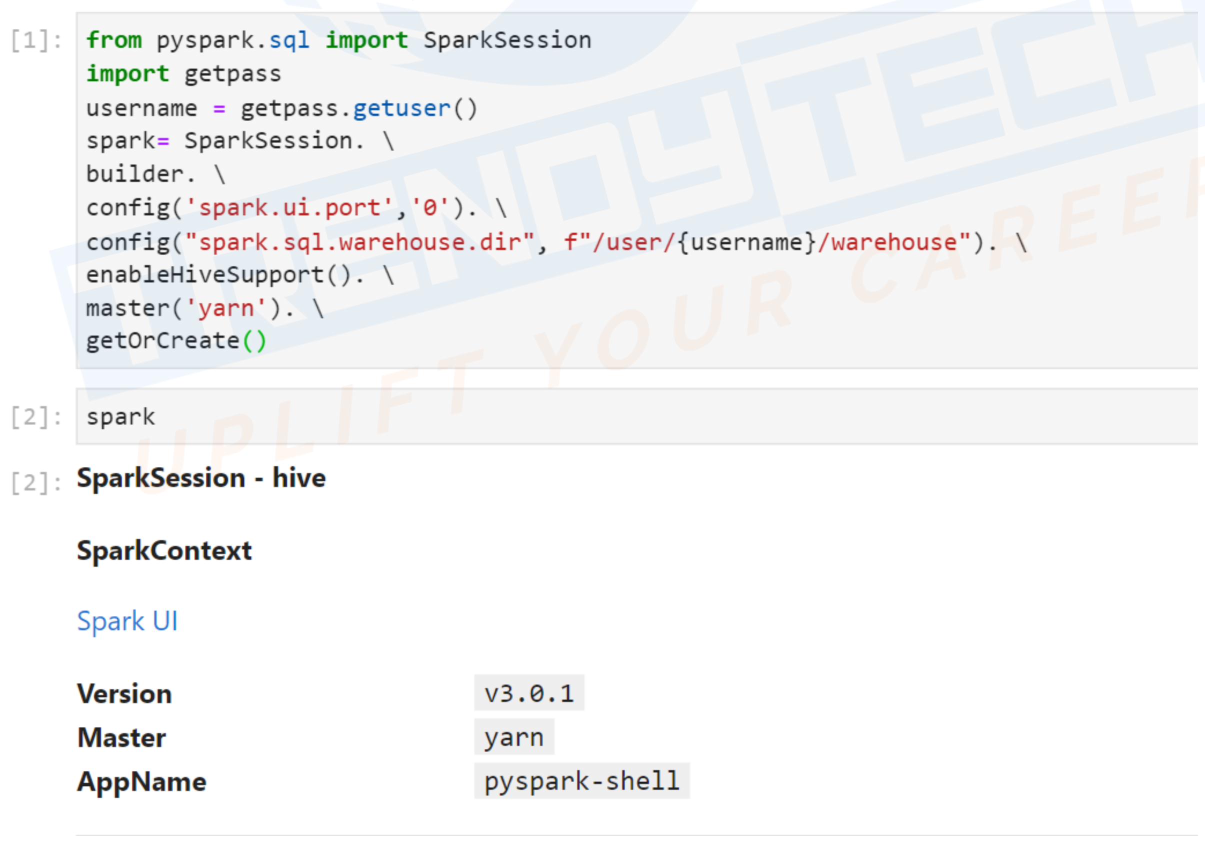
ASSIGNMENT SOLUTION

The following Common Boilerplate code to create a Spark Session has to be executed before running the queries.

```
from pyspark.sql import SparkSession

from pyspark.sql import SparkSession

import getpass

username = getpass.getuser()

spark= SparkSession. \

builder. \

config('spark.ui.port','0'). \

config("spark.sql.warehouse.dir", f"/user/{username}/warehouse"). \

enableHiveSupport(). \

master('yarn'). \

getOrCreate()
```



```
[1]: from pyspark.sql import SparkSession
     import getpass
     username = getpass.getuser()
     spark= SparkSession. \
     builder. \
     config('spark.ui.port','0'). \
     config("spark.sql.warehouse.dir", f"/user/{username}/warehouse"). \
     enableHiveSupport(). \
     master('yarn'). \
     getOrCreate()

[2]: spark

[2]: SparkSession - hive

     SparkContext

     Spark UI

     Version    v3.0.1
     Master     yarn
     AppName    pyspark-shell
```

Question 1

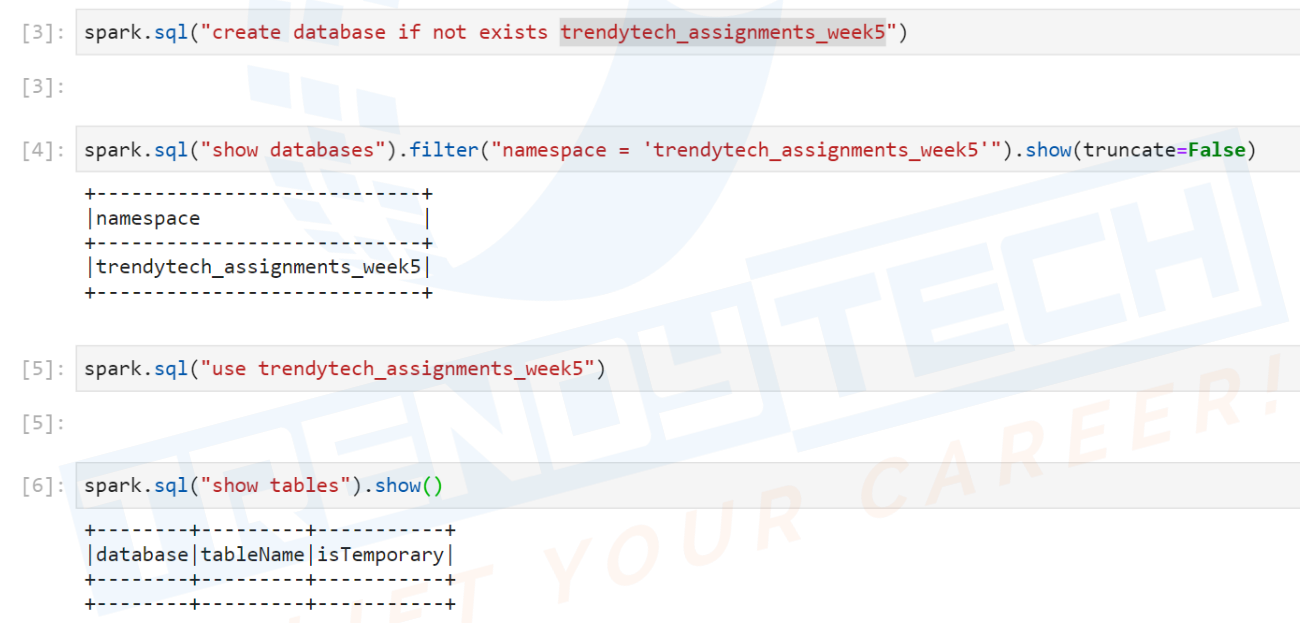
1.1

spark.sql("create database if not exists trendytech_assignments_week5")

spark.sql("show databases").filter("namespace = 'trendytech_assignments_week5'").show(truncate=False)

spark.sql("use trendytech_assignments_week5")

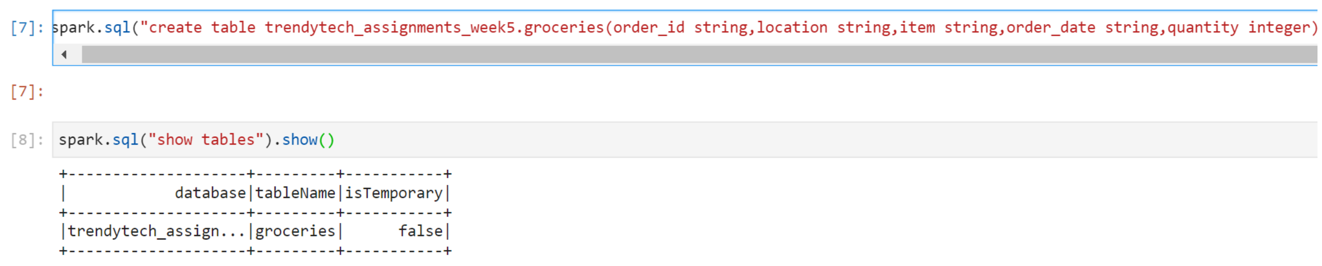
spark.sql("show tables").show()



```
[3]: spark.sql("create database if not exists trendytech_assignments_week5")
[3]:

[4]: spark.sql("show databases").filter("namespace = 'trendytech_assignments_week5'").show(truncate=False)
+----------------------------+
|namespace                   |
+----------------------------+
|trendytech_assignments_week5|
+----------------------------+

[5]: spark.sql("use trendytech_assignments_week5")
[5]:

[6]: spark.sql("show tables").show()
+--------+---------+-----------+
|database|tableName|isTemporary|
+--------+---------+-----------+
+--------+---------+-----------+
```

spark.sql("create table trendytech_assignments_week5.groceries(order_id string,location string,item string,order_date string,quantity integer)")

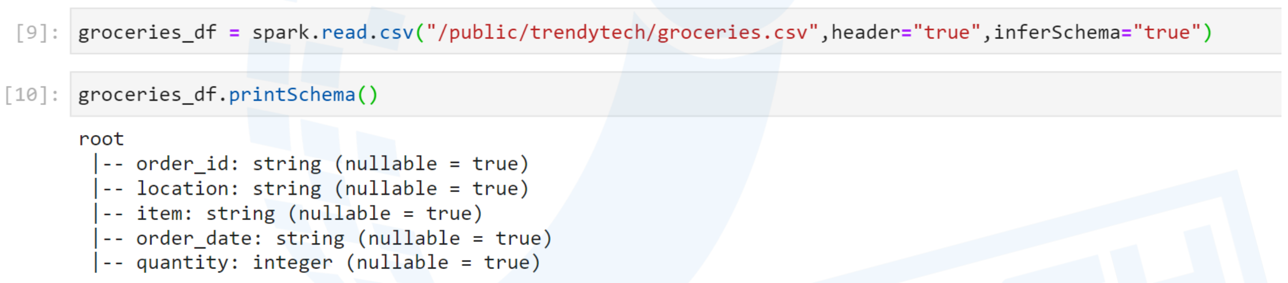
spark.sql("show tables").show()



```
[7]: spark.sql("create table trendytech_assignments_week5.groceries(order_id string,location string,item string,order_date string,quantity integer)

[7]:

[8]: spark.sql("show tables").show()

+--------------------+---------+-----------+
|            database|tableName|isTemporary|
+--------------------+---------+-----------+
|trendytech_assign...|groceries|      false|
+--------------------+---------+-----------+
```

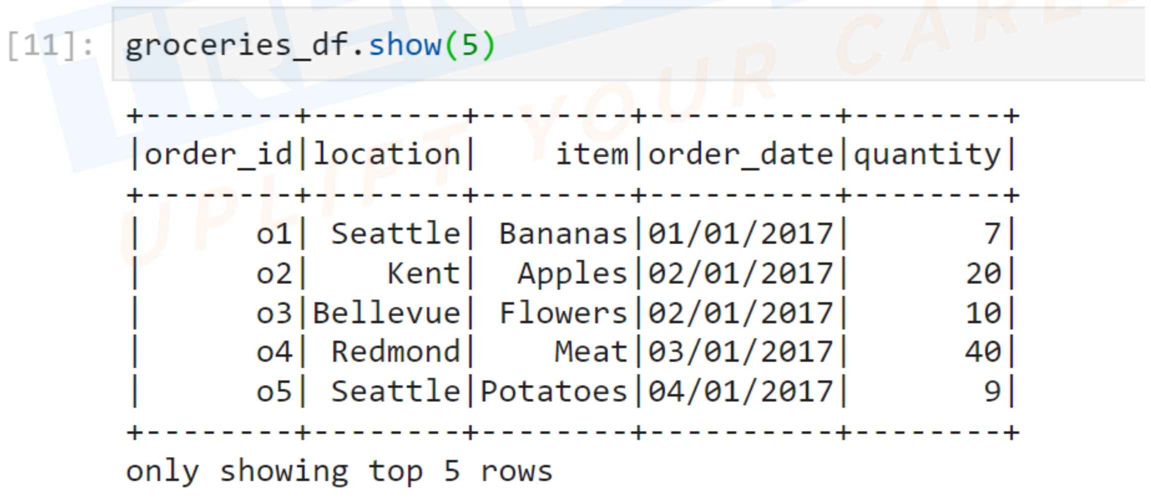
groceries_df = spark.read.csv("/public/trendytech/groceries.csv",header="true",inferSchema="true")

groceries_df.printSchema()



```
[9]: groceries_df = spark.read.csv("/public/trendytech/groceries.csv",header="true",inferSchema="true")

[10]: groceries_df.printSchema()

root
 |-- order_id: string (nullable = true)
 |-- location: string (nullable = true)
 |-- item: string (nullable = true)
 |-- order_date: string (nullable = true)
 |-- quantity: integer (nullable = true)
```

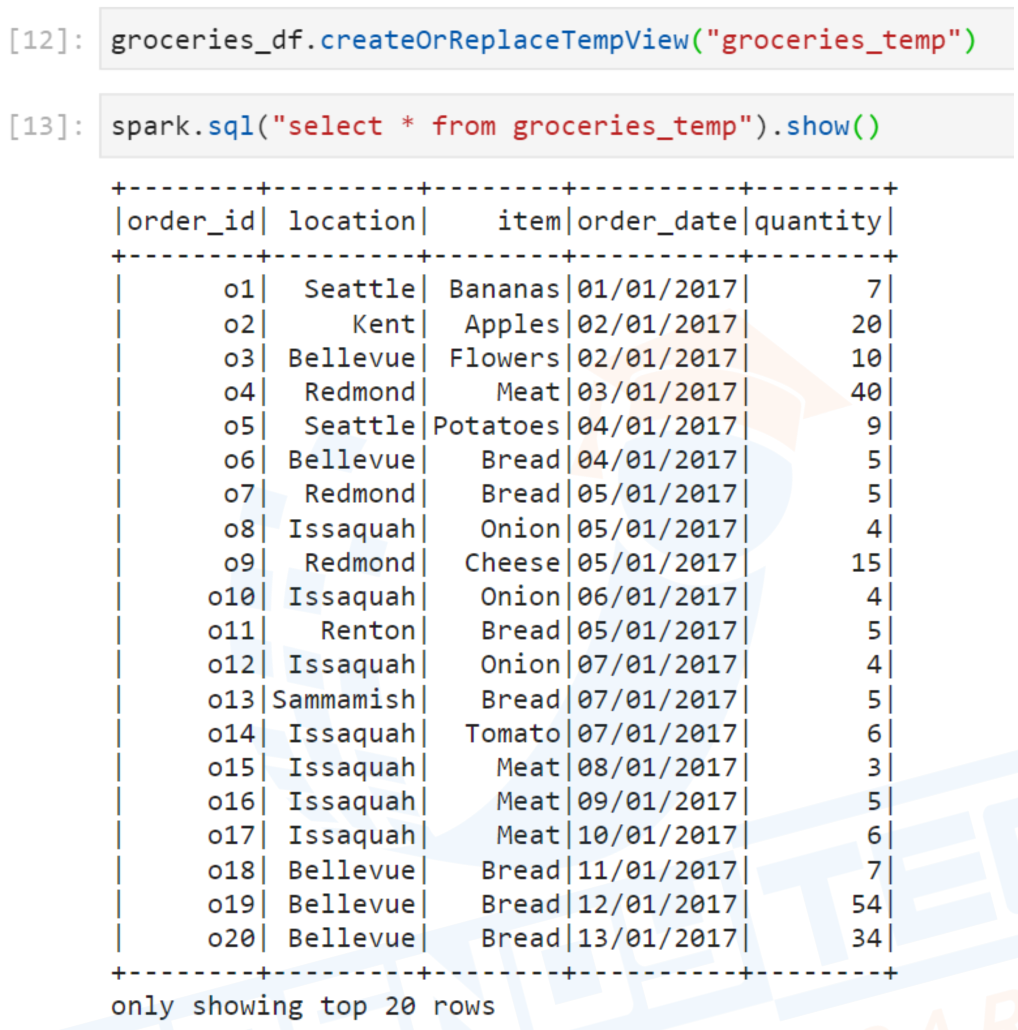
groceries_df.show(5)



```
[11]: groceries_df.show(5)

+--------+--------+--------+----------+--------+
|order_id|location|    item|order_date|quantity|
+--------+--------+--------+----------+--------+
|      o1| Seattle| Bananas|01/01/2017|       7|
|      o2|    Kent|  Apples|02/01/2017|      20|
|      o3|Bellevue| Flowers|02/01/2017|      10|
|      o4| Redmond|    Meat|03/01/2017|      40|
|      o5| Seattle|Potatoes|04/01/2017|       9|
+--------+--------+--------+----------+--------+
only showing top 5 rows
```

groceries_df.createOrReplaceTempView("groceries_temp")

spark.sql("select * from groceries_temp").show()



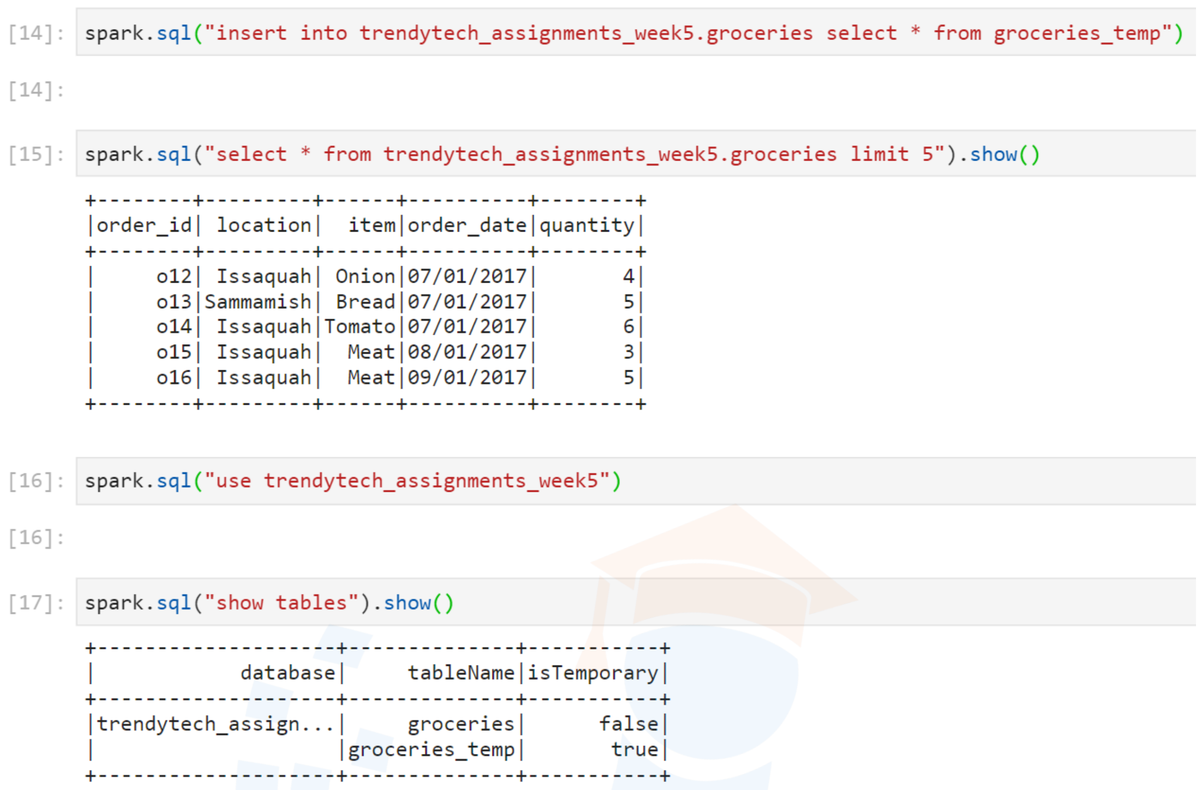
```
[12]: groceries_df.createOrReplaceTempView("groceries_temp")

[13]: spark.sql("select * from groceries_temp").show()

+--------+---------+--------+----------+--------+
|order_id| location|    item|order_date|quantity|
+--------+---------+--------+----------+--------+
|      o1|  Seattle| Bananas|01/01/2017|       7|
|      o2|     Kent|  Apples|02/01/2017|      20|
|      o3| Bellevue| Flowers|02/01/2017|      10|
|      o4|  Redmond|    Meat|03/01/2017|      40|
|      o5|  Seattle|Potatoes|04/01/2017|       9|
|      o6| Bellevue|   Bread|04/01/2017|       5|
|      o7|  Redmond|   Bread|05/01/2017|       5|
|      o8| Issaquah|   Onion|05/01/2017|       4|
|      o9|  Redmond|  Cheese|05/01/2017|      15|
|     o10| Issaquah|   Onion|06/01/2017|       4|
|     o11|   Renton|   Bread|05/01/2017|       5|
|     o12| Issaquah|   Onion|07/01/2017|       4|
|     o13|Sammamish|   Bread|07/01/2017|       5|
|     o14| Issaquah|  Tomato|07/01/2017|       6|
|     o15| Issaquah|    Meat|08/01/2017|       3|
|     o16| Issaquah|    Meat|09/01/2017|       5|
|     o17| Issaquah|    Meat|10/01/2017|       6|
|     o18| Bellevue|   Bread|11/01/2017|       7|
|     o19| Bellevue|   Bread|12/01/2017|      54|
|     o20| Bellevue|   Bread|13/01/2017|      34|
+--------+---------+--------+----------+--------+
only showing top 20 rows
```

spark.sql("insert into trendytech_assignments_week5.groceries select * from groceries_temp")

spark.sql("select * from trendytech_assignments_week5.groceries limit 5").show()

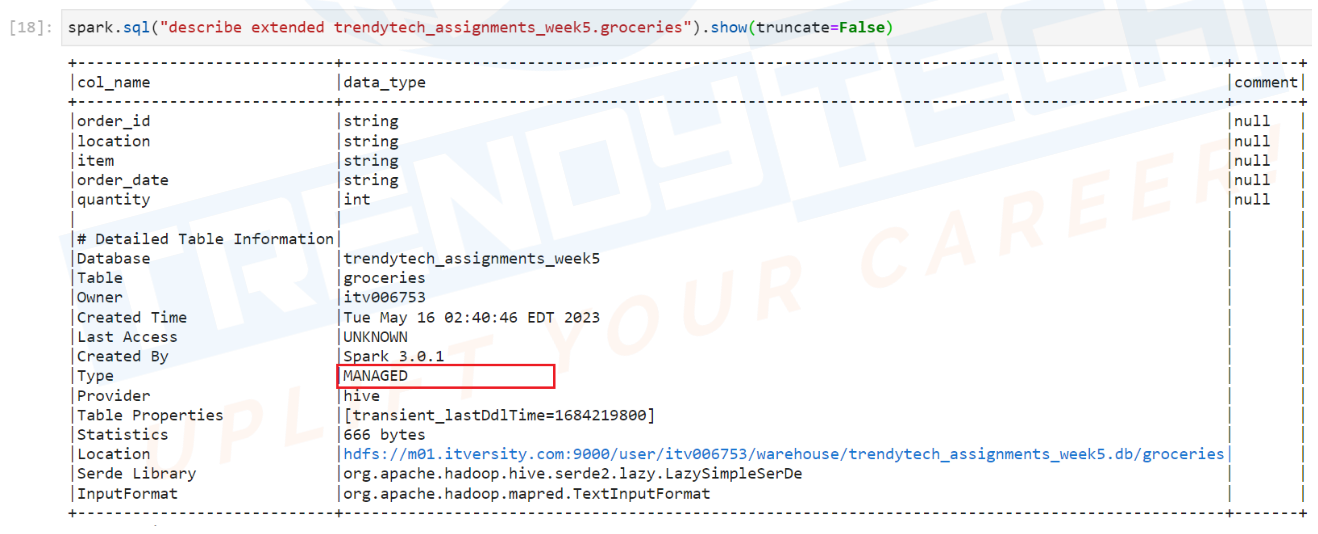
spark.sql("use trendytech_assignments_week5")

spark.sql("show tables").show()



```
[14]: spark.sql("insert into trendytech_assignments_week5.groceries select * from groceries_temp")

[14]:

[15]: spark.sql("select * from trendytech_assignments_week5.groceries limit 5").show()

+--------+---------+------+----------+--------+
|order_id| location|  item|order_date|quantity|
+--------+---------+------+----------+--------+
|     o12| Issaquah| Onion|07/01/2017|       4|
|     o13|Sammamish| Bread|07/01/2017|       5|
|     o14| Issaquah|Tomato|07/01/2017|       6|
|     o15| Issaquah|  Meat|08/01/2017|       3|
|     o16| Issaquah|  Meat|09/01/2017|       5|
+--------+---------+------+----------+--------+

[16]: spark.sql("use trendytech_assignments_week5")

[16]:

[17]: spark.sql("show tables").show()

+--------------------+--------------+-----------+
|            database|     tableName|isTemporary|
+--------------------+--------------+-----------+
|trendytech_assign...|     groceries|      false|
|                    |groceries_temp|       true|
+--------------------+--------------+-----------+
```

spark.sql("describe extended trendytech_assignments_week5.groceries").show(truncate=False)



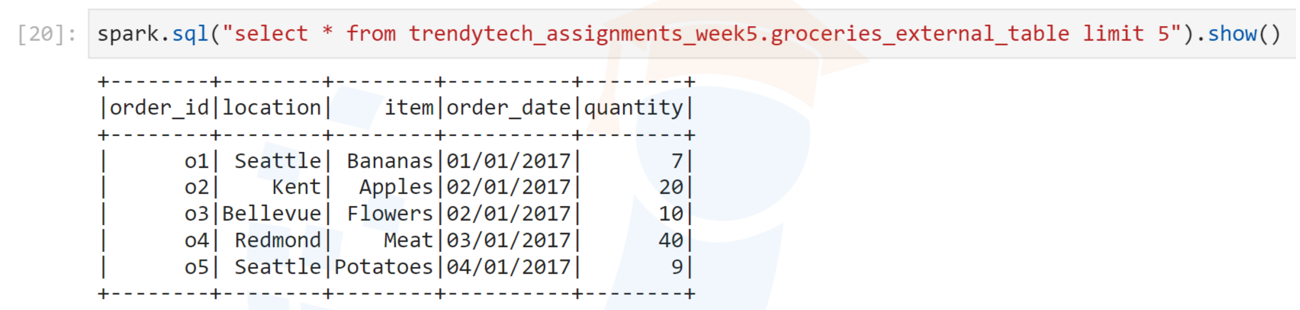
```
[18]: spark.sql("describe extended trendytech_assignments_week5.groceries").show(truncate=False)

+----------------------------+------------------------------------------------------------------------------------------------+-------+
|col_name                    |data_type                                                                                       |comment|
+----------------------------+------------------------------------------------------------------------------------------------+-------+
|order_id                    |string                                                                                          |null   |
|location                    |string                                                                                          |null   |
|item                        |string                                                                                          |null   |
|order_date                  |string                                                                                          |null   |
|quantity                    |int                                                                                             |null   |
|                            |                                                                                                |       |
|# Detailed Table Information|                                                                                                |       |
|Database                    |trendytech_assignments_week5                                                                    |       |
|Table                       |groceries                                                                                       |       |
|Owner                       |itv006753                                                                                       |       |
|Created Time                |Tue May 16 02:40:46 EDT 2023                                                                    |       |
|Last Access                 |UNKNOWN                                                                                         |       |
|Created By                  |Spark 3.0.1                                                                                     |       |
|Type                        |MANAGED                                                                                         |       |
|Provider                    |hive                                                                                            |       |
|Table Properties            |[transient_lastDdlTime=1684219800]                                                              |       |
|Statistics                  |666 bytes                                                                                       |       |
|Location                    |hdfs://m01.itversity.com:9000/user/itv006753/warehouse/trendytech_assignments_week5.db/groceries|       |
|Serde Library               |org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe                                              |       |
|InputFormat                 |org.apache.hadoop.mapred.TextInputFormat                                                        |       |
+----------------------------+------------------------------------------------------------------------------------------------+-------+
```

1.2

# If the CSV file has a header row, set the header option to true to skip that row from being considered as data.

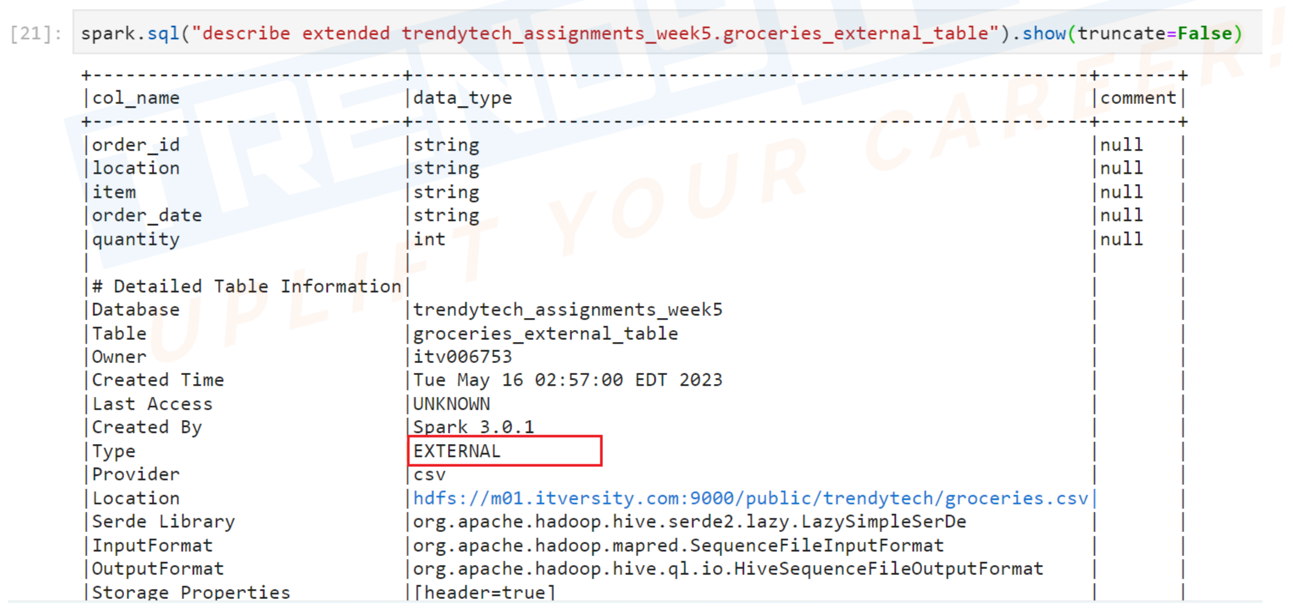
spark.sql("create table trendytech_assignments_week5.groceries_external_table(order_id string,location string,item string,order_date string,quantity integer) using csv options(header='true') location '/public/trendytech/groceries.csv'")

spark.sql("select * from trendytech_assignments_week5.groceries_external_table limit 5").show()



```
[20]: spark.sql("select * from trendytech_assignments_week5.groceries_external_table limit 5").show()

+--------+--------+--------+----------+--------+
|order_id|location|    item|order_date|quantity|
+--------+--------+--------+----------+--------+
|      o1| Seattle| Bananas|01/01/2017|       7|
|      o2|    Kent|  Apples|02/01/2017|      20|
|      o3|Bellevue| Flowers|02/01/2017|      10|
|      o4| Redmond|    Meat|03/01/2017|      40|
|      o5| Seattle|Potatoes|04/01/2017|       9|
+--------+--------+--------+----------+--------+
```

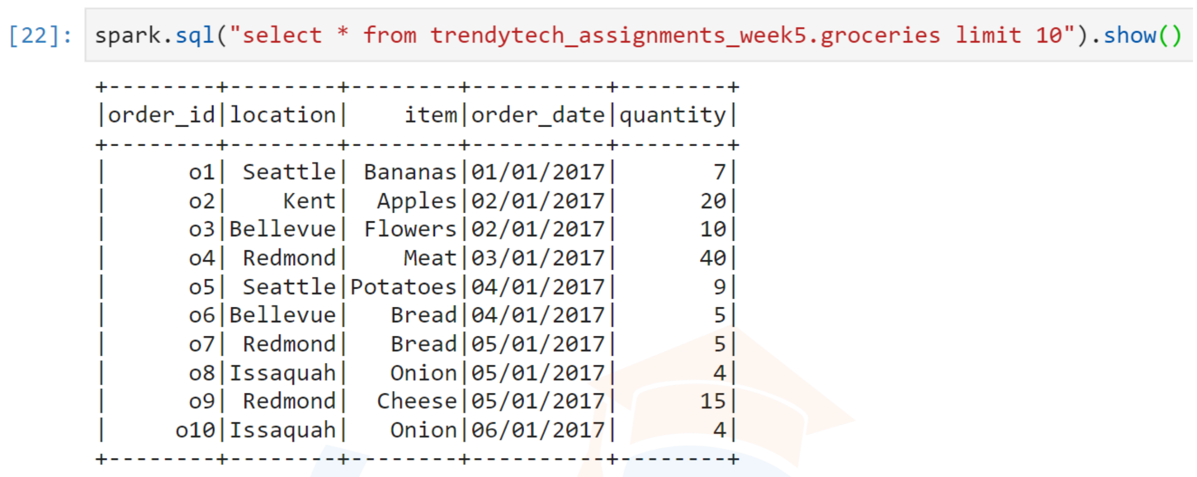
spark.sql("describe extended trendytech_assignments_week5.groceries_external_table").show(truncate=False)



```
[21]: spark.sql("describe extended trendytech_assignments_week5.groceries_external_table").show(truncate=False)

+----------------------------+--------------------------------------------------------------+-------+
|col_name                    |data_type                                                     |comment|
+----------------------------+--------------------------------------------------------------+-------+
|order_id                    |string                                                        |null   |
|location                    |string                                                        |null   |
|item                        |string                                                        |null   |
|order_date                  |string                                                        |null   |
|quantity                    |int                                                           |null   |
|                            |                                                              |       |
|# Detailed Table Information|                                                              |       |
|Database                    |trendytech_assignments_week5                                  |       |
|Table                       |groceries_external_table                                      |       |
|Owner                       |itv006753                                                     |       |
|Created Time                |Tue May 16 02:57:00 EDT 2023                                  |       |
|Last Access                 |UNKNOWN                                                       |       |
|Created By                  |Spark 3.0.1                                                   |       |
|Type                        |EXTERNAL                                                      |       |
|Provider                    |csv                                                           |       |
|Location                    |hdfs://m01.itversity.com:9000/public/trendytech/groceries.csv|       |
|Serde Library               |org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe            |       |
|InputFormat                 |org.apache.hadoop.mapred.SequenceFileInputFormat              |       |
|OutputFormat                |org.apache.hadoop.hive.ql.io.HiveSequenceFileOutputFormat     |       |
|Storage Properties          |[header=true]                                                 |       |
```

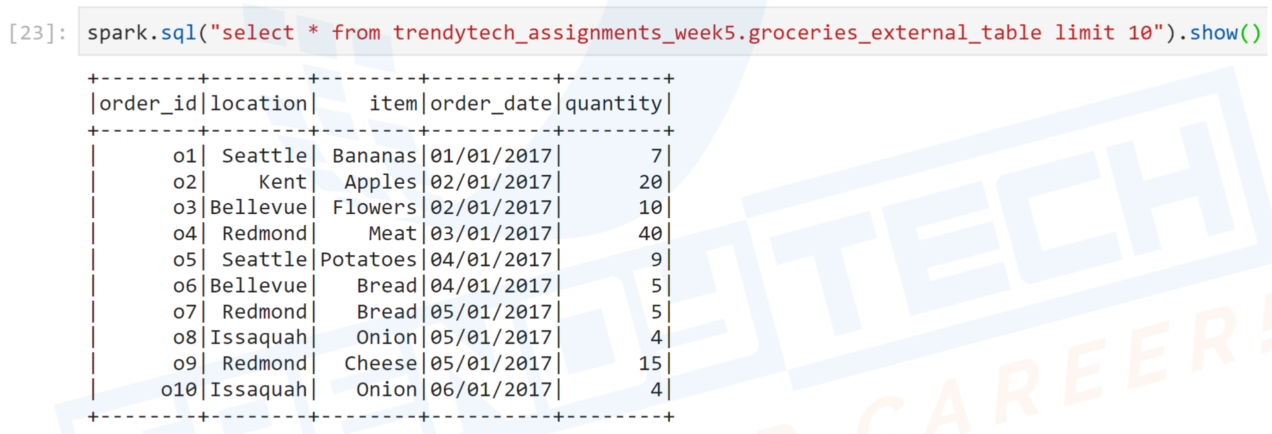
1.3

spark.sql("select * from trendytech_assignments_week5.groceries limit 10").show()



```
[22]: spark.sql("select * from trendytech_assignments_week5.groceries limit 10").show()

+--------+--------+--------+----------+--------+
|order_id|location|    item|order_date|quantity|
+--------+--------+--------+----------+--------+
|      o1| Seattle| Bananas|01/01/2017|       7|
|      o2|    Kent|  Apples|02/01/2017|      20|
|      o3|Bellevue| Flowers|02/01/2017|      10|
|      o4| Redmond|    Meat|03/01/2017|      40|
|      o5| Seattle|Potatoes|04/01/2017|       9|
|      o6|Bellevue|   Bread|04/01/2017|       5|
|      o7| Redmond|   Bread|05/01/2017|       5|
|      o8|Issaquah|   Onion|05/01/2017|       4|
|      o9| Redmond|  Cheese|05/01/2017|      15|
|     o10|Issaquah|   Onion|06/01/2017|       4|
+--------+--------+--------+----------+--------+
```

spark.sql("select * from trendytech_assignments_week5.groceries_external_table limit 10").show()



```
[23]: spark.sql("select * from trendytech_assignments_week5.groceries_external_table limit 10").show()

+--------+--------+--------+----------+--------+
|order_id|location|    item|order_date|quantity|
+--------+--------+--------+----------+--------+
|      o1| Seattle| Bananas|01/01/2017|       7|
|      o2|    Kent|  Apples|02/01/2017|      20|
|      o3|Bellevue| Flowers|02/01/2017|      10|
|      o4| Redmond|    Meat|03/01/2017|      40|
|      o5| Seattle|Potatoes|04/01/2017|       9|
|      o6|Bellevue|   Bread|04/01/2017|       5|
|      o7| Redmond|   Bread|05/01/2017|       5|
|      o8|Issaquah|   Onion|05/01/2017|       4|
|      o9| Redmond|  Cheese|05/01/2017|      15|
|     o10|Issaquah|   Onion|06/01/2017|       4|
+--------+--------+--------+----------+--------+
```

1.4

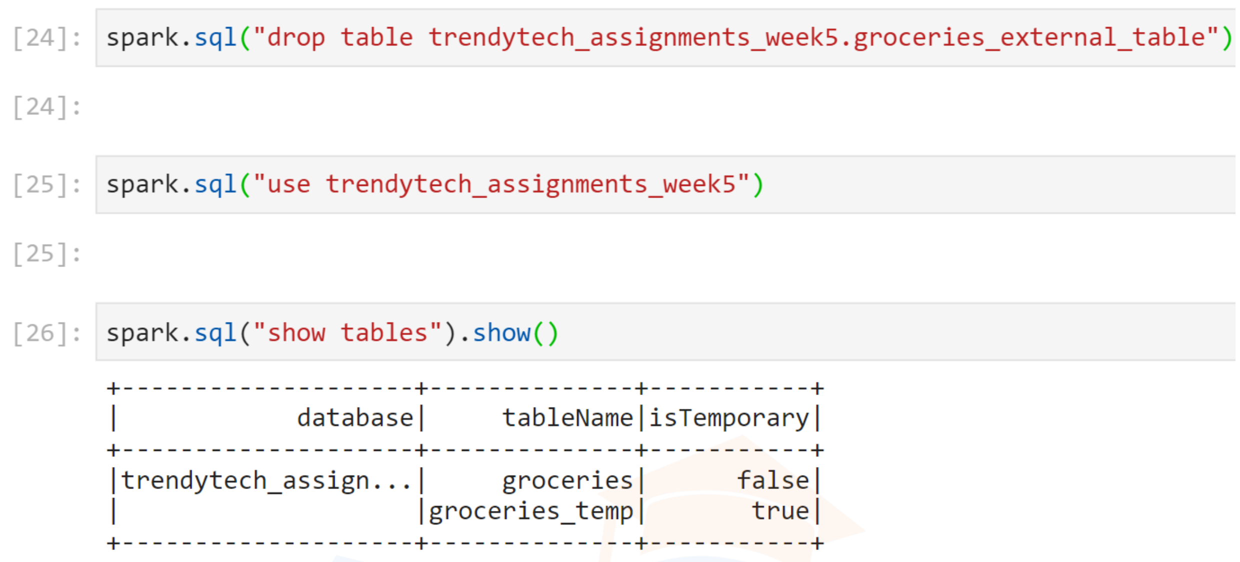
Dropping external table

Here only the metadata gets deleted and data still resides in the HDFS path.

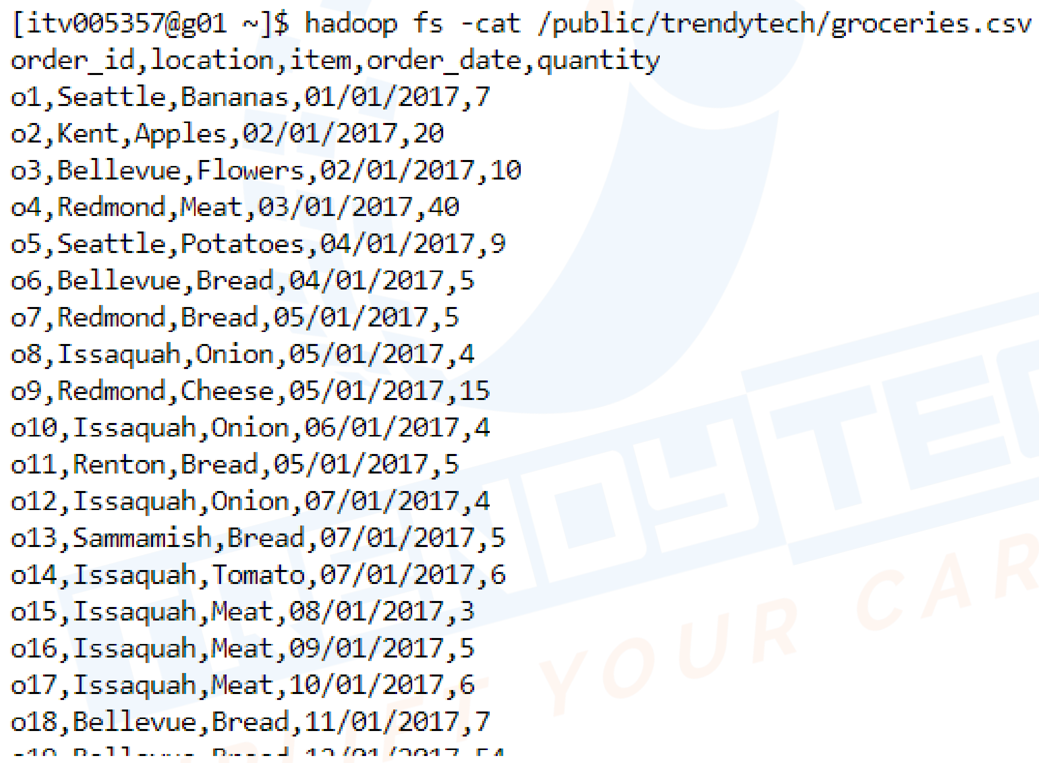
spark.sql("drop table trendytech_assignments_week5.groceries_external_table")

spark.sql("use trendytech_assignments_week5")

spark.sql("show tables").show()



```
[24]: spark.sql("drop table trendytech_assignments_week5.groceries_external_table")

[24]:

[25]: spark.sql("use trendytech_assignments_week5")

[25]:

[26]: spark.sql("show tables").show()

      +--------------------+-------------+-----------+
      |            database|    tableName|isTemporary|
      +--------------------+-------------+-----------+
      |trendytech_assign...|    groceries|      false|
      |                    |groceries_temp|       true|
      +--------------------+-------------+-----------+
```



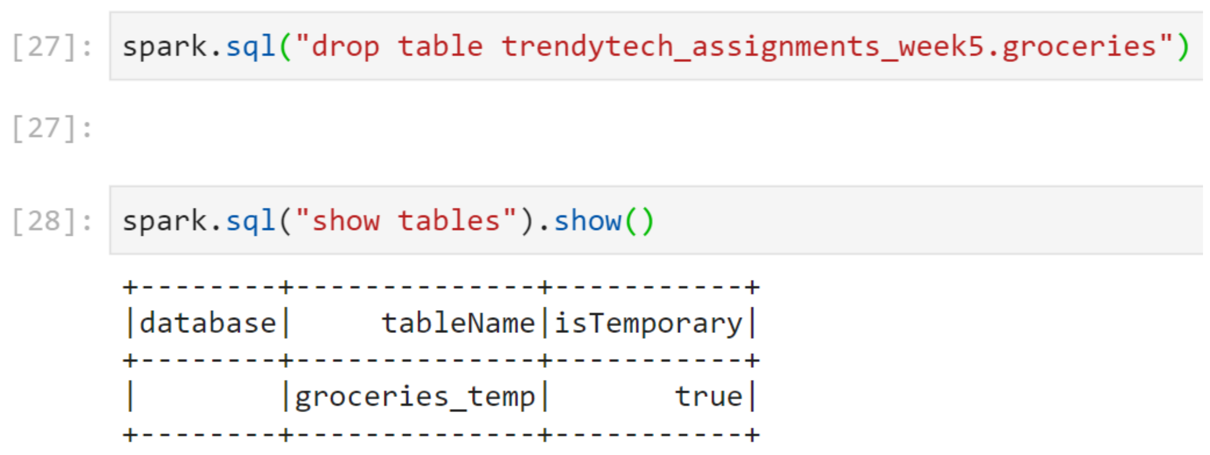
```
[itv005357@g01 ~]$ hadoop fs -cat /public/trendytech/groceries.csv
order_id,location,item,order_date,quantity
o1,Seattle,Bananas,01/01/2017,7
o2,Kent,Apples,02/01/2017,20
o3,Bellevue,Flowers,02/01/2017,10
o4,Redmond,Meat,03/01/2017,40
o5,Seattle,Potatoes,04/01/2017,9
o6,Bellevue,Bread,04/01/2017,5
o7,Redmond,Bread,05/01/2017,5
o8,Issaquah,Onion,05/01/2017,4
o9,Redmond,Cheese,05/01/2017,15
o10,Issaquah,Onion,06/01/2017,4
o11,Renton,Bread,05/01/2017,5
o12,Issaquah,Onion,07/01/2017,4
o13,Sammamish,Bread,07/01/2017,5
o14,Issaquah,Tomato,07/01/2017,6
o15,Issaquah,Meat,08/01/2017,3
o16,Issaquah,Meat,09/01/2017,5
o17,Issaquah,Meat,10/01/2017,6
o18,Bellevue,Bread,11/01/2017,7
o19,Bellevue,Bread,12/01/2017,54
```

Dropping Managed table

Here both the metadata and data gets deleted.

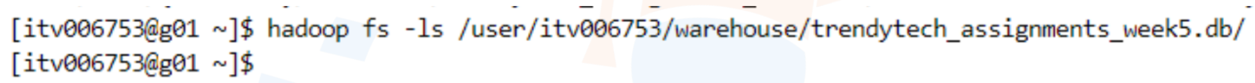
spark.sql("drop table trendytech_assignments_week5.groceries")

spark.sql("show tables").show()



```
[27]: spark.sql("drop table trendytech_assignments_week5.groceries")

[27]:

[28]: spark.sql("show tables").show()

      +--------+-------------+-----------+
      |database|    tableName|isTemporary|
      +--------+-------------+-----------+
      |        |groceries_temp|       true|
      +--------+-------------+-----------+
```

hadoop fs -ls
/user/<username>/warehouse/trendytech_assignments_week5.db/



```
[itv006753@g01 ~]$ hadoop fs -ls /user/itv006753/warehouse/trendytech_assignments_week5.db/
[itv006753@g01 ~]$
```

Here you can see inside the data also gets deleted.

1.5

#Performing all the above tasks with the json file

1.5.1

spark.sql("create database if not exists trendytech_assignments_week5")

spark.sql("use trendytech_assignments_week5")

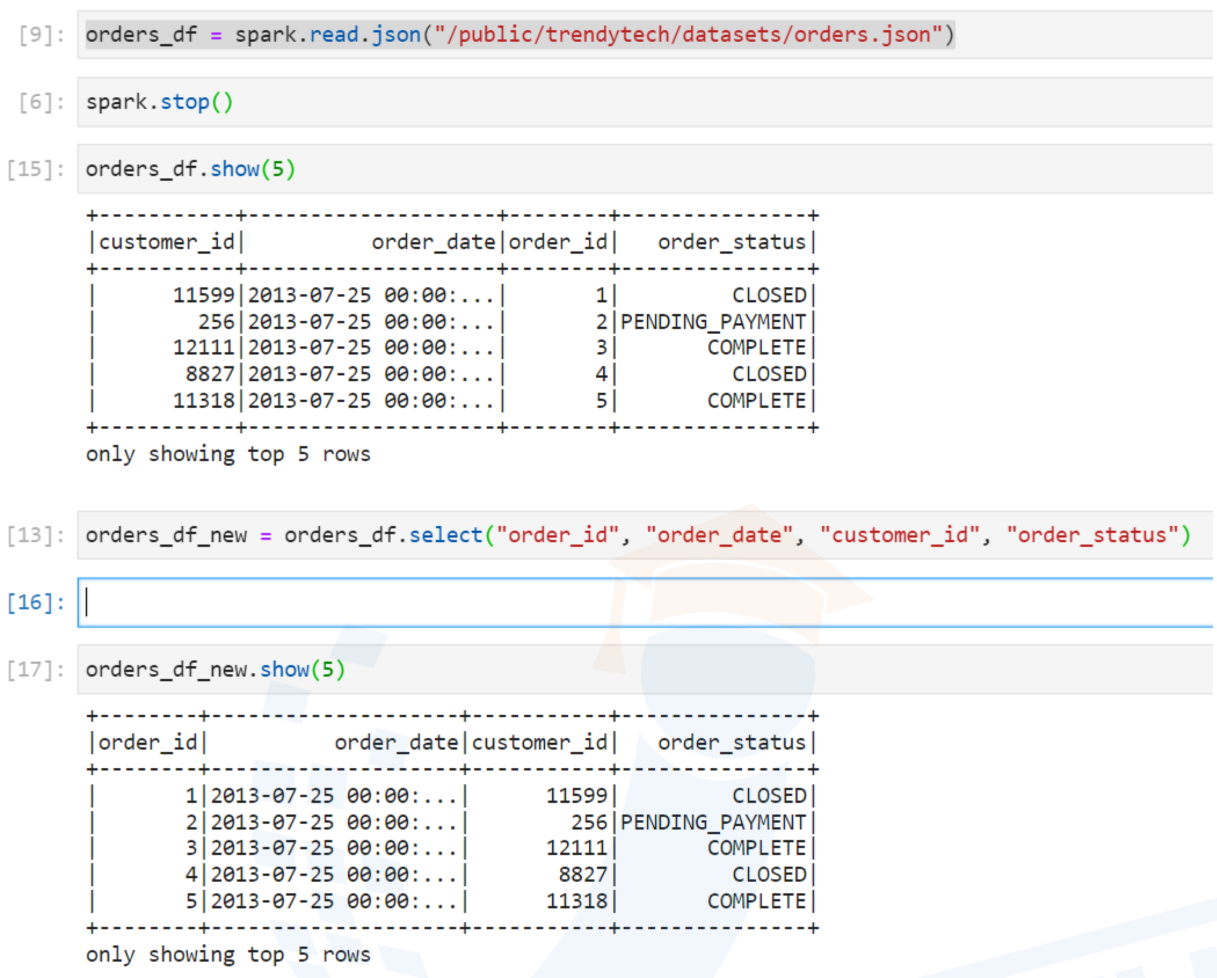
spark.sql("create table trendytech_assignments_week5.orders(order_id integer,order_date string,customer_id integer,order_status string)")

orders_df = spark.read.json("/public/trendytech/datasets/orders.json")

#we are giving select() and giving the column names because there can be chance of column names displayed in dis-ordered form.

orders_df_new = orders_df.select("order_id", "order_date", "customer_id", "order_status")

orders_df_new.show(5)



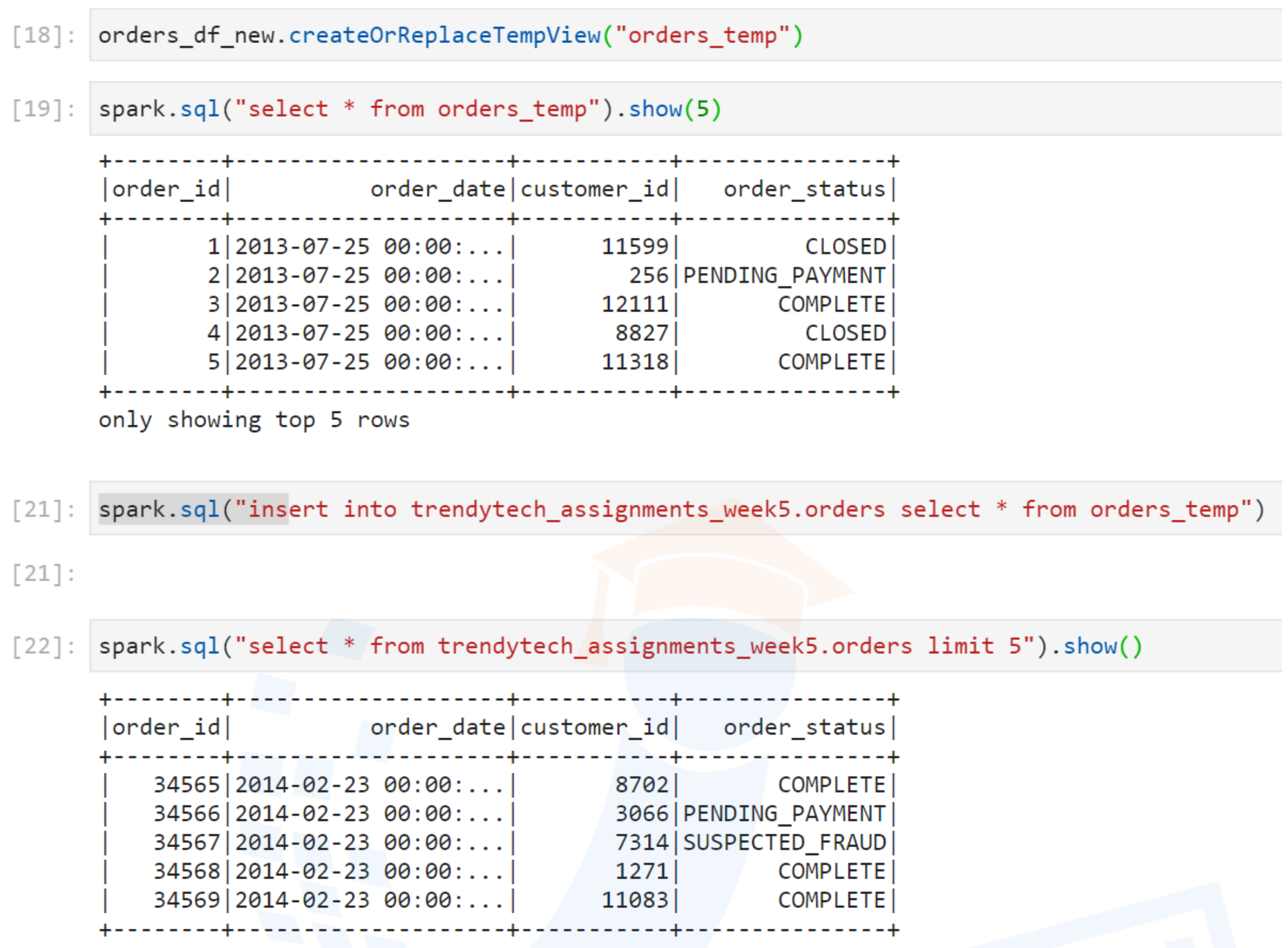
```
[9]:  orders_df = spark.read.json("/public/trendytech/datasets/orders.json")

[6]:  spark.stop()

[15]: orders_df.show(5)

      +-----------+--------------------+--------+---------------+
      |customer_id|          order_date|order_id|   order_status|
      +-----------+--------------------+--------+---------------+
      |      11599|2013-07-25 00:00:...|       1|         CLOSED|
      |        256|2013-07-25 00:00:...|       2|PENDING_PAYMENT|
      |      12111|2013-07-25 00:00:...|       3|       COMPLETE|
      |       8827|2013-07-25 00:00:...|       4|         CLOSED|
      |      11318|2013-07-25 00:00:...|       5|       COMPLETE|
      +-----------+--------------------+--------+---------------+
      only showing top 5 rows

[13]: orders_df_new = orders_df.select("order_id", "order_date", "customer_id", "order_status")

[16]: |

[17]: orders_df_new.show(5)

      +--------+--------------------+-----------+---------------+
      |order_id|          order_date|customer_id|   order_status|
      +--------+--------------------+-----------+---------------+
      |       1|2013-07-25 00:00:...|      11599|         CLOSED|
      |       2|2013-07-25 00:00:...|        256|PENDING_PAYMENT|
      |       3|2013-07-25 00:00:...|      12111|       COMPLETE|
      |       4|2013-07-25 00:00:...|       8827|         CLOSED|
      |       5|2013-07-25 00:00:...|      11318|       COMPLETE|
      +--------+--------------------+-----------+---------------+
      only showing top 5 rows
```

orders_df_new.createOrReplaceTempView("orders_temp")

spark.sql("select * from orders_temp").show(5)

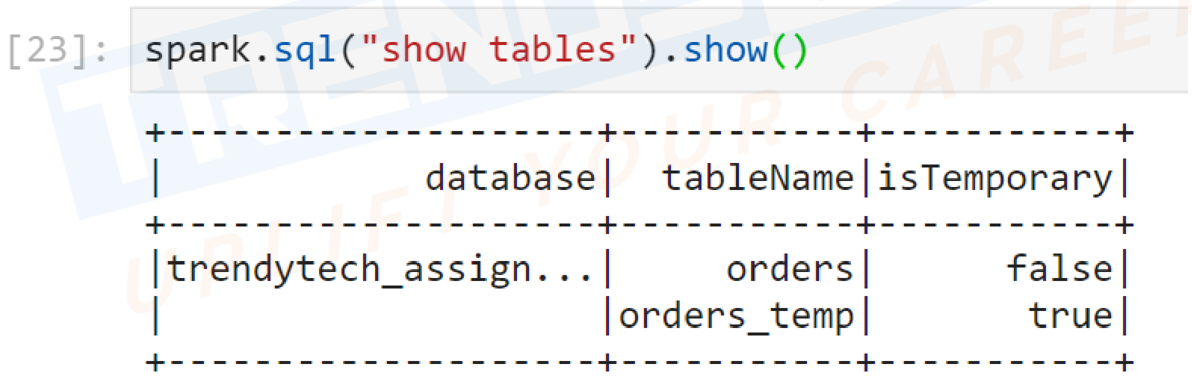
spark.sql("insert into trendytech_assignments_week5.orders select * from orders_temp")]

spark.sql("select * from trendytech_assignments_week5.orders limit 5").show()



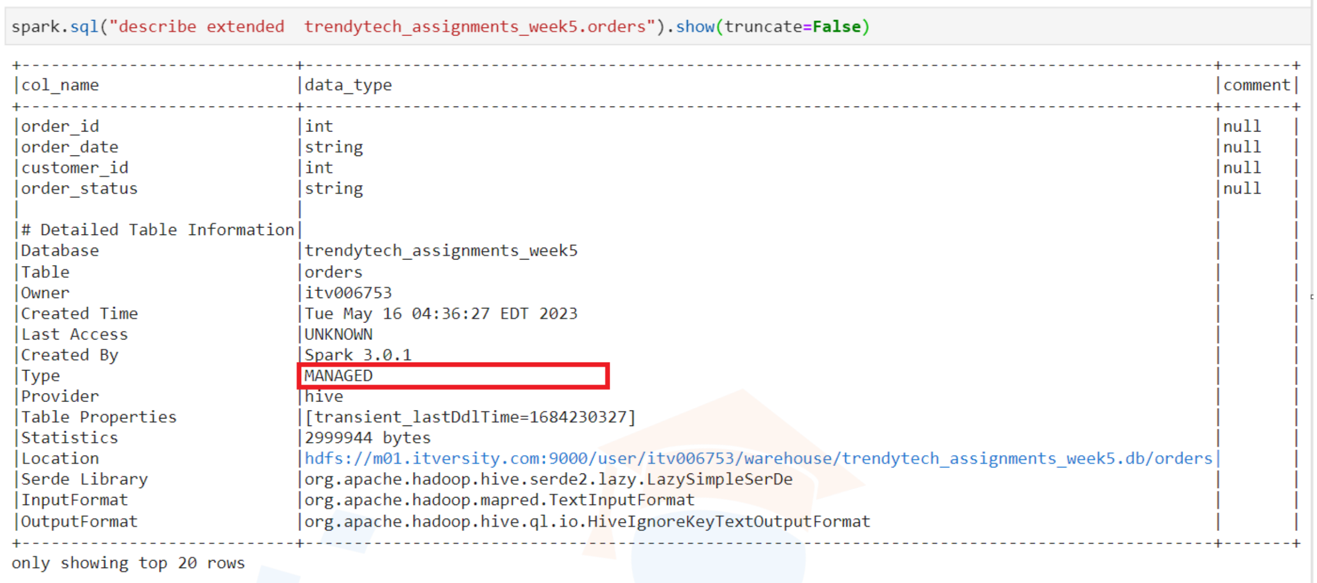
```
[18]: orders_df_new.createOrReplaceTempView("orders_temp")

[19]: spark.sql("select * from orders_temp").show(5)

+--------+-------------------+-----------+---------------+
|order_id|         order_date|customer_id|   order_status|
+--------+-------------------+-----------+---------------+
|       1|2013-07-25 00:00:...|      11599|         CLOSED|
|       2|2013-07-25 00:00:...|        256|PENDING_PAYMENT|
|       3|2013-07-25 00:00:...|      12111|       COMPLETE|
|       4|2013-07-25 00:00:...|       8827|         CLOSED|
|       5|2013-07-25 00:00:...|      11318|       COMPLETE|
+--------+-------------------+-----------+---------------+
only showing top 5 rows

[21]: spark.sql("insert into trendytech_assignments_week5.orders select * from orders_temp")

[21]:

[22]: spark.sql("select * from trendytech_assignments_week5.orders limit 5").show()

+--------+-------------------+-----------+---------------+
|order_id|         order_date|customer_id|   order_status|
+--------+-------------------+-----------+---------------+
|   34565|2014-02-23 00:00:...|       8702|       COMPLETE|
|   34566|2014-02-23 00:00:...|       3066|PENDING_PAYMENT|
|   34567|2014-02-23 00:00:...|       7314|SUSPECTED_FRAUD|
|   34568|2014-02-23 00:00:...|       1271|       COMPLETE|
|   34569|2014-02-23 00:00:...|      11083|       COMPLETE|
+--------+-------------------+-----------+---------------+
```

spark.sql("show tables").show()



```
[23]: spark.sql("show tables").show()

+--------------------+-----------+-----------+
|            database|  tableName|isTemporary|
+--------------------+-----------+-----------+
|trendytech_assign...|     orders|      false|
|                    |orders_temp|       true|
+--------------------+-----------+-----------+
```

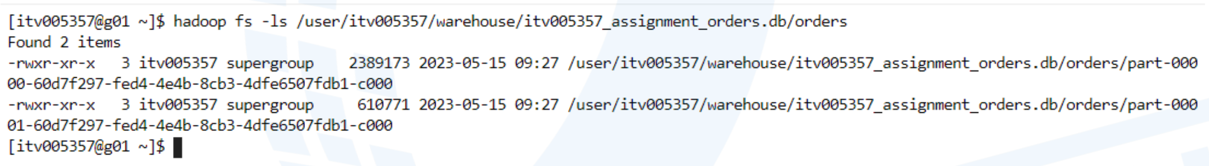
spark.sql("describe extended trendytech_assignments_week5.orders").show(truncate=False)



```
spark.sql("describe extended  trendytech_assignments_week5.orders").show(truncate=False)
```

```
+----------------------------+---------------------------------------------------------------------------------------------+-------+
|col_name                    |data_type                                                                                    |comment|
+----------------------------+---------------------------------------------------------------------------------------------+-------+
|order_id                    |int                                                                                          |null   |
|order_date                  |string                                                                                       |null   |
|customer_id                 |int                                                                                          |null   |
|order_status                |string                                                                                       |null   |
|                            |                                                                                             |       |
|# Detailed Table Information|                                                                                             |       |
|Database                    |trendytech_assignments_week5                                                                 |       |
|Table                       |orders                                                                                       |       |
|Owner                       |itv006753                                                                                    |       |
|Created Time                |Tue May 16 04:36:27 EDT 2023                                                                 |       |
|Last Access                 |UNKNOWN                                                                                      |       |
|Created By                  |Spark 3.0.1                                                                                  |       |
|Type                        |MANAGED                                                                                      |       |
|Provider                    |hive                                                                                         |       |
|Table Properties            |[transient_lastDdlTime=1684230327]                                                           |       |
|Statistics                  |2999944 bytes                                                                                |       |
|Location                    |hdfs://m01.itversity.com:9000/user/itv006753/warehouse/trendytech_assignments_week5.db/orders|       |
|Serde Library               |org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe                                           |       |
|InputFormat                 |org.apache.hadoop.mapred.TextInputFormat                                                     |       |
|OutputFormat                |org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat                                   |       |
+----------------------------+---------------------------------------------------------------------------------------------+-------+
only showing top 20 rows
```

Data is present in the HDFS Location

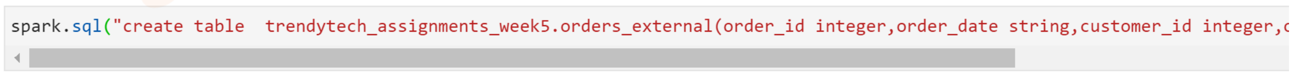


```
[itv005357@g01 ~]$ hadoop fs -ls /user/itv005357/warehouse/itv005357_assignment_orders.db/orders
Found 2 items
-rwxr-xr-x   3 itv005357 supergroup    2389173 2023-05-15 09:27 /user/itv005357/warehouse/itv005357_assignment_orders.db/orders/part-000
00-60d7f297-fed4-4e4b-8cb3-4dfe6507fdb1-c000
-rwxr-xr-x   3 itv005357 supergroup     610771 2023-05-15 09:27 /user/itv005357/warehouse/itv005357_assignment_orders.db/orders/part-000
01-60d7f297-fed4-4e4b-8cb3-4dfe6507fdb1-c000
[itv005357@g01 ~]$ 
```

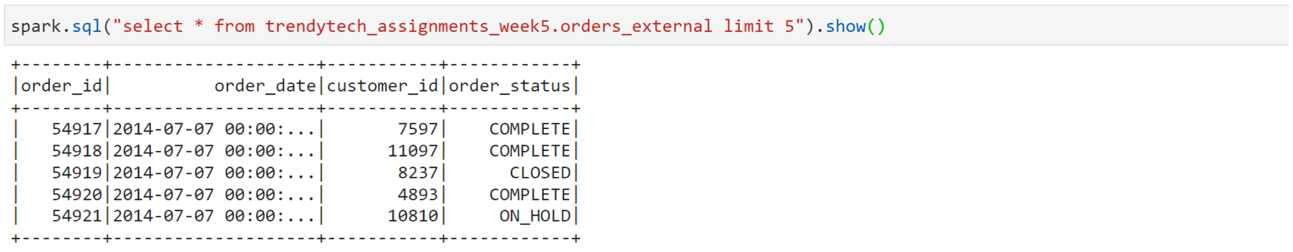
1.5.2

spark.sql("create table trendytech_assignments_week5.orders_external(order_id integer,order_date string,customer_id integer,order_status string) using json location '/public/trendytech/datasets/orders.json'")

spark.sql("select * from trendytech_assignments_week5.orders_external limit 5").show()



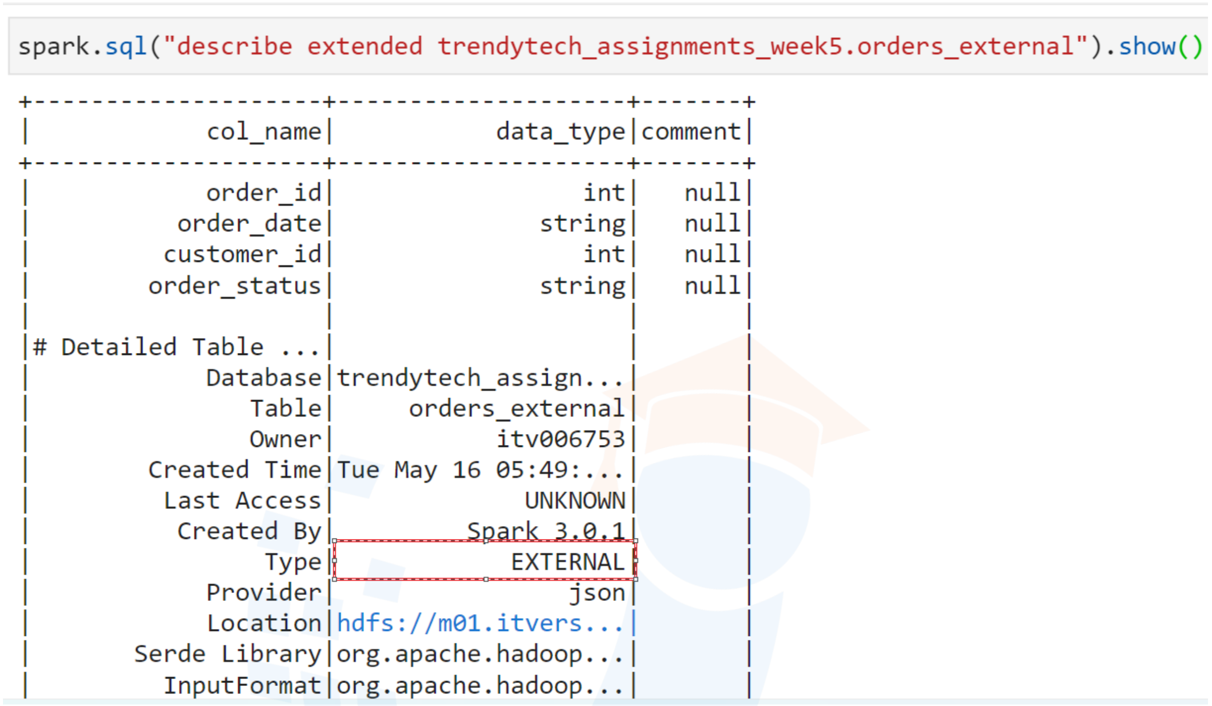
```
spark.sql("create table  trendytech_assignments_week5.orders_external(order_id integer,order_date string,customer_id integer,c
```



```
spark.sql("select * from trendytech_assignments_week5.orders_external limit 5").show()
```

```
+--------+--------------------+-----------+------------+
|order_id|          order_date|customer_id|order_status|
+--------+--------------------+-----------+------------+
|   54917|2014-07-07 00:00:...|       7597|    COMPLETE|
|   54918|2014-07-07 00:00:...|      11097|    COMPLETE|
|   54919|2014-07-07 00:00:...|       8237|      CLOSED|
|   54920|2014-07-07 00:00:...|       4893|    COMPLETE|
|   54921|2014-07-07 00:00:...|      10810|     ON_HOLD|
+--------+--------------------+-----------+------------+
```

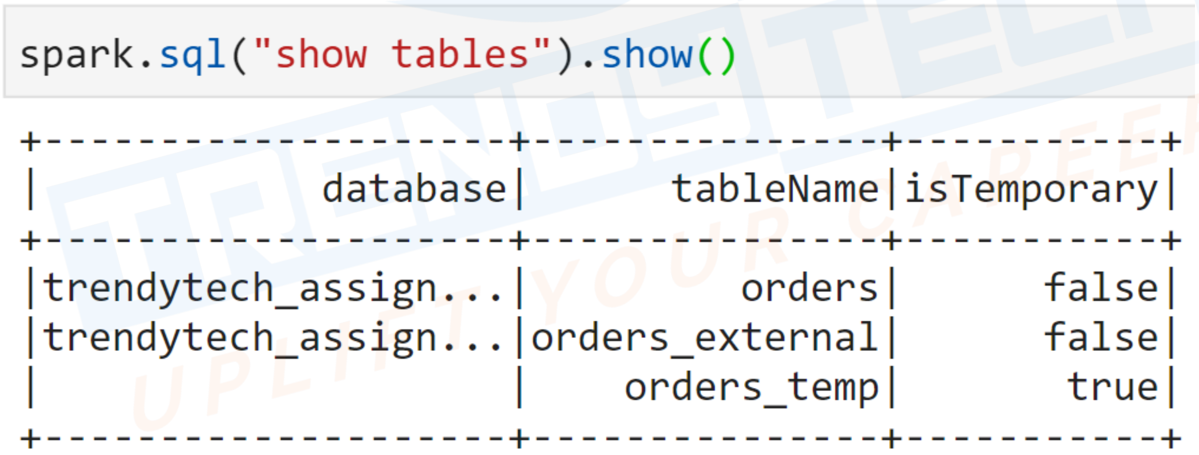
spark.sql("describe extended trendytech_assignments_week5.orders_external").show()



```
spark.sql("describe extended trendytech_assignments_week5.orders_external").show()
```

```
+--------------------+--------------------+-------+
|            col_name|           data_type|comment|
+--------------------+--------------------+-------+
|            order_id|                 int|   null|
|          order_date|              string|   null|
|         customer_id|                 int|   null|
|        order_status|              string|   null|
|                    |                    |       |
|# Detailed Table ...|                    |       |
|            Database|trendytech_assign...|       |
|               Table|     orders_external|       |
|               Owner|           itv006753|       |
|        Created Time|Tue May 16 05:49:...|       |
|         Last Access|             UNKNOWN|       |
|          Created By|         Spark 3.0.1|       |
|                Type|            EXTERNAL|       |
|            Provider|                json|       |
|            Location|hdfs://m01.itvers...|       |
|       Serde Library|org.apache.hadoop...|       |
|         InputFormat|org.apache.hadoop...|       |
```

spark.sql("show tables").show()

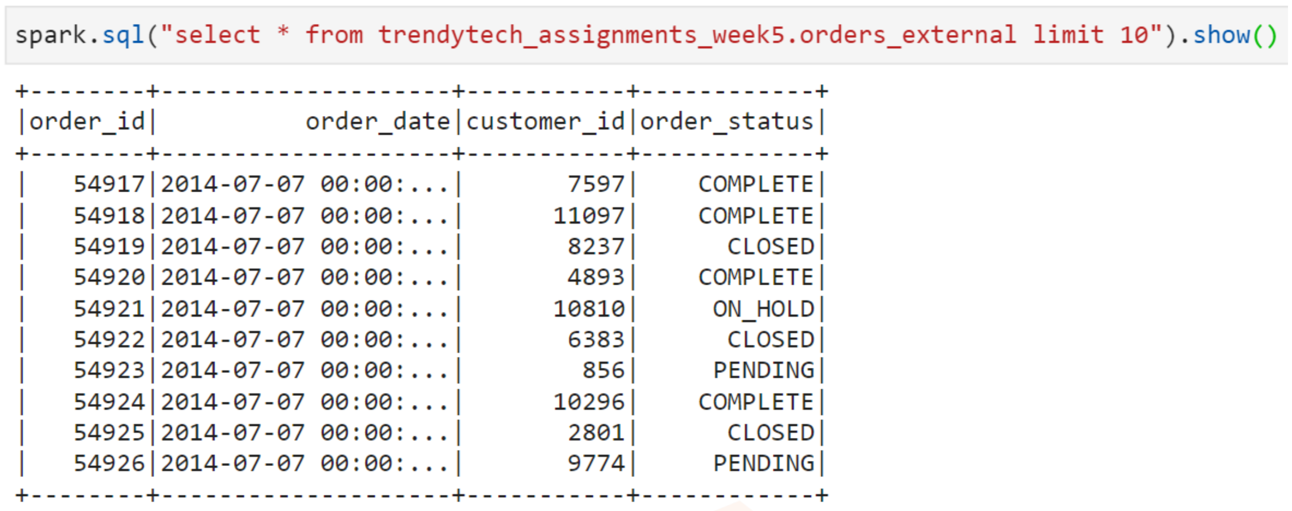


```
spark.sql("show tables").show()
```

```
+--------------------+---------------+-----------+
|            database|      tableName|isTemporary|
+--------------------+---------------+-----------+
|trendytech_assign...|         orders|      false|
|trendytech_assign...|orders_external|      false|
|                    |    orders_temp|       true|
+--------------------+---------------+-----------+
```

1.5.3

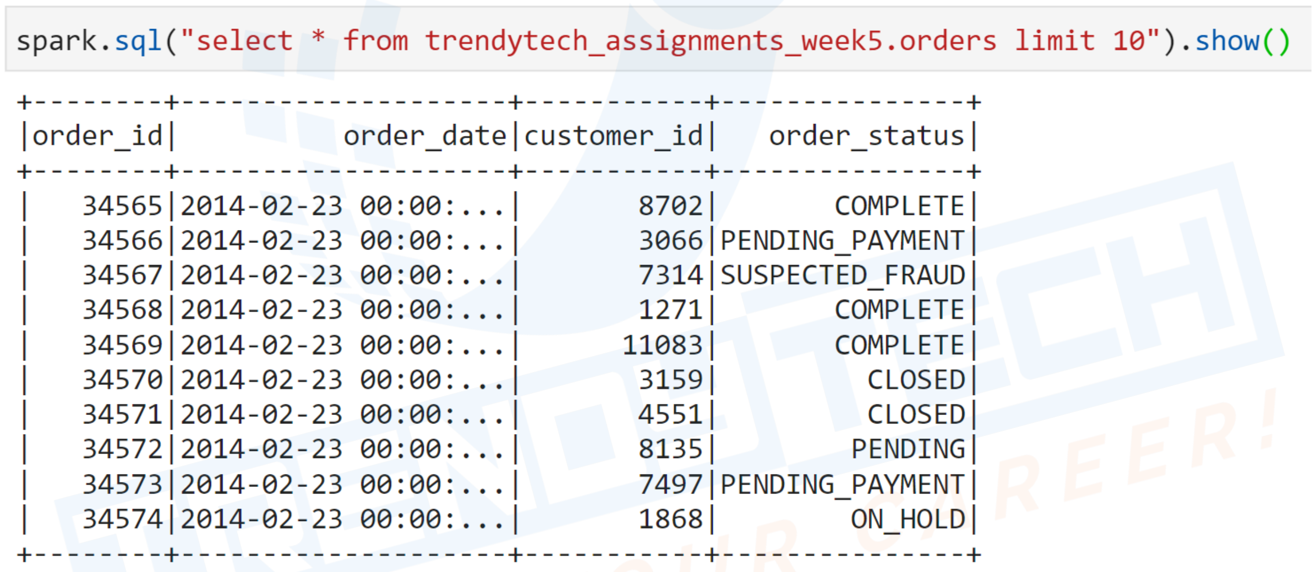
spark.sql("select * from trendytech_assignments_week5.orders_external limit 10").show()



```
spark.sql("select * from trendytech_assignments_week5.orders_external limit 10").show()
```

```
+--------+--------------------+-----------+------------+
|order_id|          order_date|customer_id|order_status|
+--------+--------------------+-----------+------------+
|   54917|2014-07-07 00:00:...|       7597|    COMPLETE|
|   54918|2014-07-07 00:00:...|      11097|    COMPLETE|
|   54919|2014-07-07 00:00:...|       8237|      CLOSED|
|   54920|2014-07-07 00:00:...|       4893|    COMPLETE|
|   54921|2014-07-07 00:00:...|      10810|     ON_HOLD|
|   54922|2014-07-07 00:00:...|       6383|      CLOSED|
|   54923|2014-07-07 00:00:...|        856|     PENDING|
|   54924|2014-07-07 00:00:...|      10296|    COMPLETE|
|   54925|2014-07-07 00:00:...|       2801|      CLOSED|
|   54926|2014-07-07 00:00:...|       9774|     PENDING|
+--------+--------------------+-----------+------------+
```

spark.sql("select * from trendytech_assignments_week5.orders limit 10").show()



```
spark.sql("select * from trendytech_assignments_week5.orders limit 10").show()
```

```
+--------+--------------------+-----------+---------------+
|order_id|          order_date|customer_id|   order_status|
+--------+--------------------+-----------+---------------+
|   34565|2014-02-23 00:00:...|       8702|       COMPLETE|
|   34566|2014-02-23 00:00:...|       3066|PENDING_PAYMENT|
|   34567|2014-02-23 00:00:...|       7314|SUSPECTED_FRAUD|
|   34568|2014-02-23 00:00:...|       1271|       COMPLETE|
|   34569|2014-02-23 00:00:...|      11083|       COMPLETE|
|   34570|2014-02-23 00:00:...|       3159|         CLOSED|
|   34571|2014-02-23 00:00:...|       4551|         CLOSED|
|   34572|2014-02-23 00:00:...|       8135|        PENDING|
|   34573|2014-02-23 00:00:...|       7497|PENDING_PAYMENT|
|   34574|2014-02-23 00:00:...|       1868|        ON_HOLD|
+--------+--------------------+-----------+---------------+
```

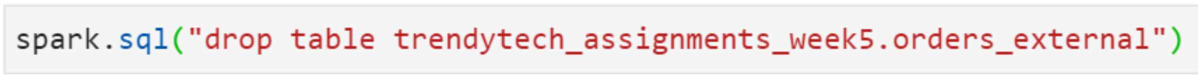
1.5.4

Dropping external table

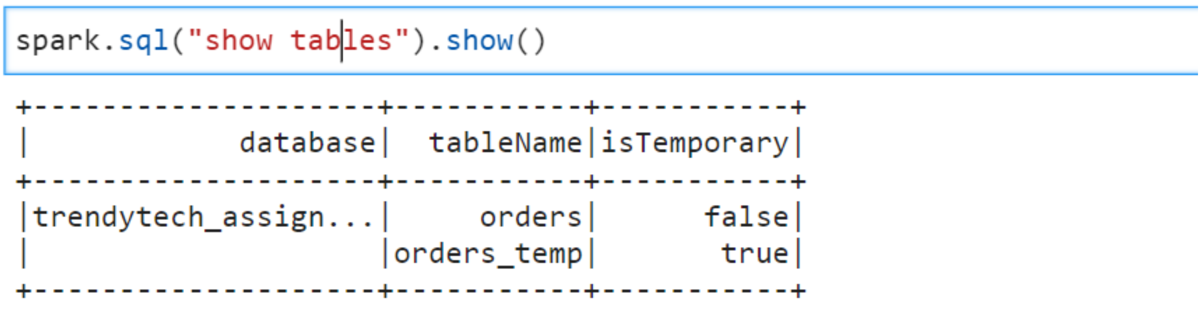
Here only the metadata gets deleted and data still resides in the HDFS path.

spark.sql("drop table trendytech_assignments_week5.orders_external")

spark.sql("show tables").show()

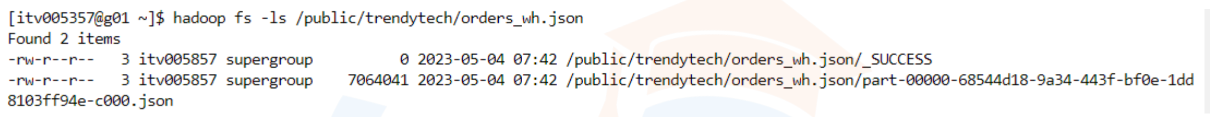


```
spark.sql("drop table trendytech_assignments_week5.orders_external")
```



```
spark.sql("show tables").show()
```

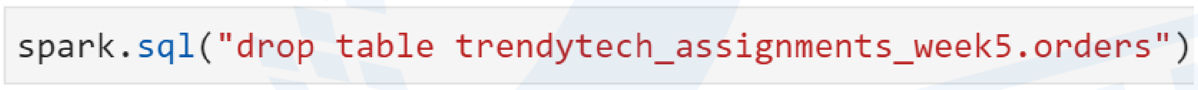
```
+--------------------+-----------+-----------+
|            database|  tableName|isTemporary|
+--------------------+-----------+-----------+
|trendytech_assign...|     orders|      false|
|                    |orders_temp|       true|
+--------------------+-----------+-----------+
```



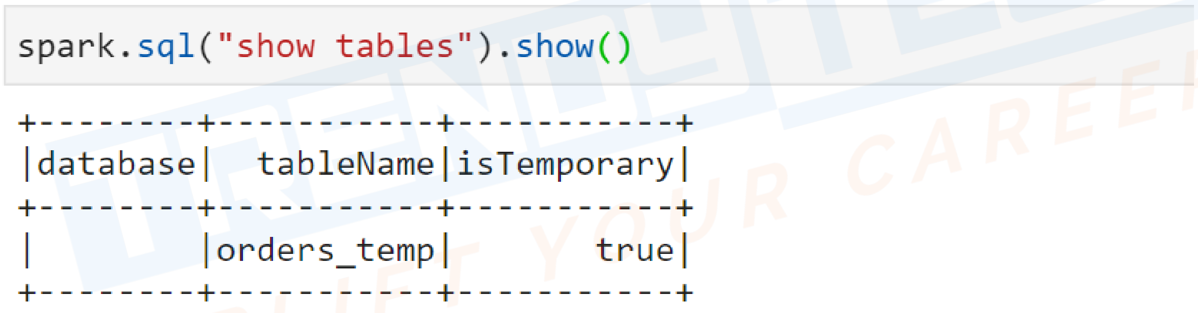
```
[itv005357@g01 ~]$ hadoop fs -ls /public/trendytech/orders_wh.json
Found 2 items
-rw-r--r--   3 itv005857 supergroup          0 2023-05-04 07:42 /public/trendytech/orders_wh.json/_SUCCESS
-rw-r--r--   3 itv005857 supergroup    7064041 2023-05-04 07:42 /public/trendytech/orders_wh.json/part-00000-68544d18-9a34-443f-bf0e-1dd
8103ff94e-c000.json
```

Dropping Managed table

Here both the metadata and data gets deleted.

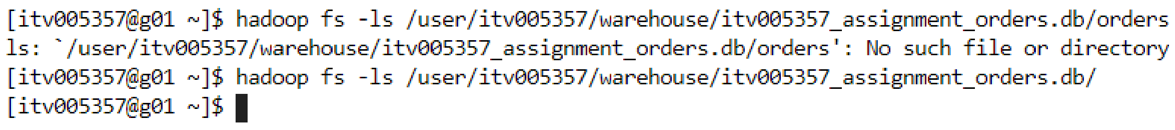


```
spark.sql("drop table trendytech_assignments_week5.orders")
```



```
spark.sql("show tables").show()
```

```
+--------+-----------+-----------+
|database|  tableName|isTemporary|
+--------+-----------+-----------+
|        |orders_temp|       true|
+--------+-----------+-----------+
```



```
[itv005357@g01 ~]$ hadoop fs -ls /user/itv005357/warehouse/itv005357_assignment_orders.db/orders
ls: `/user/itv005357/warehouse/itv005357_assignment_orders.db/orders': No such file or directory
[itv005357@g01 ~]$ hadoop fs -ls /user/itv005357/warehouse/itv005357_assignment_orders.db/
[itv005357@g01 ~]$ 
```

Question 2

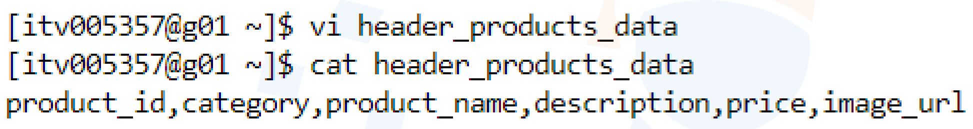
#First we are adding headers to the dataset, as there is no header in the file given. (This method of adding headers to the dataset is just for the demo purpose. We will be learning the industry standard way of adding the headers in the upcoming weeks.)

vi header_products_data

[Insert the following into the opened file -
product_id,category,product_name,description,price,image_url

]

cat header_products_data

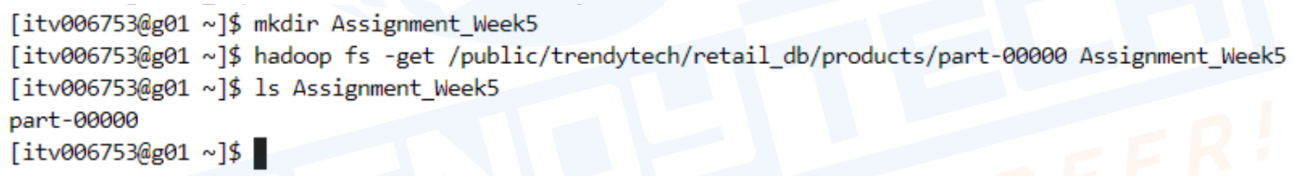


```
[itv005357@g01 ~]$ vi header_products_data
[itv005357@g01 ~]$ cat header_products_data
product_id,category,product_name,description,price,image_url
```

mkdir Week5_Assignment

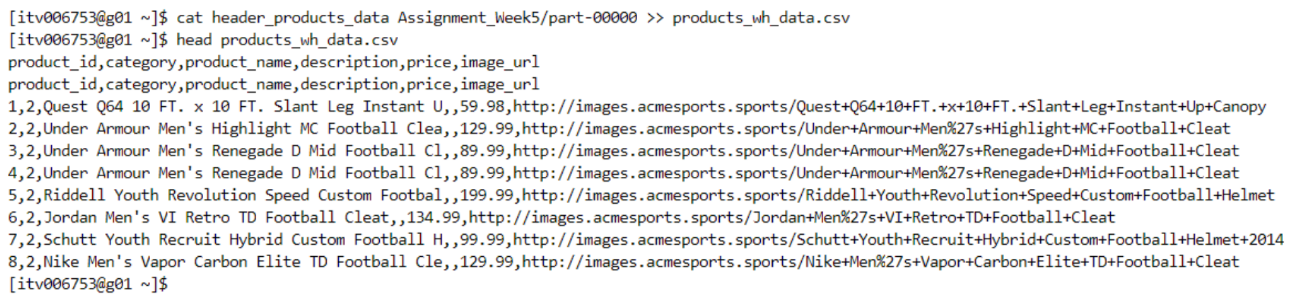
hadoop fs -get /public/trendytech/retail_db/products/part-00000 Week5_Assignment

ls Assignment_Week5



```
[itv006753@g01 ~]$ mkdir Assignment_Week5
[itv006753@g01 ~]$ hadoop fs -get /public/trendytech/retail_db/products/part-00000 Assignment_Week5
[itv006753@g01 ~]$ ls Assignment_Week5
part-00000
[itv006753@g01 ~]$ █
```

cat header_products_data Assignment_Week5/part-00000 >> products_wh_data.csv

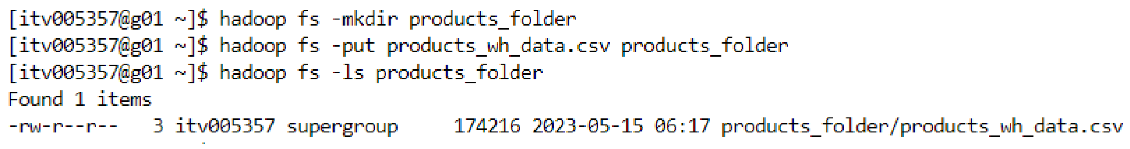


```
[itv006753@g01 ~]$ cat header_products_data Assignment_Week5/part-00000 >> products_wh_data.csv
[itv006753@g01 ~]$ head products_wh_data.csv
product_id,category,product_name,description,price,image_url
product_id,category,product_name,description,price,image_url
1,2,Quest Q64 10 FT. x 10 FT. Slant Leg Instant U,,59.98,http://images.acmesports.sports/Quest+Q64+10+FT.+x+10+FT.+Slant+Leg+Instant+Up+Canopy
2,2,Under Armour Men's Highlight MC Football Clea,,129.99,http://images.acmesports.sports/Under+Armour+Men%27s+Highlight+MC+Football+Cleat
3,2,Under Armour Men's Renegade D Mid Football Cl,,89.99,http://images.acmesports.sports/Under+Armour+Men%27s+Renegade+D+Mid+Football+Cleat
4,2,Under Armour Men's Renegade D Mid Football Cl,,89.99,http://images.acmesports.sports/Under+Armour+Men%27s+Renegade+D+Mid+Football+Cleat
5,2,Riddell Youth Revolution Speed Custom Footbal,,199.99,http://images.acmesports.sports/Riddell+Youth+Revolution+Speed+Custom+Football+Helmet
6,2,Jordan Men's VI Retro TD Football Cleat,,134.99,http://images.acmesports.sports/Jordan+Men%27s+VI+Retro+TD+Football+Cleat
7,2,Schutt Youth Recruit Hybrid Custom Football H,,99.99,http://images.acmesports.sports/Schutt+Youth+Recruit+Hybrid+Custom+Football+Helmet+2014
8,2,Nike Men's Vapor Carbon Elite TD Football Cle,,129.99,http://images.acmesports.sports/Nike+Men%27s+Vapor+Carbon+Elite+TD+Football+Cleat
[itv006753@g01 ~]$
```

hadoop fs -mkdir products_folder

hadoop fs -put products_wh_data.csv products_folder

hadoop fs -ls products_folder



```
[itv005357@g01 ~]$ hadoop fs -mkdir products_folder
[itv005357@g01 ~]$ hadoop fs -put products_wh_data.csv products_folder
[itv005357@g01 ~]$ hadoop fs -ls products_folder
Found 1 items
-rw-r--r--   3 itv005357 supergroup     174216 2023-05-15 06:17 products_folder/products_wh_data.csv
```

So, now the hdfs location will be -
/user/<username>/products_folder/products_wh_data.csv

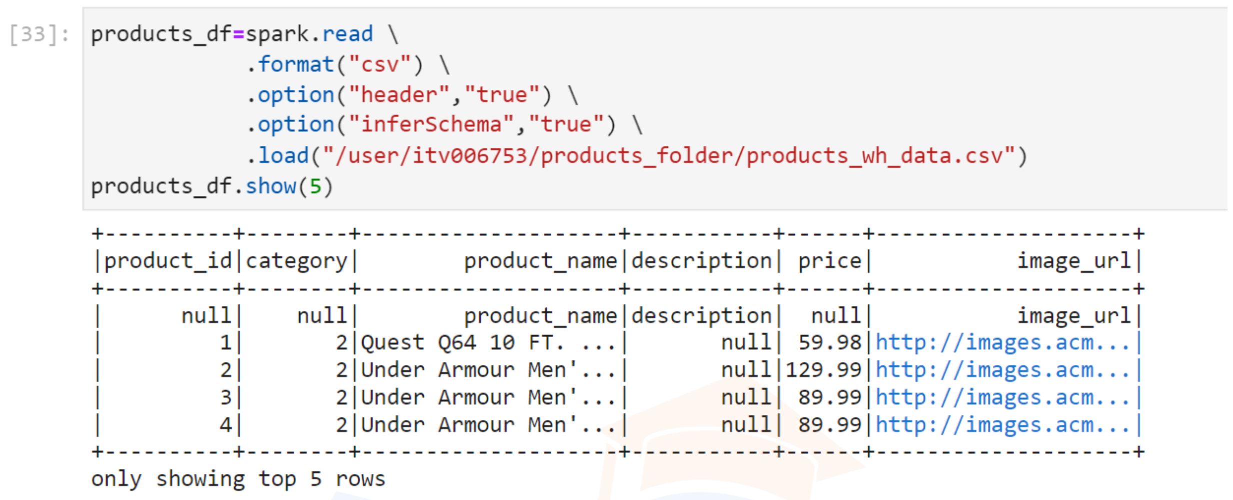
Boiler plate code:

```
from pyspark.sql import SparkSession
from pyspark.sql import SparkSession
import getpass
username = getpass.getuser()
spark= SparkSession. \
builder. \
config('spark.ui.port','0'). \
config("spark.sql.warehouse.dir", f"/user/{username}/warehouse"). \
enableHiveSupport(). \
master('yarn'). \
getOrCreate()
```

```
#Performing the tasks using Dataframe
products_df=spark.read \
        .format("csv") \
        .option("header","true") \
        .option("inferSchema","true") \
```

.load("/user/itv005357/products_folder/products_wh_data.csv")

products_df.show(5)

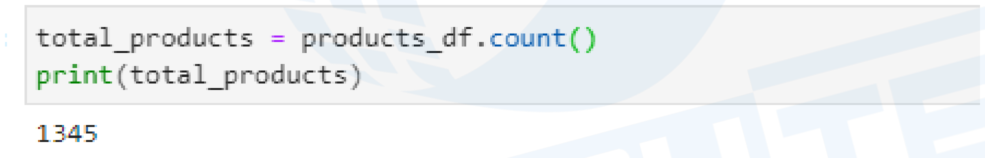


```
[33]: products_df=spark.read \
                .format("csv") \
                .option("header","true") \
                .option("inferSchema","true") \
                .load("/user/itv006753/products_folder/products_wh_data.csv")
      products_df.show(5)

      +----------+--------+--------------------+-----------+------+--------------------+
      |product_id|category|        product_name|description| price|           image_url|
      +----------+--------+--------------------+-----------+------+--------------------+
      |      null|    null|        product_name|description|  null|           image_url|
      |         1|       2|Quest Q64 10 FT. ...|       null| 59.98|http://images.acm...|
      |         2|       2|Under Armour Men'...|       null|129.99|http://images.acm...|
      |         3|       2|Under Armour Men'...|       null| 89.99|http://images.acm...|
      |         4|       2|Under Armour Men'...|       null| 89.99|http://images.acm...|
      +----------+--------+--------------------+-----------+------+--------------------+
      only showing top 5 rows
```

2.1

total_products = products_df.count()

print(total_products)

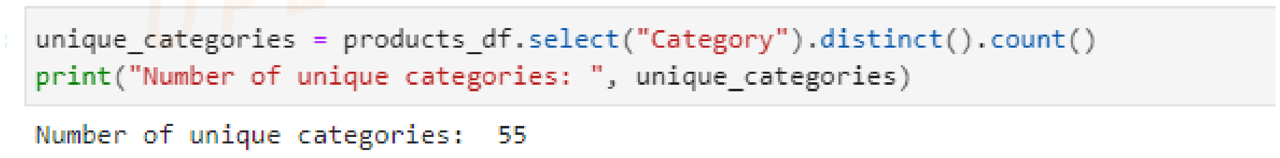


```
total_products = products_df.count()
print(total_products)

1345
```

2.2

unique_categories = products_df.select("Category").distinct().count()

print("Number of unique categories: ", unique_categories)

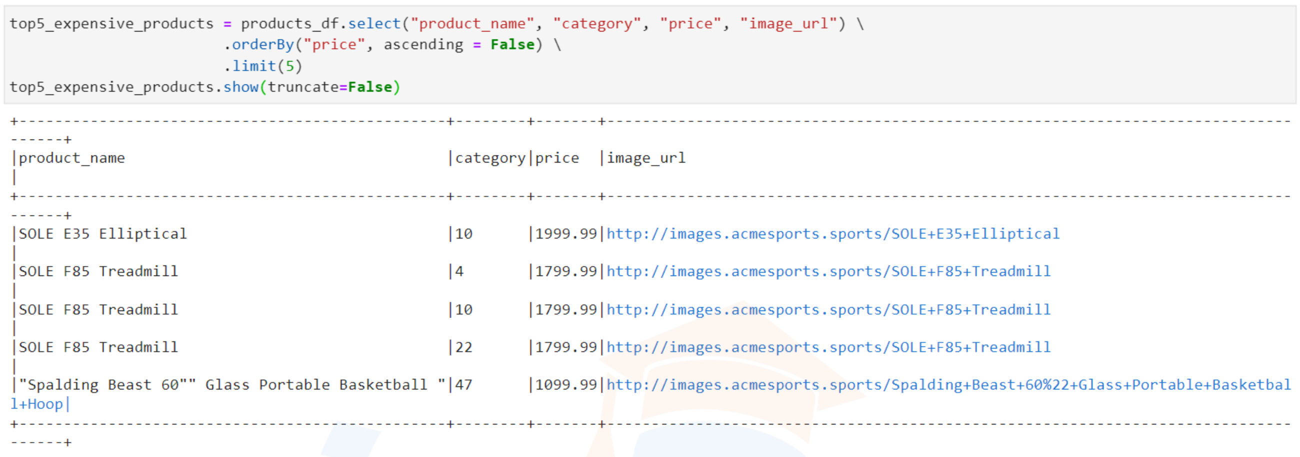


```
unique_categories = products_df.select("Category").distinct().count()
print("Number of unique categories: ", unique_categories)

Number of unique categories:  55
```

2.3

top5_expensive_products = products_df.select("product_name", "category", "price", "image_url") \

```
                    .orderBy("price", ascending = False) \
                    .limit(5)
top5_expensive_products.show(truncate=False)
```



```
top5_expensive_products = products_df.select("product_name", "category", "price", "image_url") \
                          .orderBy("price", ascending = False) \
                          .limit(5)
top5_expensive_products.show(truncate=False)
```

```
+------------------------------------------------+--------+-------+-----------------------------------------------------------------------------
------+
|product_name                                    |category|price  |image_url
|
+------------------------------------------------+--------+-------+-----------------------------------------------------------------------------
------+
|SOLE E35 Elliptical                             |10      |1999.99|http://images.acmesports.sports/SOLE+E35+Elliptical
|
|SOLE F85 Treadmill                              |4       |1799.99|http://images.acmesports.sports/SOLE+F85+Treadmill
|
|SOLE F85 Treadmill                              |10      |1799.99|http://images.acmesports.sports/SOLE+F85+Treadmill
|
|SOLE F85 Treadmill                              |22      |1799.99|http://images.acmesports.sports/SOLE+F85+Treadmill
|
|"Spalding Beast 60"" Glass Portable Basketball "|47      |1099.99|http://images.acmesports.sports/Spalding+Beast+60%22+Glass+Portable+Basketbal
l+Hoop|
+------------------------------------------------+--------+-------+-----------------------------------------------------------------------------
------+
```

2.4

```
products_above_100 = products_df.filter("price > 100") \
                     .groupBy("Category") \
                     .count() \
                     .withColumnRenamed("count", "Number_of_products")
products_above_100.show(truncate=False)
```

```
products_above_100 = products_df.filter("price > 100") \
                        .groupBy("Category") \
                        .count() \
                        .withColumnRenamed("count", "Number_of_products")
products_above_100.show(truncate=False)
```

```
+--------+------------------+
|Category|Number_of_products|
+--------+------------------+
|31      |17                |
|53      |16                |
|34      |15                |
|44      |9                 |
|12      |3                 |
|22      |4                 |
|47      |10                |
|52      |5                 |
|13      |1                 |
|6       |5                 |
|16      |11                |
|3       |5                 |
|20      |7                 |
|57      |6                 |
|54      |6                 |
|48      |17                |
|5       |11                |
|19      |13                |
|41      |11                |
|43      |23                |
+--------+------------------+
only showing top 20 rows
```

2.5

expensive_cat_products = products_df.filter("price > 200 and category = 5")

result = expensive_cat_products.select("product_name", "price")

result.show(truncate=False)

```
expensive_cat_products = products_df.filter("price > 200 and category = 5")
result = expensive_cat_products.select("product_name", "price")
result.show(truncate=False)
```

```
+------------------------------------------------+------+
|product_name                                    |price |
+------------------------------------------------+------+
|"Goaliath 54"" In-Ground Basketball Hoop with P"|499.99|
|Fitness Gear 300 lb Olympic Weight Set          |209.99|
|Teeter Hang Ups NXT-S Inversion Table           |299.99|
+------------------------------------------------+------+
```

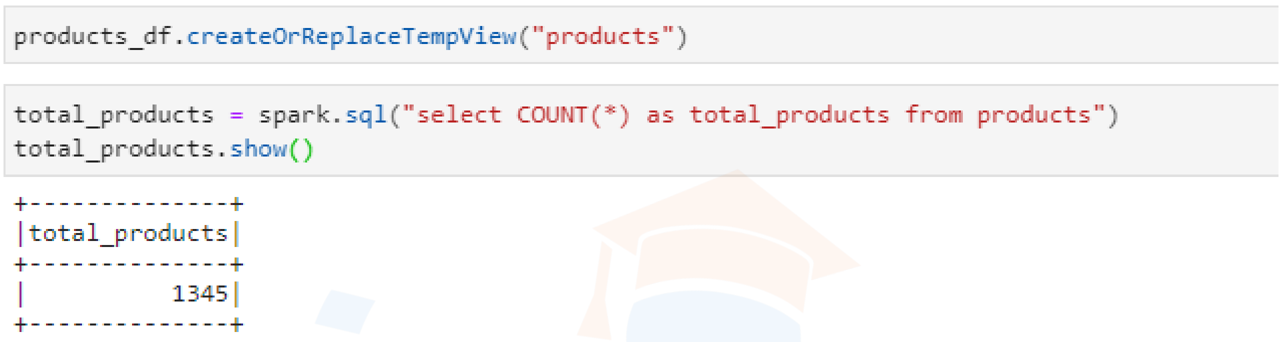
#Performing the tasks using Spark SQL

2.1

products_df.createOrReplaceTempView("products")

total_products = spark.sql("select COUNT(*) as total_products from products")

total_products.show()



```
products_df.createOrReplaceTempView("products")
```

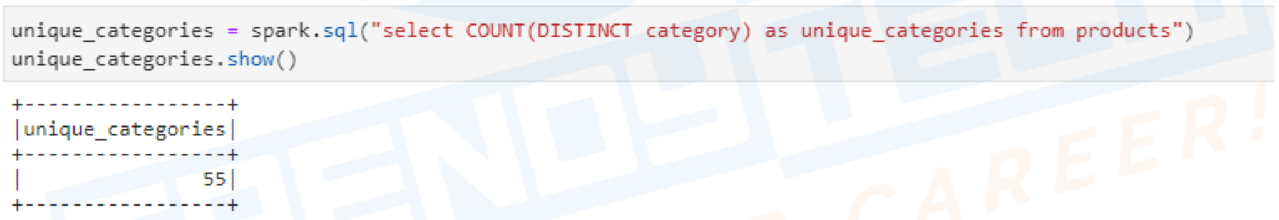
```
total_products = spark.sql("select COUNT(*) as total_products from products")
total_products.show()
```

```
+--------------+
|total_products|
+--------------+
|          1345|
+--------------+
```

2.2

unique_categories = spark.sql("select COUNT(DISTINCT category) as unique_categories from products")

unique_categories.show()



```
unique_categories = spark.sql("select COUNT(DISTINCT category) as unique_categories from products")
unique_categories.show()
```

```
+-----------------+
|unique_categories|
+-----------------+
|               55|
+-----------------+
```

2.3

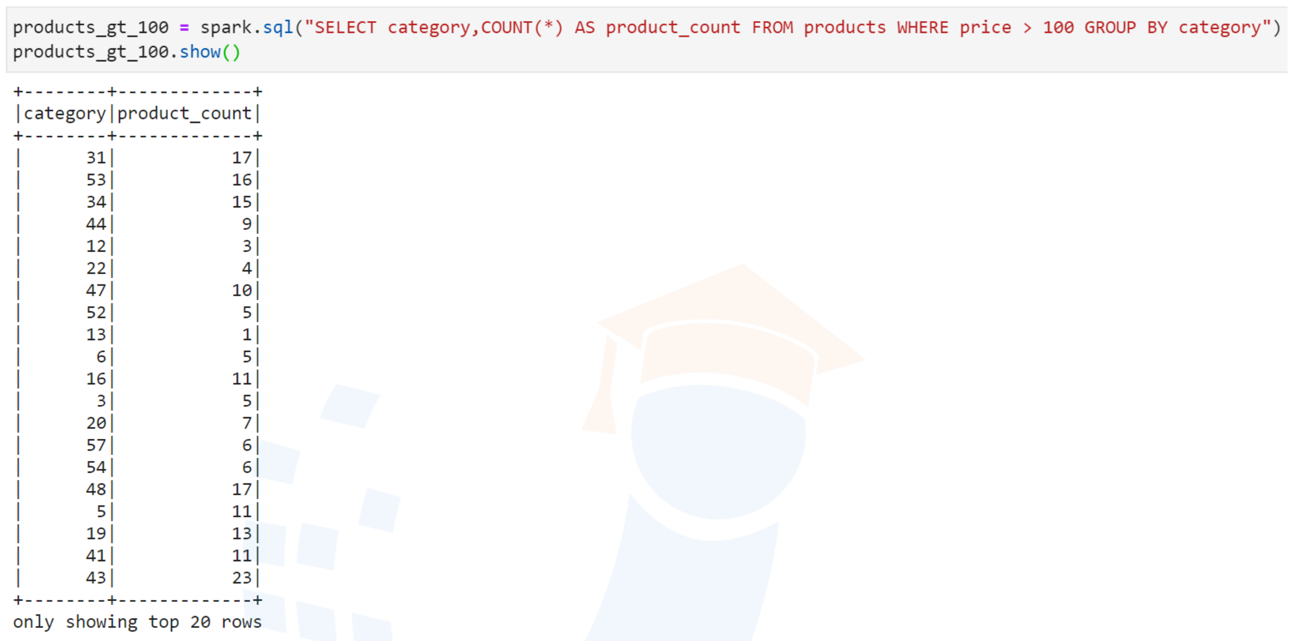
top_5_expensive_products = spark.sql("select product_name,category,price,image_url from products ORDER BY Price DESC LIMIT 5")

top_5_expensive_products.show(truncate=False)

2.4

products_gt_100 = spark.sql("SELECT category,COUNT(*) AS product_count FROM products WHERE price > 100 GROUP BY category")

products_gt_100.show()



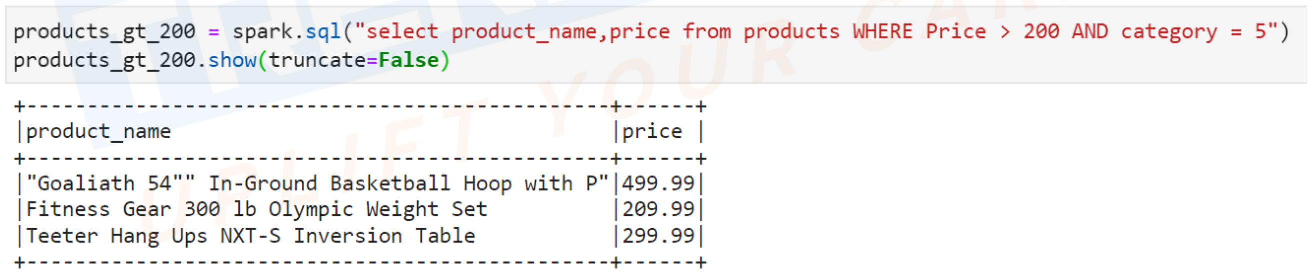
```
products_gt_100 = spark.sql("SELECT category,COUNT(*) AS product_count FROM products WHERE price > 100 GROUP BY category")
products_gt_100.show()
```

```
+--------+-------------+
|category|product_count|
+--------+-------------+
|      31|           17|
|      53|           16|
|      34|           15|
|      44|            9|
|      12|            3|
|      22|            4|
|      47|           10|
|      52|            5|
|      13|            1|
|       6|            5|
|      16|           11|
|       3|            5|
|      20|            7|
|      57|            6|
|      54|            6|
|      48|           17|
|       5|           11|
|      19|           13|
|      41|           11|
|      43|           23|
+--------+-------------+
only showing top 20 rows
```

2.5

products_gt_200 = spark.sql("select product_name,price from products WHERE Price > 200 AND category = 5")

products_gt_200.show(truncate=False)



```
products_gt_200 = spark.sql("select product_name,price from products WHERE Price > 200 AND category = 5")
products_gt_200.show(truncate=False)
```

```
+---------------------------------------------+------+
|product_name                                 |price |
+---------------------------------------------+------+
|"Goaliath 54"" In-Ground Basketball Hoop with P"|499.99|
|Fitness Gear 300 lb Olympic Weight Set       |209.99|
|Teeter Hang Ups NXT-S Inversion Table        |299.99|
+---------------------------------------------+------+
```

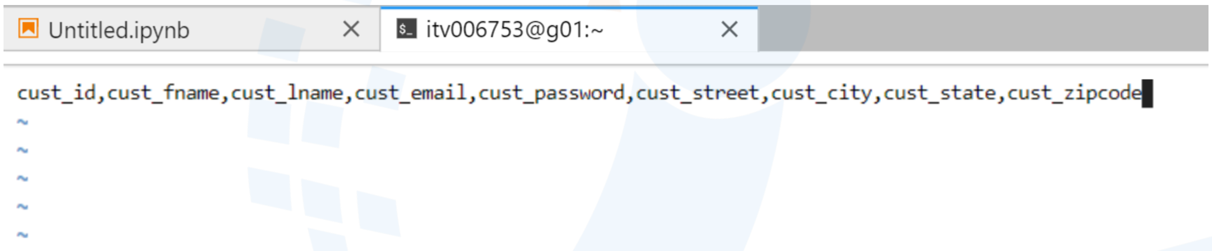
Question 3

#First we are adding headers to the dataset, as there is no header in the file given.(This method of adding headers to the dataset is just for the demo purpose. We will be learning the industry standard way of adding the headers in the upcoming weeks.)

vi customer_header_data

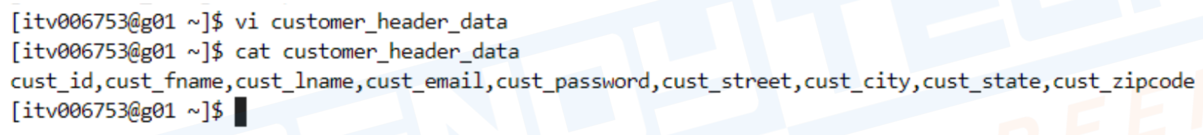
[Insert the following into the opened file -
cust_id,cust_fname,cust_lname,cust_email,cust_password,cust_street,cust_city,cust_state,cust_zipcode

]



```
cust_id,cust_fname,cust_lname,cust_email,cust_password,cust_street,cust_city,cust_state,cust_zipcode
~
~
~
~
~
```

cat customer_header_data



```
[itv006753@g01 ~]$ vi customer_header_data
[itv006753@g01 ~]$ cat customer_header_data
cust_id,cust_fname,cust_lname,cust_email,cust_password,cust_street,cust_city,cust_state,cust_zipcode
[itv006753@g01 ~]$
```

mkdir Assignment_customer

hadoop fs -get /public/trendytech/retail_db/customers/part-00000 Assignment_customer

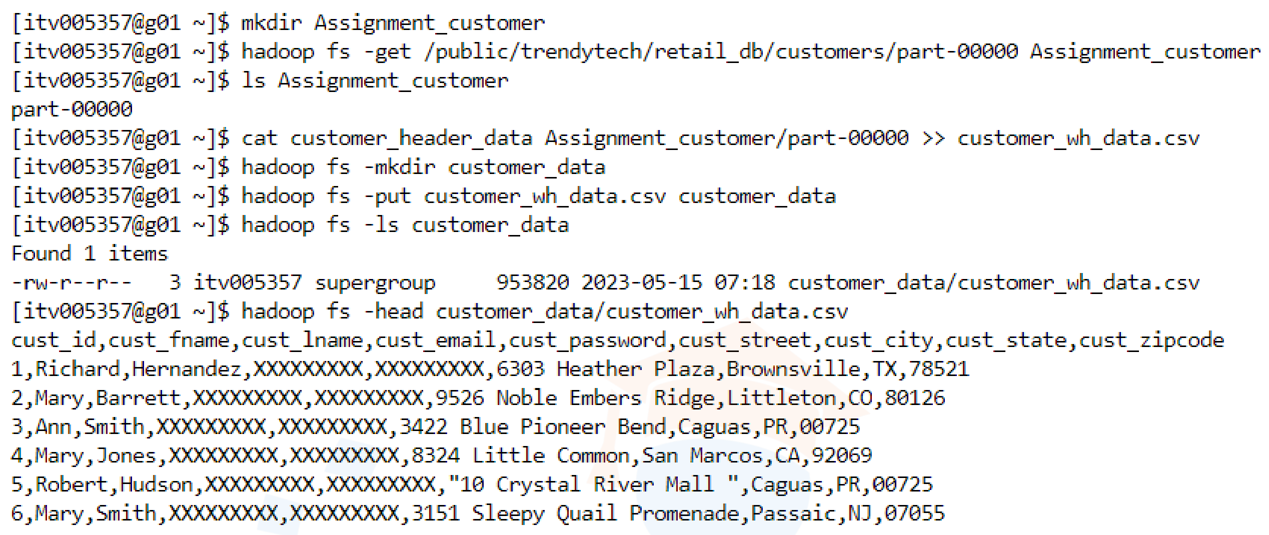
cat customer_header_data Assignment_customer/part-00000 >> customer_wh_data.csv

hadoop fs -mkdir customer_data

hadoop fs -put customer_wh_data.csv customer_data

hadoop fs -ls customer_data

hadoop fs -head customer_data/customer_wh_data.csv



```
[itv005357@g01 ~]$ mkdir Assignment_customer
[itv005357@g01 ~]$ hadoop fs -get /public/trendytech/retail_db/customers/part-00000 Assignment_customer
[itv005357@g01 ~]$ ls Assignment_customer
part-00000
[itv005357@g01 ~]$ cat customer_header_data Assignment_customer/part-00000 >> customer_wh_data.csv
[itv005357@g01 ~]$ hadoop fs -mkdir customer_data
[itv005357@g01 ~]$ hadoop fs -put customer_wh_data.csv customer_data
[itv005357@g01 ~]$ hadoop fs -ls customer_data
Found 1 items
-rw-r--r--   3 itv005357 supergroup     953820 2023-05-15 07:18 customer_data/customer_wh_data.csv
[itv005357@g01 ~]$ hadoop fs -head customer_data/customer_wh_data.csv
cust_id,cust_fname,cust_lname,cust_email,cust_password,cust_street,cust_city,cust_state,cust_zipcode
1,Richard,Hernandez,XXXXXXXXX,XXXXXXXXX,6303 Heather Plaza,Brownsville,TX,78521
2,Mary,Barrett,XXXXXXXXX,XXXXXXXXX,9526 Noble Embers Ridge,Littleton,CO,80126
3,Ann,Smith,XXXXXXXXX,XXXXXXXXX,3422 Blue Pioneer Bend,Caguas,PR,00725
4,Mary,Jones,XXXXXXXXX,XXXXXXXXX,8324 Little Common,San Marcos,CA,92069
5,Robert,Hudson,XXXXXXXXX,XXXXXXXXX,"10 Crystal River Mall ",Caguas,PR,00725
6,Mary,Smith,XXXXXXXXX,XXXXXXXXX,3151 Sleepy Quail Promenade,Passaic,NJ,07055
```

So, the path of the data in HDFS is :
/user/<username>/customer_data/customer_wh_data.csv

Boiler plate code:

from pyspark.sql import SparkSession

from pyspark.sql import SparkSession

import getpass

username = getpass.getuser()

spark= SparkSession. \

builder. \

config('spark.ui.port','0'). \

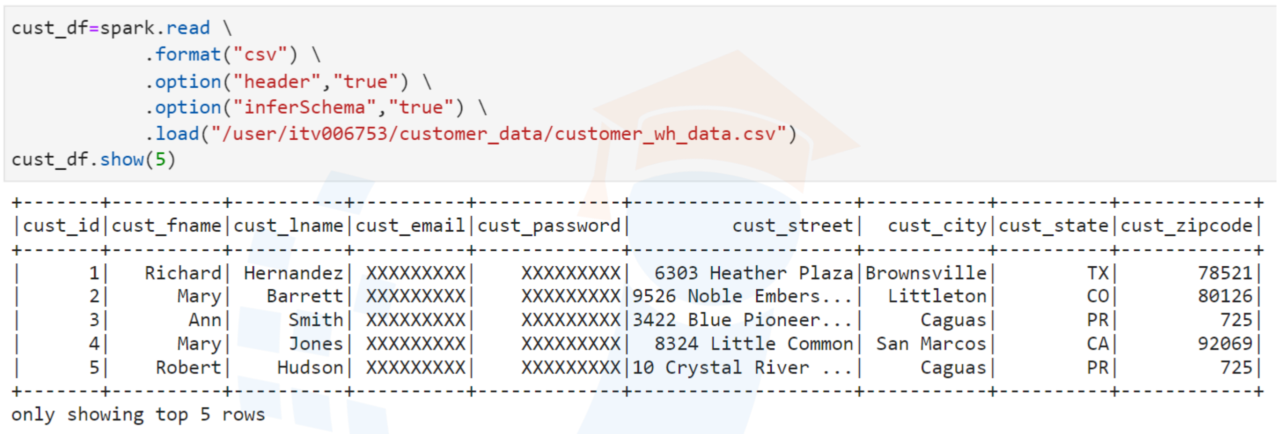
config("spark.sql.warehouse.dir", f"/user/{username}/warehouse"). \

enableHiveSupport(). \

master('yarn'). \

getOrCreate()

```
#Performing the tasks using Dataframe
cust_df=spark.read \
        .format("csv") \
        .option("header","true") \
        .option("inferSchema","true") \
        .load("/user/itv005357/customer_data/customer_wh_data.csv")
cust_df.show(5)
```



```
cust_df=spark.read \
            .format("csv") \
            .option("header","true") \
            .option("inferSchema","true") \
            .load("/user/itv006753/customer_data/customer_wh_data.csv")
cust_df.show(5)
```

```
+-------+----------+----------+----------+-------------+--------------------+-----------+----------+------------+
|cust_id|cust_fname|cust_lname|cust_email|cust_password|         cust_street|  cust_city|cust_state|cust_zipcode|
+-------+----------+----------+----------+-------------+--------------------+-----------+----------+------------+
|      1|   Richard| Hernandez| XXXXXXXXX|    XXXXXXXXX|  6303 Heather Plaza|Brownsville|        TX|       78521|
|      2|      Mary|   Barrett| XXXXXXXXX|    XXXXXXXXX|9526 Noble Embers...|  Littleton|        CO|       80126|
|      3|       Ann|     Smith| XXXXXXXXX|    XXXXXXXXX|3422 Blue Pioneer...|     Caguas|        PR|         725|
|      4|      Mary|     Jones| XXXXXXXXX|    XXXXXXXXX|  8324 Little Common| San Marcos|        CA|       92069|
|      5|    Robert|    Hudson| XXXXXXXXX|    XXXXXXXXX|10 Crystal River ...|     Caguas|        PR|         725|
+-------+----------+----------+----------+-------------+--------------------+-----------+----------+------------+
only showing top 5 rows
```

3.1

```
state_counts = cust_df.groupBy("cust_state").count().orderBy("cust_state")
state_counts.show()
```

```
state_counts = cust_df.groupBy("cust_state").count().orderBy("cust_state")
state_counts.show()
```

```
+----------+-----+
|cust_state|count|
+----------+-----+
|        AL|    3|
|        AR|   12|
|        AZ|  213|
|        CA| 2012|
|        CO|  122|
|        CT|   73|
|        DC|   42|
|        DE|   23|
|        FL|  374|
|        GA|  169|
|        HI|   87|
|        IA|    5|
|        ID|    9|
|        IL|  523|
|        IN|   40|
|        KS|   29|
|        KY|   35|
|        LA|   63|
|        MA|  113|
|        MD|  164|
+----------+-----+
only showing top 20 rows
```

3.2

last_name_counts = cust_df.groupBy("cust_lname").count().orderBy("count",ascending=False).limit (5)

last_name_counts.show()

```
last_name_counts = cust_df.groupBy("cust_lname").count().orderBy("count",ascending=False).limit(5)
last_name_counts.show()
```

```
+----------+-----+
|cust_lname|count|
+----------+-----+
|     Smith| 4626|
|   Johnson|   76|
|  Williams|   69|
|     Jones|   65|
|     Brown|   62|
+----------+-----+
```

3.3

from pyspark.sql.functions import length

invalid_zips = cust_df.filter(length("cust_zipcode") != 5)

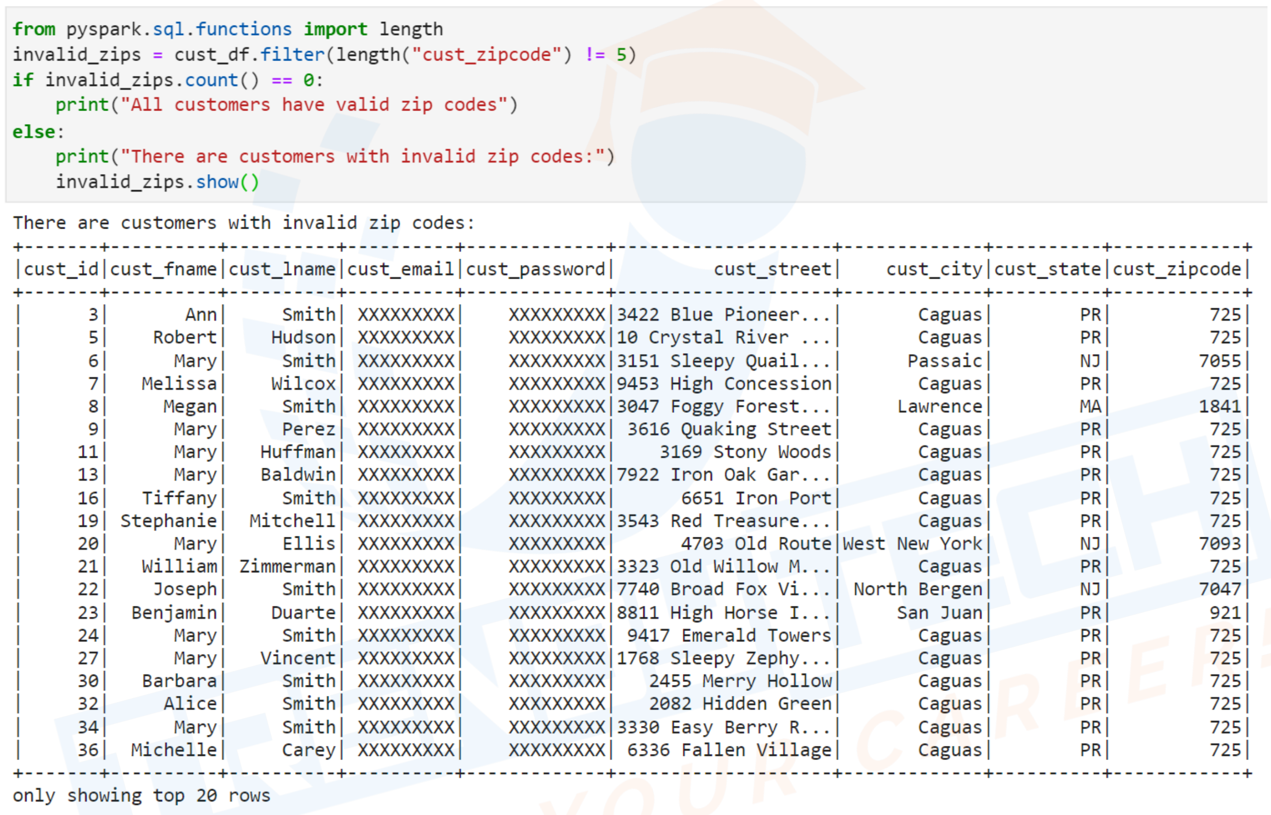
if invalid_zips.count() == 0:

    print("All customers have valid zip codes")

else:

    print("There are customers with invalid zip codes:")

    invalid_zips.show()



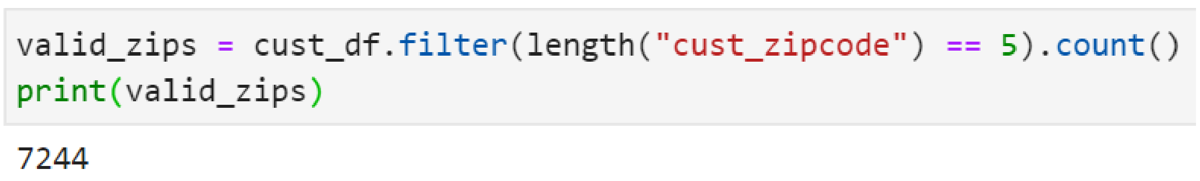
```
from pyspark.sql.functions import length
invalid_zips = cust_df.filter(length("cust_zipcode") != 5)
if invalid_zips.count() == 0:
    print("All customers have valid zip codes")
else:
    print("There are customers with invalid zip codes:")
    invalid_zips.show()
```

```
There are customers with invalid zip codes:
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
|cust_id|cust_fname|cust_lname|cust_email|cust_password|         cust_street|    cust_city|cust_state|cust_zipcode|
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
|      3|       Ann|     Smith| XXXXXXXXX|    XXXXXXXXX|3422 Blue Pioneer...|       Caguas|        PR|         725|
|      5|    Robert|    Hudson| XXXXXXXXX|    XXXXXXXXX|10 Crystal River ...|       Caguas|        PR|         725|
|      6|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX|3151 Sleepy Quail...|      Passaic|        NJ|        7055|
|      7|   Melissa|    Wilcox| XXXXXXXXX|    XXXXXXXXX|9453 High Concession|       Caguas|        PR|         725|
|      8|     Megan|     Smith| XXXXXXXXX|    XXXXXXXXX|3047 Foggy Forest...|     Lawrence|        MA|        1841|
|      9|      Mary|     Perez| XXXXXXXXX|    XXXXXXXXX| 3616 Quaking Street|       Caguas|        PR|         725|
|     11|      Mary|   Huffman| XXXXXXXXX|    XXXXXXXXX|   3169 Stony Woods|       Caguas|        PR|         725|
|     13|      Mary|   Baldwin| XXXXXXXXX|    XXXXXXXXX|7922 Iron Oak Gar...|       Caguas|        PR|         725|
|     16|   Tiffany|     Smith| XXXXXXXXX|    XXXXXXXXX|     6651 Iron Port|       Caguas|        PR|         725|
|     19| Stephanie|  Mitchell| XXXXXXXXX|    XXXXXXXXX|3543 Red Treasure...|       Caguas|        PR|         725|
|     20|      Mary|     Ellis| XXXXXXXXX|    XXXXXXXXX|      4703 Old Route|West New York|        NJ|        7093|
|     21|   William| Zimmerman| XXXXXXXXX|    XXXXXXXXX|3323 Old Willow M...|       Caguas|        PR|         725|
|     22|    Joseph|     Smith| XXXXXXXXX|    XXXXXXXXX|7740 Broad Fox Vi...| North Bergen|        NJ|        7047|
|     23|  Benjamin|    Duarte| XXXXXXXXX|    XXXXXXXXX|8811 High Horse I...|     San Juan|        PR|         921|
|     24|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX| 9417 Emerald Towers|       Caguas|        PR|         725|
|     27|      Mary|   Vincent| XXXXXXXXX|    XXXXXXXXX|1768 Sleepy Zephy...|       Caguas|        PR|         725|
|     30|   Barbara|     Smith| XXXXXXXXX|    XXXXXXXXX|   2455 Merry Hollow|       Caguas|        PR|         725|
|     32|     Alice|     Smith| XXXXXXXXX|    XXXXXXXXX|   2082 Hidden Green|       Caguas|        PR|         725|
|     34|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX|3330 Easy Berry R...|       Caguas|        PR|         725|
|     36|  Michelle|     Carey| XXXXXXXXX|    XXXXXXXXX| 6336 Fallen Village|       Caguas|        PR|         725|
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
only showing top 20 rows
```

3.4

valid_zips = cust_df.filter(length("cust_zipcode") == 5).count()

print(valid_zips)



```
valid_zips = cust_df.filter(length("cust_zipcode") == 5).count()
print(valid_zips)
```

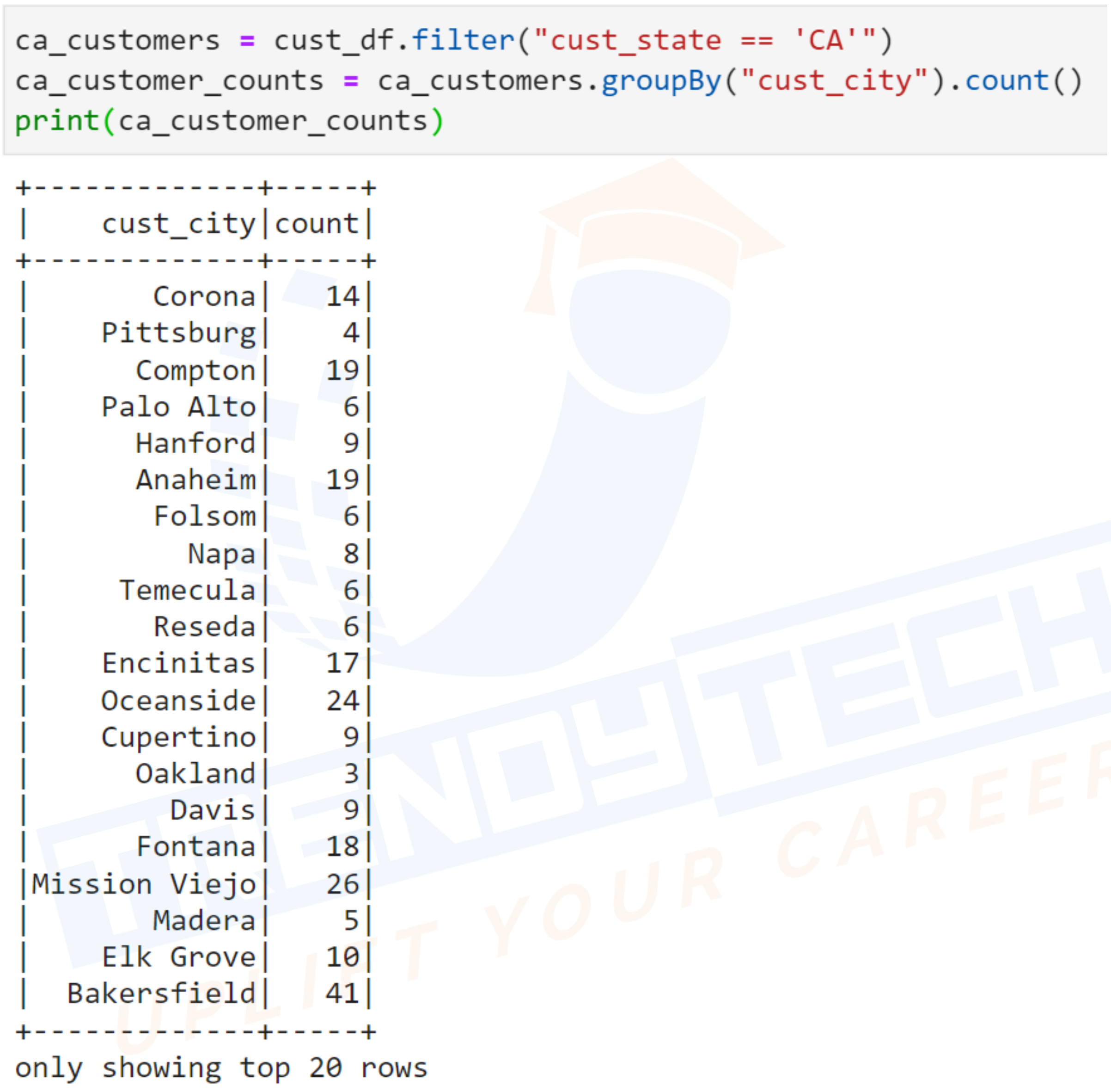
```
7244
```

3.5

ca_customers = cust_df.filter("cust_state == 'CA'")

ca_customer_counts = ca_customers.groupBy("cust_city").count()

print(ca_customer_counts)



```
ca_customers = cust_df.filter("cust_state == 'CA'")
ca_customer_counts = ca_customers.groupBy("cust_city").count()
print(ca_customer_counts)
```

```
+-------------+-----+
|    cust_city|count|
+-------------+-----+
|       Corona|   14|
|    Pittsburg|    4|
|      Compton|   19|
|    Palo Alto|    6|
|      Hanford|    9|
|      Anaheim|   19|
|       Folsom|    6|
|         Napa|    8|
|     Temecula|    6|
|       Reseda|    6|
|    Encinitas|   17|
|    Oceanside|   24|
|    Cupertino|    9|
|      Oakland|    3|
|        Davis|    9|
|      Fontana|   18|
|Mission Viejo|   26|
|       Madera|    5|
|    Elk Grove|   10|
|  Bakersfield|   41|
+-------------+-----+
only showing top 20 rows
```

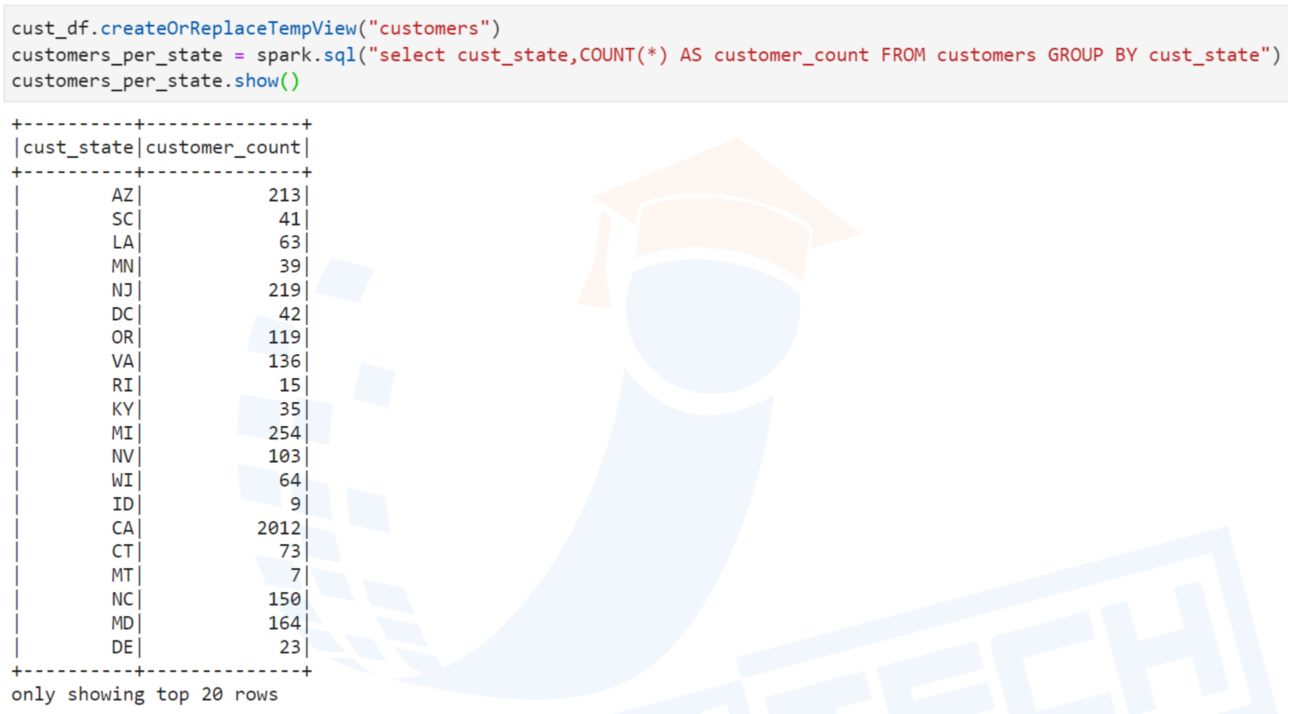
#Performing the tasks using Spark SQL

3.1

cust_df.createOrReplaceTempView("customers")

customers_per_state = spark.sql("select cust_state,COUNT(*) AS customer_count FROM customers GROUP BY cust_state")

customers_per_state.show()



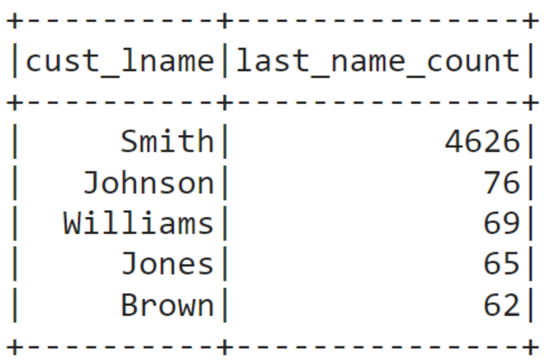
```
cust_df.createOrReplaceTempView("customers")
customers_per_state = spark.sql("select cust_state,COUNT(*) AS customer_count FROM customers GROUP BY cust_state")
customers_per_state.show()
```

```
+----------+--------------+
|cust_state|customer_count|
+----------+--------------+
|        AZ|           213|
|        SC|            41|
|        LA|            63|
|        MN|            39|
|        NJ|           219|
|        DC|            42|
|        OR|           119|
|        VA|           136|
|        RI|            15|
|        KY|            35|
|        MI|           254|
|        NV|           103|
|        WI|            64|
|        ID|             9|
|        CA|          2012|
|        CT|            73|
|        MT|             7|
|        NC|           150|
|        MD|           164|
|        DE|            23|
+----------+--------------+
only showing top 20 rows
```

3.2

top_5_last_names = spark.sql("select cust_lname,COUNT(*) AS last_name_count FROM customers GROUP BY cust_lname ORDER BY last_name_count DESC LIMIT 5")

top_5_last_names.show()

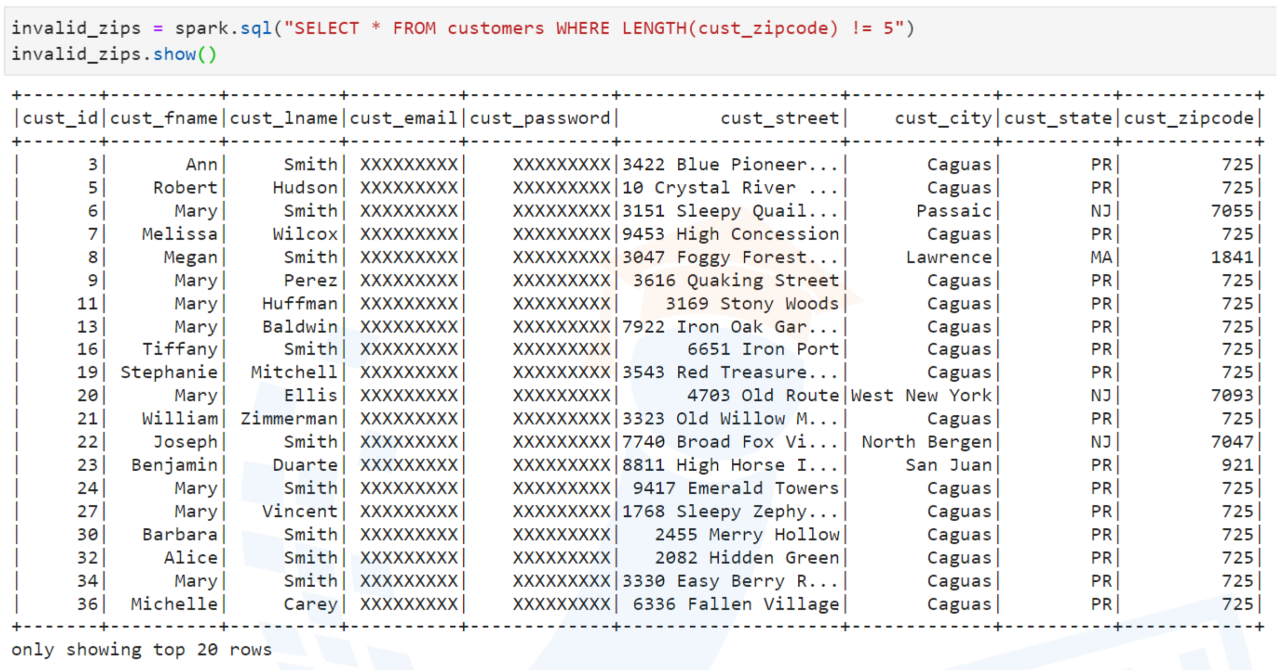


```
+----------+---------------+
|cust_lname|last_name_count|
+----------+---------------+
|     Smith|           4626|
|   Johnson|             76|
|  Williams|             69|
|     Jones|             65|
|     Brown|             62|
+----------+---------------+
```

3.3

invalid_zips = spark.sql("SELECT * FROM customers WHERE LENGTH(cust_zipcode) != 5")

invalid_zips.show()



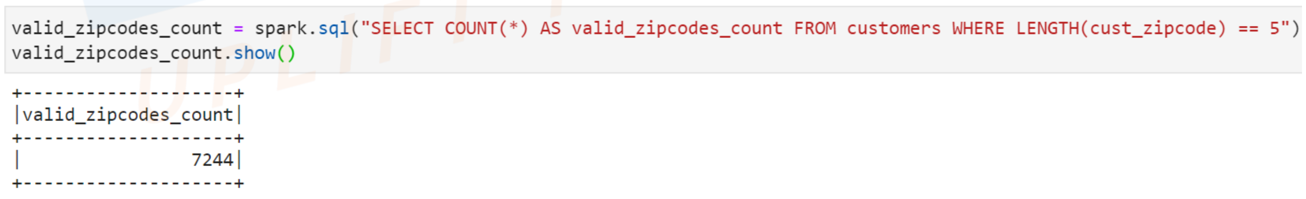
```
invalid_zips = spark.sql("SELECT * FROM customers WHERE LENGTH(cust_zipcode) != 5")
invalid_zips.show()
```

```
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
|cust_id|cust_fname|cust_lname|cust_email|cust_password|         cust_street|    cust_city|cust_state|cust_zipcode|
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
|      3|       Ann|     Smith| XXXXXXXXX|    XXXXXXXXX|3422 Blue Pioneer...|       Caguas|        PR|         725|
|      5|    Robert|    Hudson| XXXXXXXXX|    XXXXXXXXX|10 Crystal River ...|       Caguas|        PR|         725|
|      6|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX|3151 Sleepy Quail...|      Passaic|        NJ|        7055|
|      7|   Melissa|    Wilcox| XXXXXXXXX|    XXXXXXXXX|9453 High Concession|       Caguas|        PR|         725|
|      8|     Megan|     Smith| XXXXXXXXX|    XXXXXXXXX|3047 Foggy Forest...|     Lawrence|        MA|        1841|
|      9|      Mary|     Perez| XXXXXXXXX|    XXXXXXXXX| 3616 Quaking Street|       Caguas|        PR|         725|
|     11|      Mary|   Huffman| XXXXXXXXX|    XXXXXXXXX|    3169 Stony Woods|       Caguas|        PR|         725|
|     13|      Mary|   Baldwin| XXXXXXXXX|    XXXXXXXXX|7922 Iron Oak Gar...|       Caguas|        PR|         725|
|     16|   Tiffany|     Smith| XXXXXXXXX|    XXXXXXXXX|      6651 Iron Port|       Caguas|        PR|         725|
|     19| Stephanie|  Mitchell| XXXXXXXXX|    XXXXXXXXX|3543 Red Treasure...|       Caguas|        PR|         725|
|     20|      Mary|     Ellis| XXXXXXXXX|    XXXXXXXXX|      4703 Old Route|West New York|        NJ|        7093|
|     21|   William| Zimmerman| XXXXXXXXX|    XXXXXXXXX|3323 Old Willow M...|       Caguas|        PR|         725|
|     22|    Joseph|     Smith| XXXXXXXXX|    XXXXXXXXX|7740 Broad Fox Vi...| North Bergen|        NJ|        7047|
|     23|  Benjamin|    Duarte| XXXXXXXXX|    XXXXXXXXX|8811 High Horse I...|     San Juan|        PR|         921|
|     24|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX| 9417 Emerald Towers|       Caguas|        PR|         725|
|     27|      Mary|   Vincent| XXXXXXXXX|    XXXXXXXXX|1768 Sleepy Zephy...|       Caguas|        PR|         725|
|     30|   Barbara|     Smith| XXXXXXXXX|    XXXXXXXXX|   2455 Merry Hollow|       Caguas|        PR|         725|
|     32|     Alice|     Smith| XXXXXXXXX|    XXXXXXXXX|   2082 Hidden Green|       Caguas|        PR|         725|
|     34|      Mary|     Smith| XXXXXXXXX|    XXXXXXXXX|3330 Easy Berry R...|       Caguas|        PR|         725|
|     36|  Michelle|     Carey| XXXXXXXXX|    XXXXXXXXX| 6336 Fallen Village|       Caguas|        PR|         725|
+-------+----------+----------+----------+-------------+--------------------+-------------+----------+------------+
only showing top 20 rows
```

3.4

valid_zipcodes_count = spark.sql("SELECT COUNT(*) AS valid_zipcodes_count FROM customers WHERE LENGTH(cust_zipcode) == 5")

valid_zipcodes_count.show()



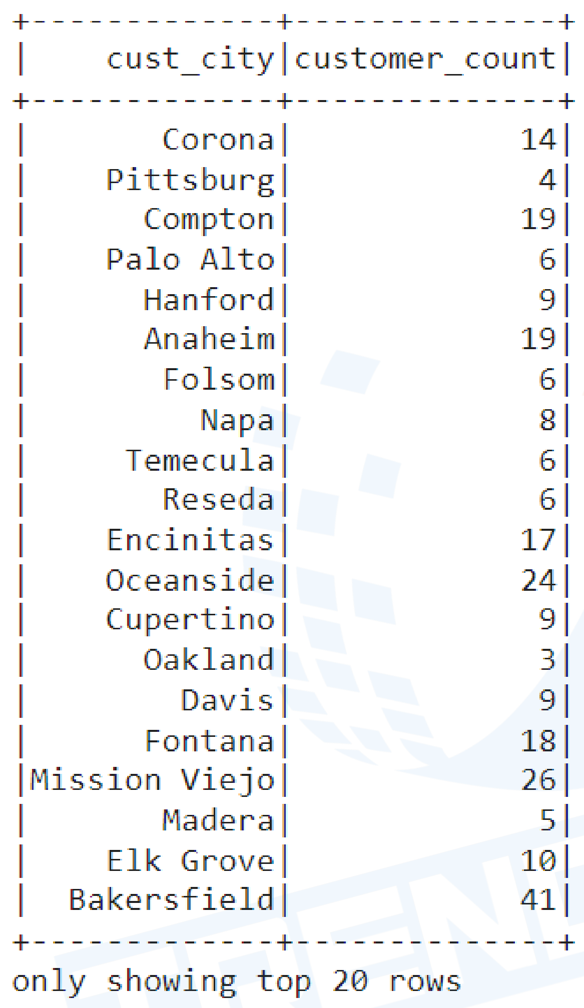
```
valid_zipcodes_count = spark.sql("SELECT COUNT(*) AS valid_zipcodes_count FROM customers WHERE LENGTH(cust_zipcode) == 5")
valid_zipcodes_count.show()
```

```
+--------------------+
|valid_zipcodes_count|
+--------------------+
|                7244|
+--------------------+
```

3.5

california_customers_per_city = spark.sql("select cust_city, COUNT(*) AS customer_count FROM customers WHERE cust_state = 'CA' GROUP BY cust_city")

california_customers_per_city.show()



```
+-------------+--------------+
|    cust_city|customer_count|
+-------------+--------------+
|       Corona|            14|
|    Pittsburg|             4|
|      Compton|            19|
|    Palo Alto|             6|
|      Hanford|             9|
|      Anaheim|            19|
|       Folsom|             6|
|         Napa|             8|
|     Temecula|             6|
|       Reseda|             6|
|    Encinitas|            17|
|    Oceanside|            24|
|    Cupertino|             9|
|      Oakland|             3|
|        Davis|             9|
|      Fontana|            18|
|Mission Viejo|            26|
|       Madera|             5|
|    Elk Grove|            10|
|  Bakersfield|            41|
+-------------+--------------+
only showing top 20 rows
```