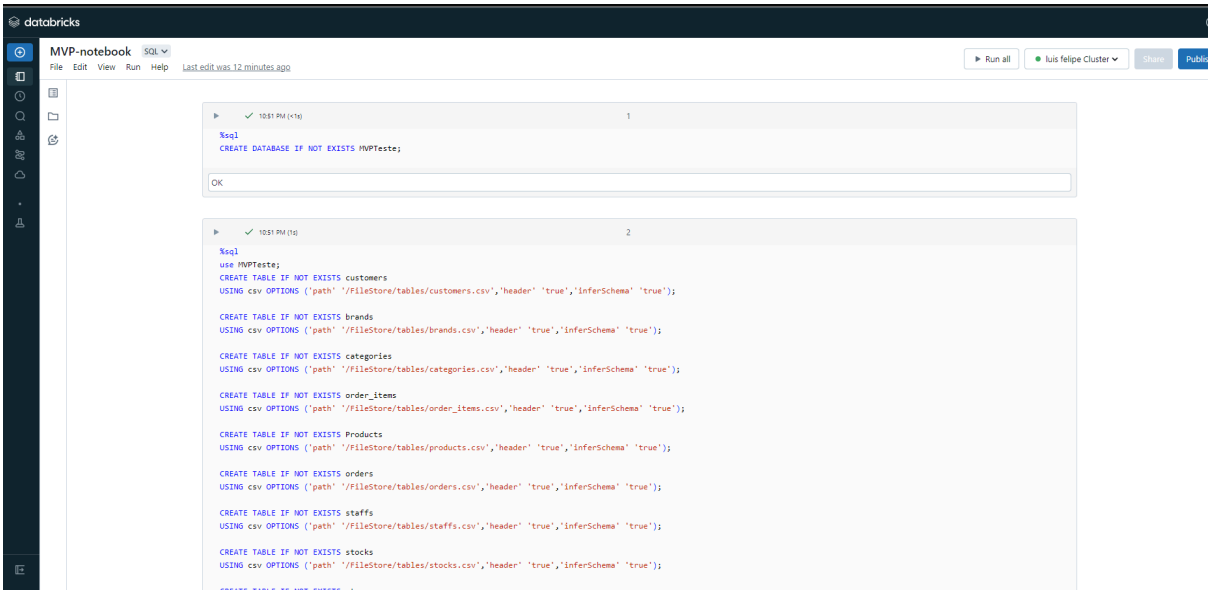
2. Coleta

Base de dados para utilizar no MVP:

Uma vez definido o conjunto de dados, devemos coletar e armazená-los na nuvem.

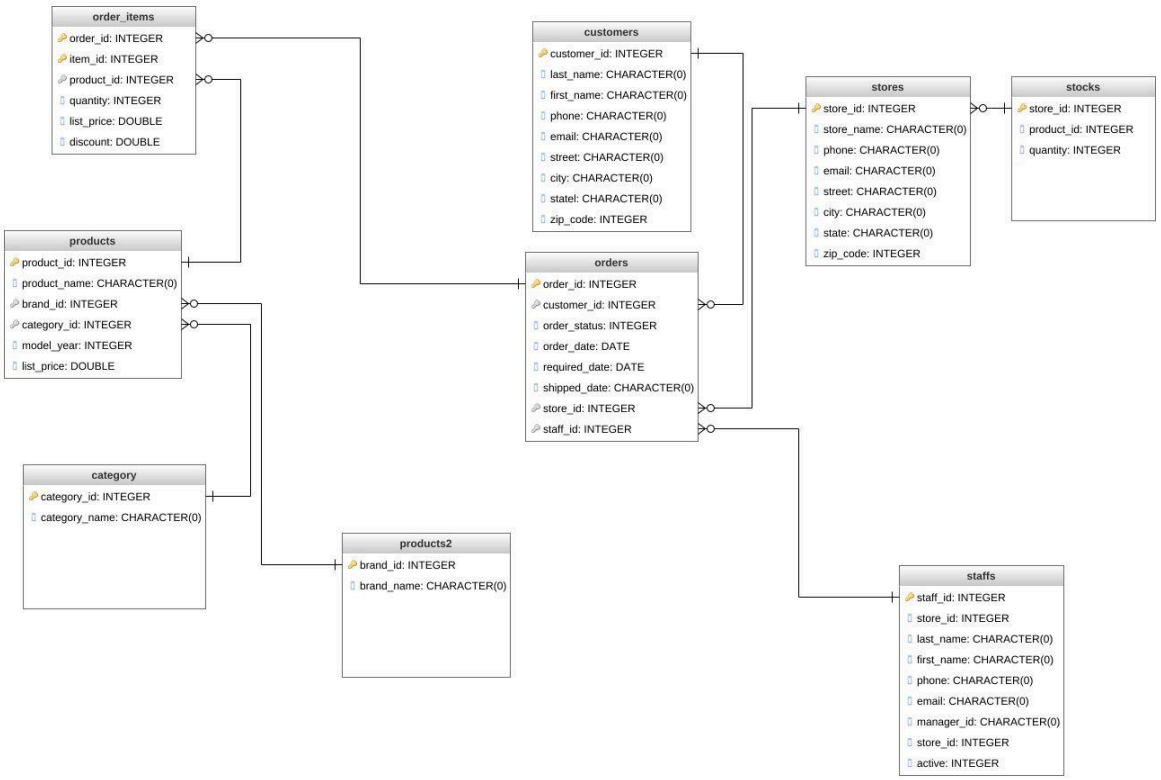
https://www.kaggle.com/datasets/dillonmyrick/bike-store-sample-database





3. Modelagem

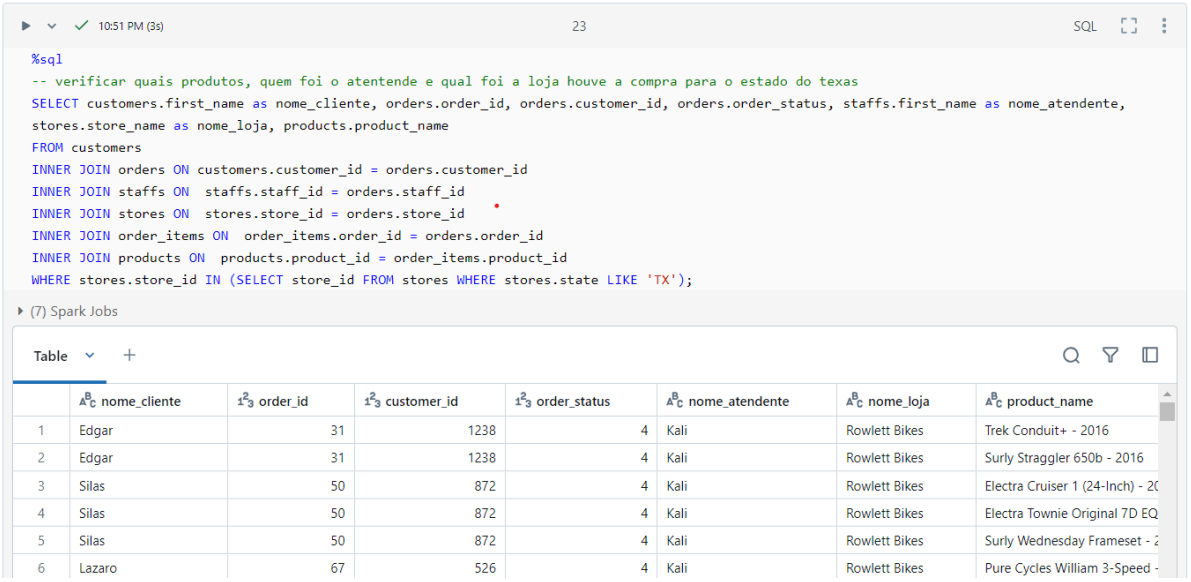
A modelagem Snowflake é uma técnica de design de banco de dados que normaliza dados de dimensão em várias tabelas menores, criando uma estrutura mais organizada e sem redundâncias. Embora essa abordagem possa trazer benefícios em termos de integridade e economia de espaço, ela também pode introduzir complexidade e impactar o desempenho das consultas. Portanto, a escolha entre um esquema Snowflake e um esquema estrela deve ser baseada nos requisitos específicos do projeto e nas prioridades de desempenho e manutenção.



4. Carga

10:51 PM (1s) 2 SQL

```
%sql
use MVPTeste;
CREATE TABLE IF NOT EXISTS customers
USING csv OPTIONS ('path' '/FileStore/tables/customers.csv','header' 'true','inferSchema' 'true');

CREATE TABLE IF NOT EXISTS brands
USING csv OPTIONS ('path' '/FileStore/tables/brands.csv','header' 'true','inferSchema' 'true');

CREATE TABLE IF NOT EXISTS categories
USING csv OPTIONS ('path' '/FileStore/tables/categories.csv','header' 'true','inferSchema' 'true');

CREATE TABLE IF NOT EXISTS order_items
USING csv OPTIONS ('path' '/FileStore/tables/order_items.csv','header' 'true','inferSchema' 'true');

CREATE TABLE IF NOT EXISTS Products
USING csv OPTIONS ('path' '/FileStore/tables/products.csv','header' 'true','inferSchema' 'true');

CREATE TABLE IF NOT EXISTS orders
USING csv OPTIONS ('path' '/FileStore/tables/orders.csv','header' 'true','inferSchema' 'true');
```



```
%sql
-- verificar quais produtos, quem foi o atentende e qual foi a loja houve a compra para o estado do texas
SELECT customers.first_name as nome_cliente, orders.order_id, orders.customer_id, orders.order_status, staffs.first_name as nome_atendente,
stores.store_name as nome_loja, products.product_name
FROM customers
INNER JOIN orders ON customers.customer_id = orders.customer_id
INNER JOIN staffs ON  staffs.staff_id = orders.staff_id
INNER JOIN stores ON  stores.store_id = orders.store_id
INNER JOIN order_items ON  order_items.order_id = orders.order_id
INNER JOIN products ON  products.product_id = order_items.product_id
WHERE stores.store_id IN (SELECT store_id FROM stores WHERE stores.state LIKE 'TX');
```

10:51 PM (3s) 23 SQL

(7) Spark Jobs

Table

	nome_cliente	order_id	customer_id	order_status	nome_atendente	nome_loja	product_name
1	Edgar	31	1238	4	Kali	Rowlett Bikes	Trek Conduit+ - 2016
2	Edgar	31	1238	4	Kali	Rowlett Bikes	Surly Straggler 650b - 2016
3	Silas	50	872	4	Kali	Rowlett Bikes	Electra Cruiser 1 (24-Inch) - 20
4	Silas	50	872	4	Kali	Rowlett Bikes	Electra Townie Original 7D EQ
5	Silas	50	872	4	Kali	Rowlett Bikes	Surly Wednesday Frameset - 2
6	Lazaro	67	526	4	Kali	Rowlett Bikes	Pure Cycles William 3-Speed -

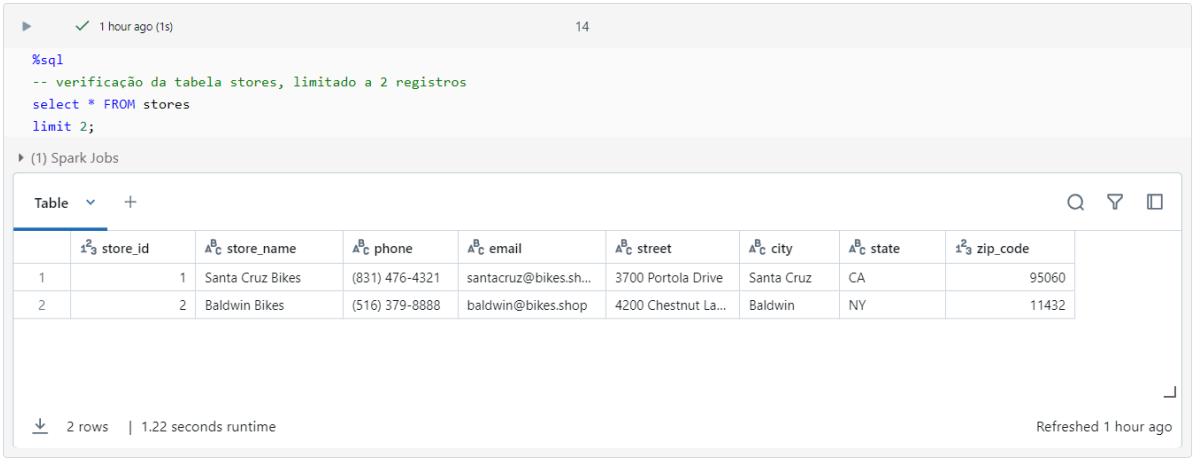
5. Análise

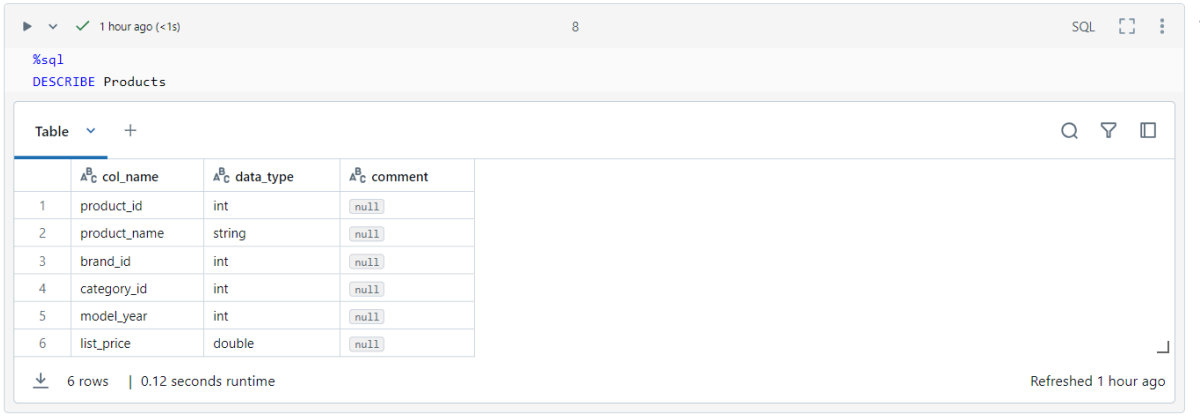
A análise da qualidade dos dados é um componente crítico de qualquer estratégia de gestão de dados. Ela ajuda a assegurar que as decisões sejam baseadas em informações precisas e confiáveis, melhorando a eficiência operacional, a satisfação do cliente, e a conformidade regulatória, além de reduzir custos e promover a segurança e a privacidade dos dados.
No item a. verificaremos todas as análises que foram efetuadas com o intuito de entender melhor o conjunto de dados.

Passos seguidos:
Entendimento do Conjunto de Dados; Análise de Valores Nulos; Verificação de Valores Duplicados; Consistência de Dados; Análise de Outliers; Verificação de Formatos de Dados

a. Qualidade de dados

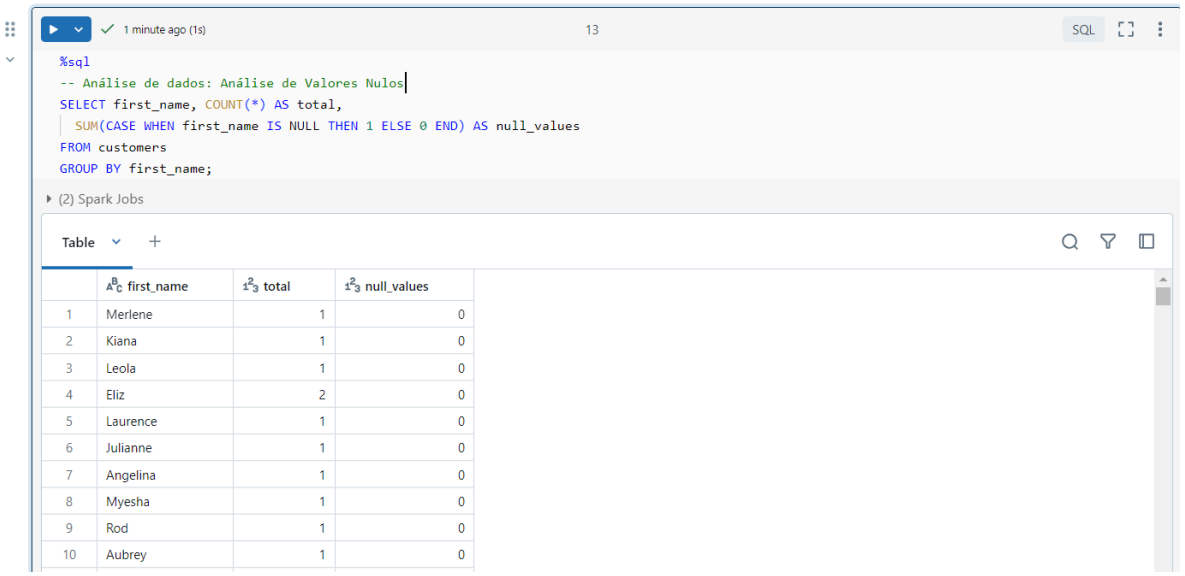
Entendimento do Conjunto de Dados





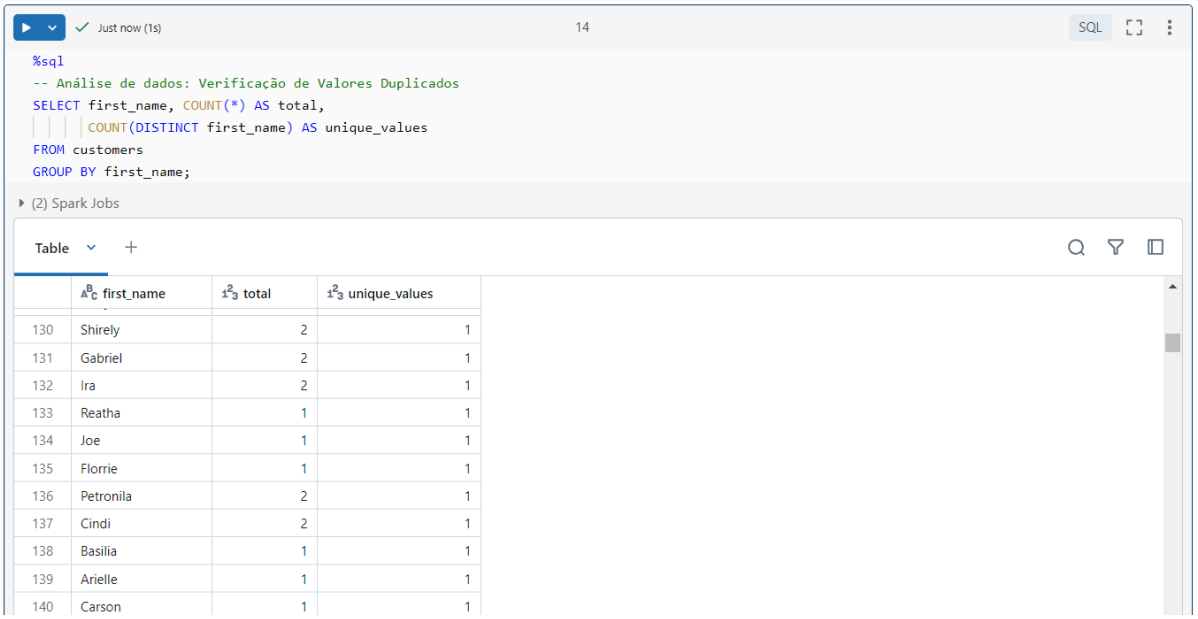
Análise de Valores Nulos

Verifique a presença de valores nulos em cada coluna. Valores nulos podem indicar dados ausentes ou incompletos.



Verificação de Valores Duplicados

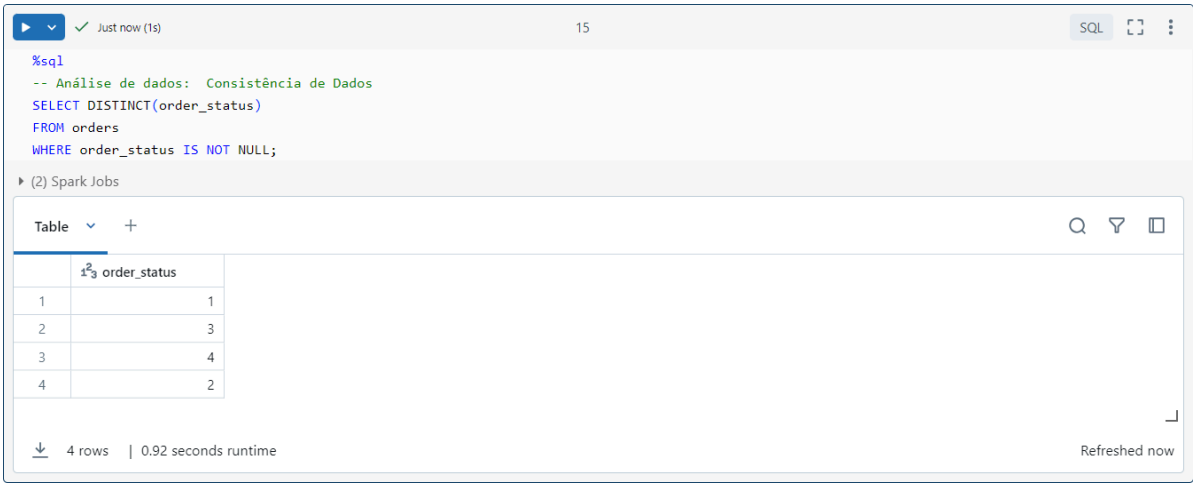
Valores duplicados podem ser problemáticos em certos contextos. Verifique se há registros duplicados que não deveriam existir.



	first_name	total	unique_values
130	Shirely	2	1
131	Gabriel	2	1
132	Ira	2	1
133	Reatha	1	1
134	Joe	1	1
135	Florrie	1	1
136	Petronila	2	1
137	Cindi	2	1
138	Basilia	1	1
139	Arielle	1	1
140	Carson	1	1

Consistência de Dados

Verifique a consistência dos dados, especialmente em colunas que deveriam ter valores consistentes.

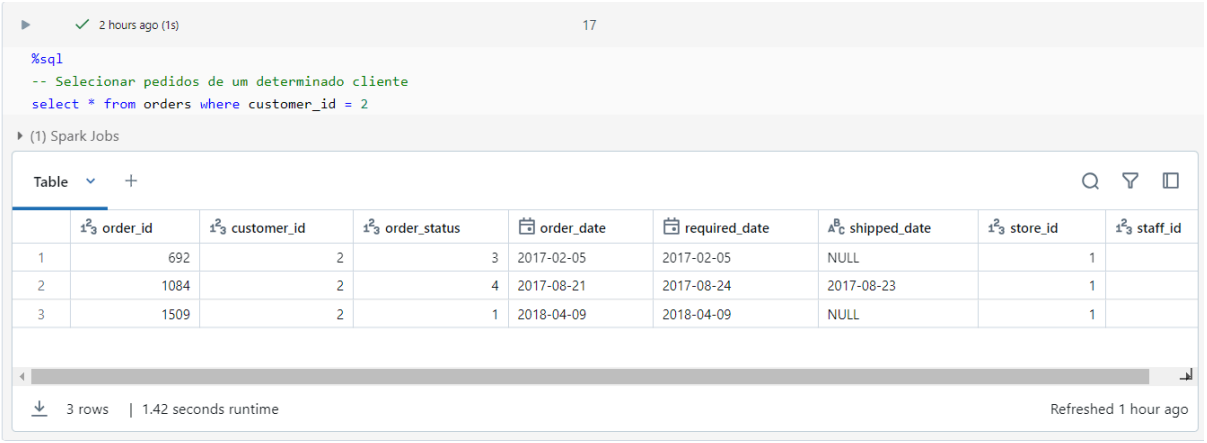


	order_status
1	1
2	3
3	4
4	2

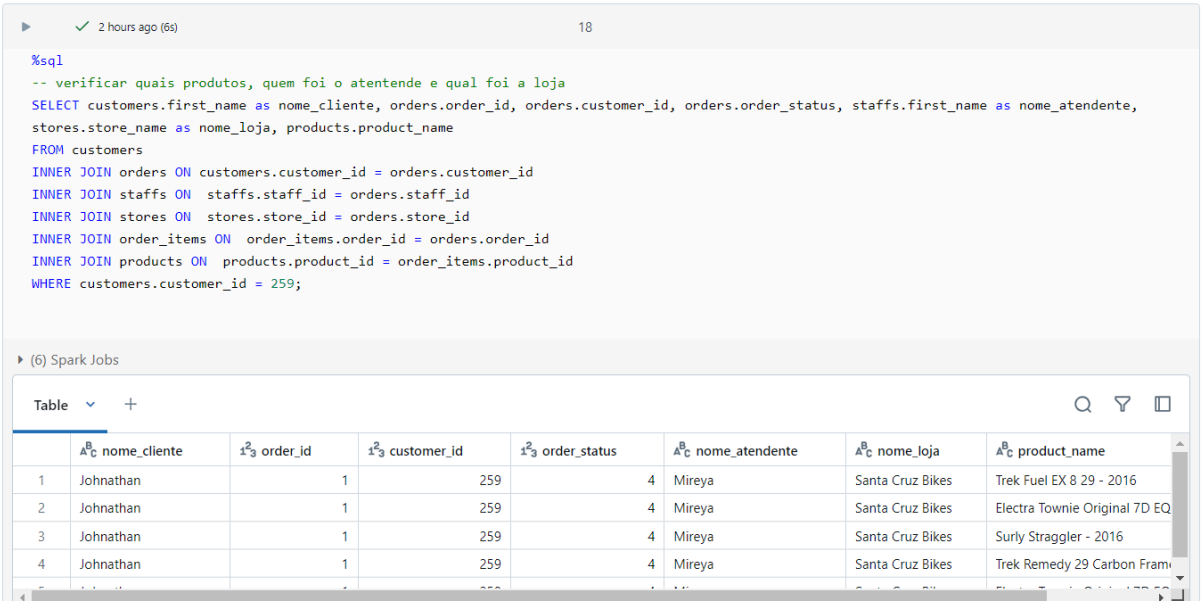
b. Solução do problema

Após análise dos dados, foram respondidas várias perguntas, como:

1.Selecionar produtos comprados por determinado cliente:



2. Verificar quais produtos foram vendidos, quem foi o atendente, qual foi a loja onde o produto foi vendido para determinado cliente.



3. Verificar qual foi o top e clientes que mais fizeram pedidos

2 hours ago (4s) 19 SQL

```
%sql
-- verificar top 3 clientes que mais fizeram pedidos
SELECT count(orders.order_id) as quantidade_pedidos, customers.first_name, customers.customer_id
FROM customers
INNER JOIN orders ON customers.customer_id = orders.customer_id
GROUP BY customers.customer_id, customers.first_name
ORDER BY COUNT(orders.order_id) DESC
LIMIT 3;
```

(3) Spark Jobs

Table

	quantidade_pedidos	first_name	customer_id
1	3	Aleta	20
2	3	Charolette	5

4. Verificar quais funcionários efetuaram mais vendas

2 hours ago (2s) 20

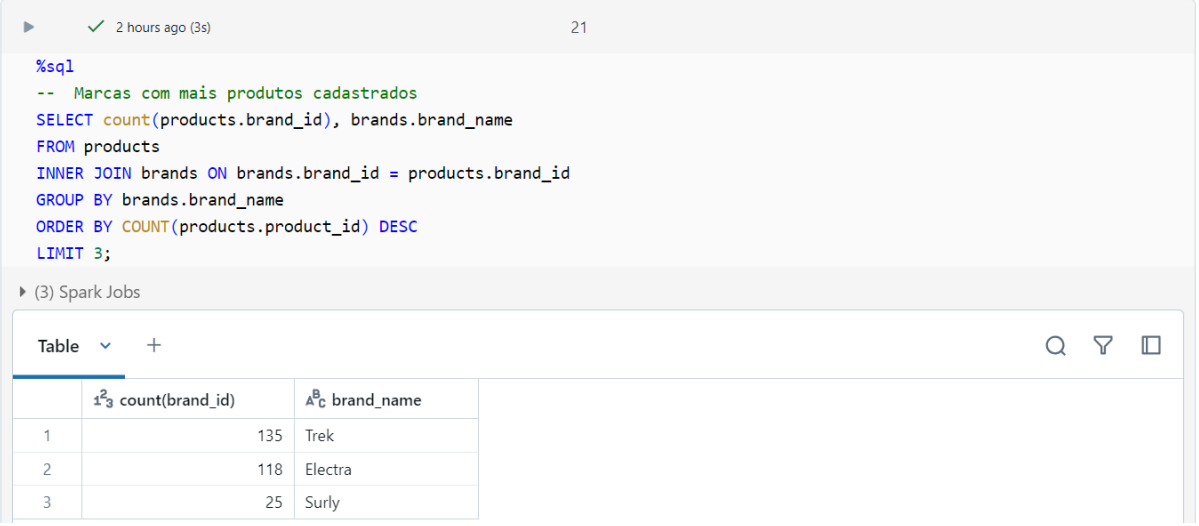
```
%sql
-- verificar quais atendentes mais efetuaram pedidos de vendas
SELECT count(orders.order_id) as quantidade_pedidos, staffs.first_name, staffs.staff_id
FROM staffs
INNER JOIN orders ON staffs.staff_id = orders.staff_id
GROUP BY staffs.staff_id, staffs.first_name
ORDER BY COUNT(orders.order_id) DESC
LIMIT 3;
```

(3) Spark Jobs

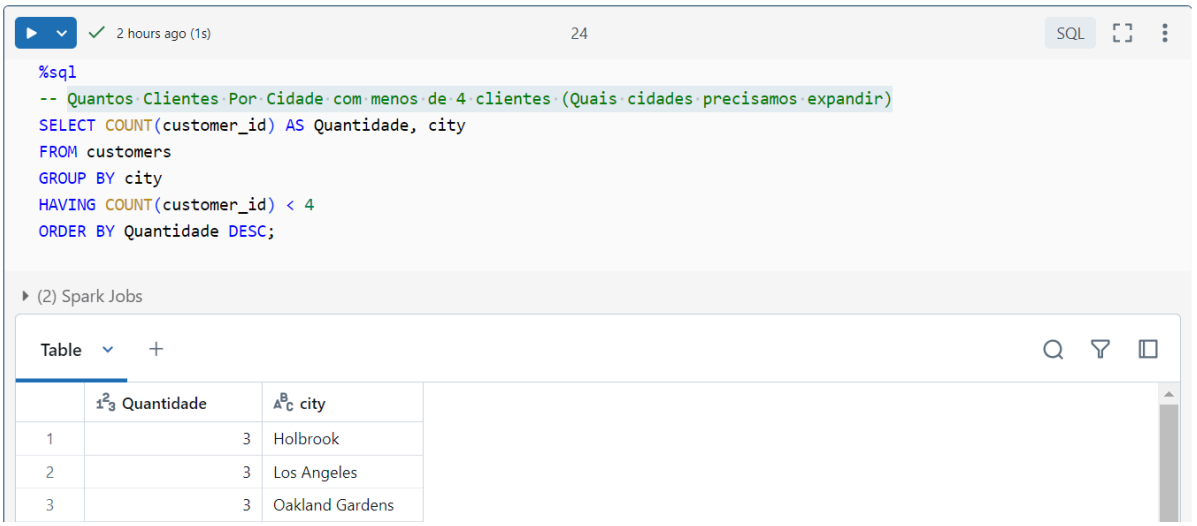
Table

	quantidade_pedidos	first_name	staff_id
1	553	Marcelene	6
2	540	Venita	7
3	184	Genna	3

5. Verificar quais marcas possuem mais produtos cadastrados:



6. Verificar quais cidades precisamos expandir(loja). Retorno de cidades onde foram feitos menos de 4 pedidos.



7. verificar quais produtos, quem foi o atentende e qual foi a loja que houve a compra no estado do texas

2 hours ago (3s) 26

```
-- verificar quais produtos, quem foi o atentende e qual foi a loja houve a compra para o estado do texas
SELECT customers.first_name as nome_cliente, orders.order_id, orders.customer_id, orders.order_status, staffs.
first_name as nome_atendente, stores.store_name as nome_loja, products.product_name
FROM customers
INNER JOIN orders ON customers.customer_id = orders.customer_id
INNER JOIN staffs ON  staffs.staff_id = orders.staff_id
INNER JOIN stores ON  stores.store_id = orders.store_id
INNER JOIN order_items ON  order_items.order_id = orders.order_id
INNER JOIN products ON  products.product_id = order_items.product_id
WHERE stores.store_id IN (SELECT store_id FROM stores WHERE stores.state LIKE 'TX');
```

(7) Spark Jobs

Table

	nome_cliente	order_id	customer_id	order_status	nome_atendente	nome_loja
1	Edgar	31	1238	4	Kali	Rowlett Bikes
2	Edgar	31	1238	4	Kali	Rowlett Bikes