KitKat Project 01

Consumer Expenditure Behaviour Analysis

Prepared by: Mihir Dhakan / mihirdhakan93.github.io

Firstly, If you have not seen the Introductory Blog on KitKat Series, please head to https://mihirdhakan.medium.com/introducing-kitkat-series-a-hub-to-practice-big-data-projects-aa782dbbdfb1

Project Name : Consumer Expenditure Behaviour Analysis

Difficulty Level: Beginner👶

Components used: MySQL, Sqoop, HDFS, Hive, HQL

Data Domain: Government (New Zealand)

EDI (Early Data Inventory): Data available to us is from 2007 Jan, till 2021 May containing below information in CSV Format.

Series_reference : A 13 digit reference number based on Category of expense
Period : Year and Month (YYYY.MM) on which the transaction(s) took place.
Data_value : Transacted Amount in Dollars
Suppressed: Y/N Flag field, not of much importance
STATUS : possible values are R,F,C. not of much importance
UNITS : Currency measurement unit
Magnitude: not of much importance
Subject: Static value as "Electronic Transaction…"
Group: Static value as "Private Values…"
Series_title_1 : possible values are Adjusted,actual.
Series_title_2 : Type of Expenses such as accomodation, supermarket, etc.

Assumptions: In this Project, we have made below assumptions to simulate the data as per industry standard.

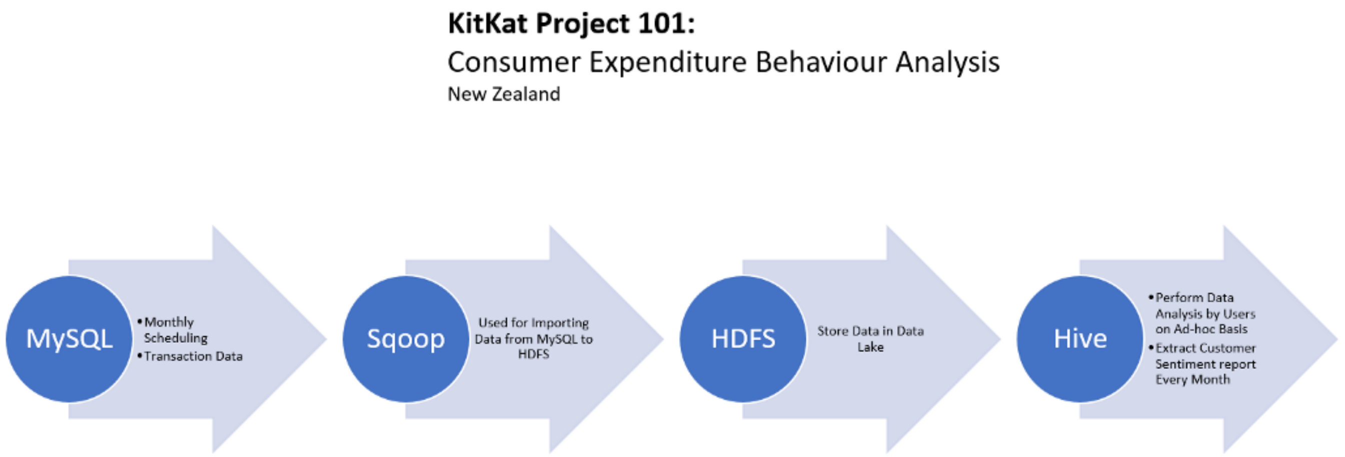
- Data is residing in MySQL Database.

Business requirement: The requirement is to bring the data from MySQL to Hadoop Data Lake and pump it every month and perform the analysis of "Consumer Expenditure Behavior". This would help to conclude the cost of living factor in New Zealand.

KPI's :

1. Top Expense category of each year till 2021. This is to know if the expenditure pattern is changing with time, and the growth of expense increasing (%) till 2021.
2. Average expense for each category spend in each year.
3. There is no end, to this. So we will limit to 2 KPI's 😉

Data Flow Diagram:



KitKat Project 101:

Consumer Expenditure Behaviour Analysis

New Zealand

MySQL
- Monthly Scheduling
- Transaction Data

Sqoop

Used for Importing Data from MySQL to HDFS

HDFS

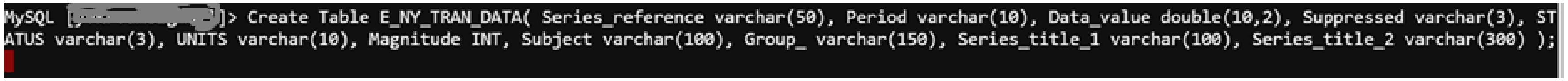
Store Data in Data Lake

Hive
- Perform Data Analysis by Users on Ad-hoc Basis
- Extract Customer Sentiment report Every Month

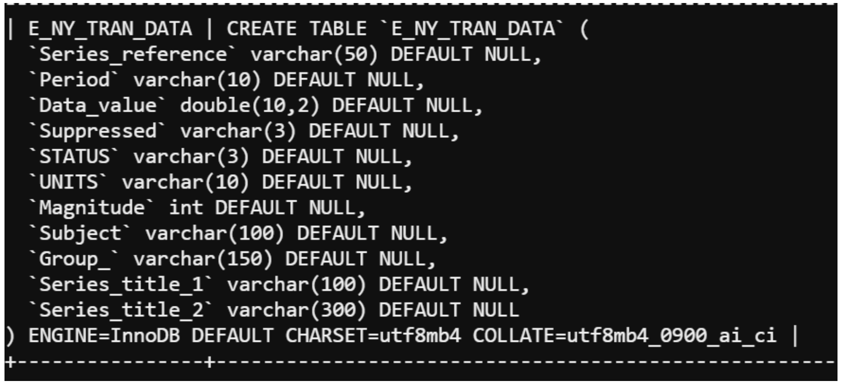
Let's Get our hands dirty then. 💻

Note: Text Version of all the source codes is available in Jupyter notebook File in Github in my repo.

1) Create Table in MySQL based on the data definition available to us. (Checkout the Datasets folder in GitHub to download the raw data)

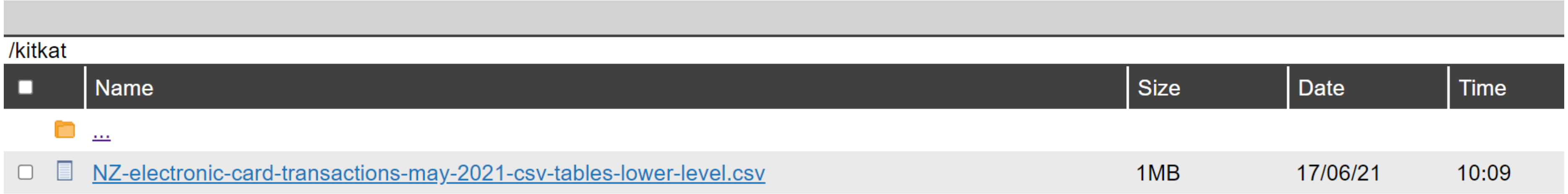


```
MySQL [[redacted]]> Create Table E_NY_TRAN_DATA( Series_reference varchar(50), Period varchar(10), Data_value double(10,2), Suppressed varchar(3), ST
ATUS varchar(3), UNITS varchar(10), Magnitude INT, Subject varchar(100), Group_ varchar(150), Series_title_1 varchar(100), Series_title_2 varchar(300) );
```



```
| E_NY_TRAN_DATA | CREATE TABLE `E_NY_TRAN_DATA` (
  `Series_reference` varchar(50) DEFAULT NULL,
  `Period` varchar(10) DEFAULT NULL,
  `Data_value` double(10,2) DEFAULT NULL,
  `Suppressed` varchar(3) DEFAULT NULL,
  `STATUS` varchar(3) DEFAULT NULL,
  `UNITS` varchar(10) DEFAULT NULL,
  `Magnitude` int DEFAULT NULL,
  `Subject` varchar(100) DEFAULT NULL,
  `Group_` varchar(150) DEFAULT NULL,
  `Series_title_1` varchar(100) DEFAULT NULL,
  `Series_title_2` varchar(300) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci |
+----------------+--------------------------------------------------
```

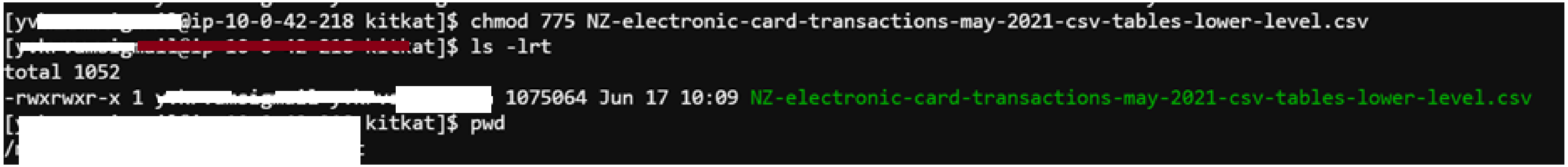
Upload the CSV File to FTP in Linux Machine



/kitkat

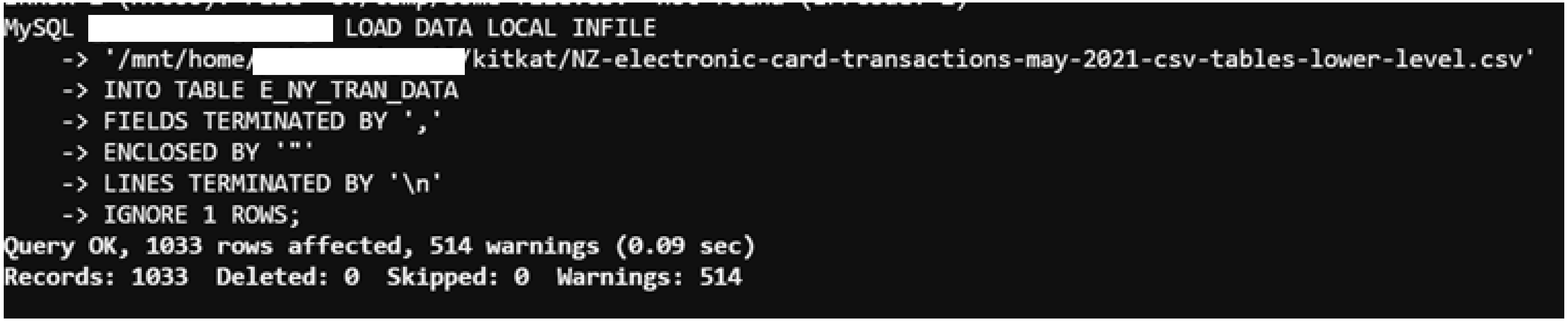
■	Name	Size	Date	Time
	...			
☐	NZ-electronic-card-transactions-may-2021-csv-tables-lower-level.csv	1MB	17/06/21	10:09

2) Access rights on file – ensure it has right access if not chmod it.



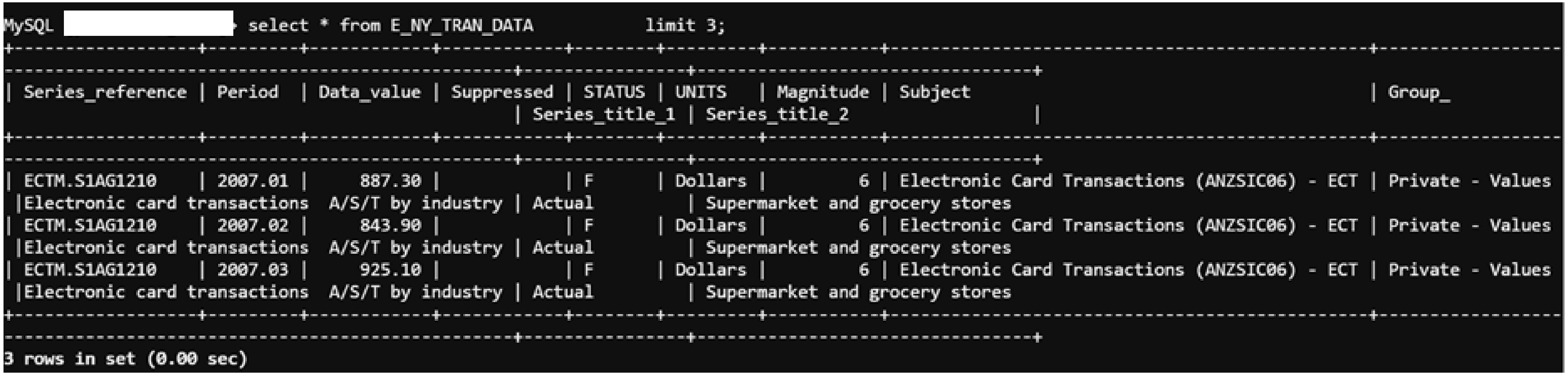
```
[[redacted]@ip-10-0-42-218 kitkat]$ chmod 775 NZ-electronic-card-transactions-may-2021-csv-tables-lower-level.csv
[[redacted]at]$ ls -lrt
total 1052
-rwxrwxr-x 1 [redacted] 1075064 Jun 17 10:09 NZ-electronic-card-transactions-may-2021-csv-tables-lower-level.csv
[[redacted] kitkat]$ pwd
/[redacted]
```

3) Load the file to Mysql



```
MySQL [redacted] LOAD DATA LOCAL INFILE
    -> '/mnt/home/[redacted]/kitkat/NZ-electronic-card-transactions-may-2021-csv-tables-lower-level.csv'
    -> INTO TABLE E_NY_TRAN_DATA
    -> FIELDS TERMINATED BY ','
    -> ENCLOSED BY '"'
    -> LINES TERMINATED BY '\n'
    -> IGNORE 1 ROWS;
Query OK, 1033 rows affected, 514 warnings (0.09 sec)
Records: 1033  Deleted: 0  Skipped: 0  Warnings: 514
```

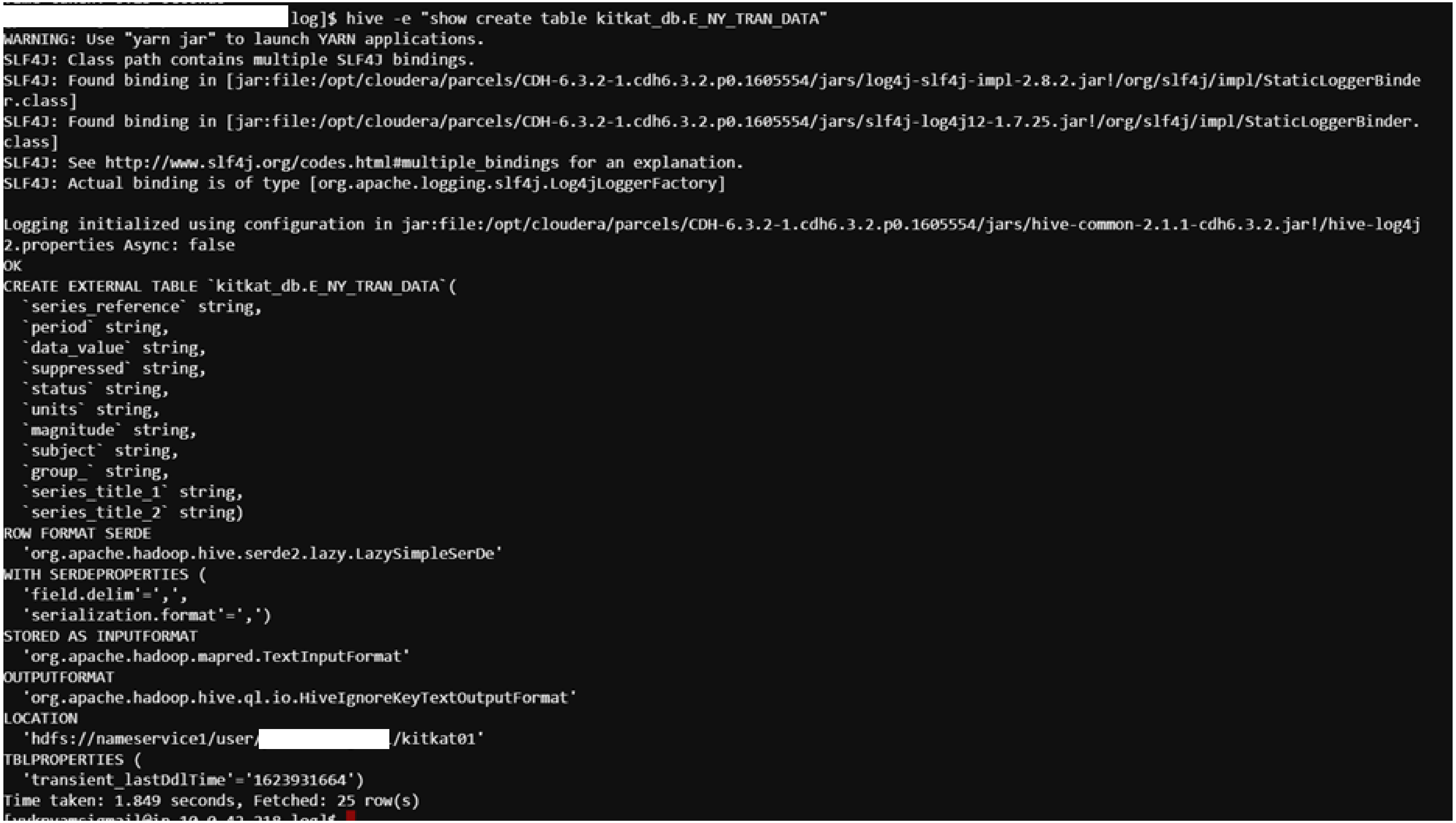
4) Check Loaded data



```
MySQL [redacted] select * from E_NY_TRAN_DATA          limit 3;
+-----------------+---------+-----------+------------+--------+---------+-----------+------------------------------------------------+------------------
------------------------------------------------+--------------+-----------------------------+
| Series_reference | Period  | Data_value | Suppressed | STATUS | UNITS   | Magnitude | Subject                                        | Group_
                                                | Series_title_1 | Series_title_2              |
+-----------------+---------+-----------+------------+--------+---------+-----------+------------------------------------------------+------------------
------------------------------------------------+--------------+-----------------------------+
| ECTM.S1AG1210    | 2007.01 |    887.30 |            | F      | Dollars |         6 | Electronic Card Transactions (ANZSIC06) - ECT | Private - Values
 |Electronic card transactions  A/S/T by industry | Actual        | Supermarket and grocery stores
| ECTM.S1AG1210    | 2007.02 |    843.90 |            | F      | Dollars |         6 | Electronic Card Transactions (ANZSIC06) - ECT | Private - Values
 |Electronic card transactions  A/S/T by industry | Actual        | Supermarket and grocery stores
| ECTM.S1AG1210    | 2007.03 |    925.10 |            | F      | Dollars |         6 | Electronic Card Transactions (ANZSIC06) - ECT | Private - Values
 |Electronic card transactions  A/S/T by industry | Actual        | Supermarket and grocery stores
+-----------------+---------+-----------+------------+--------+---------+-----------+------------------------------------------------+------------------
------------------------------------------------+--------------+-----------------------------+
3 rows in set (0.00 sec)
```

Hive:

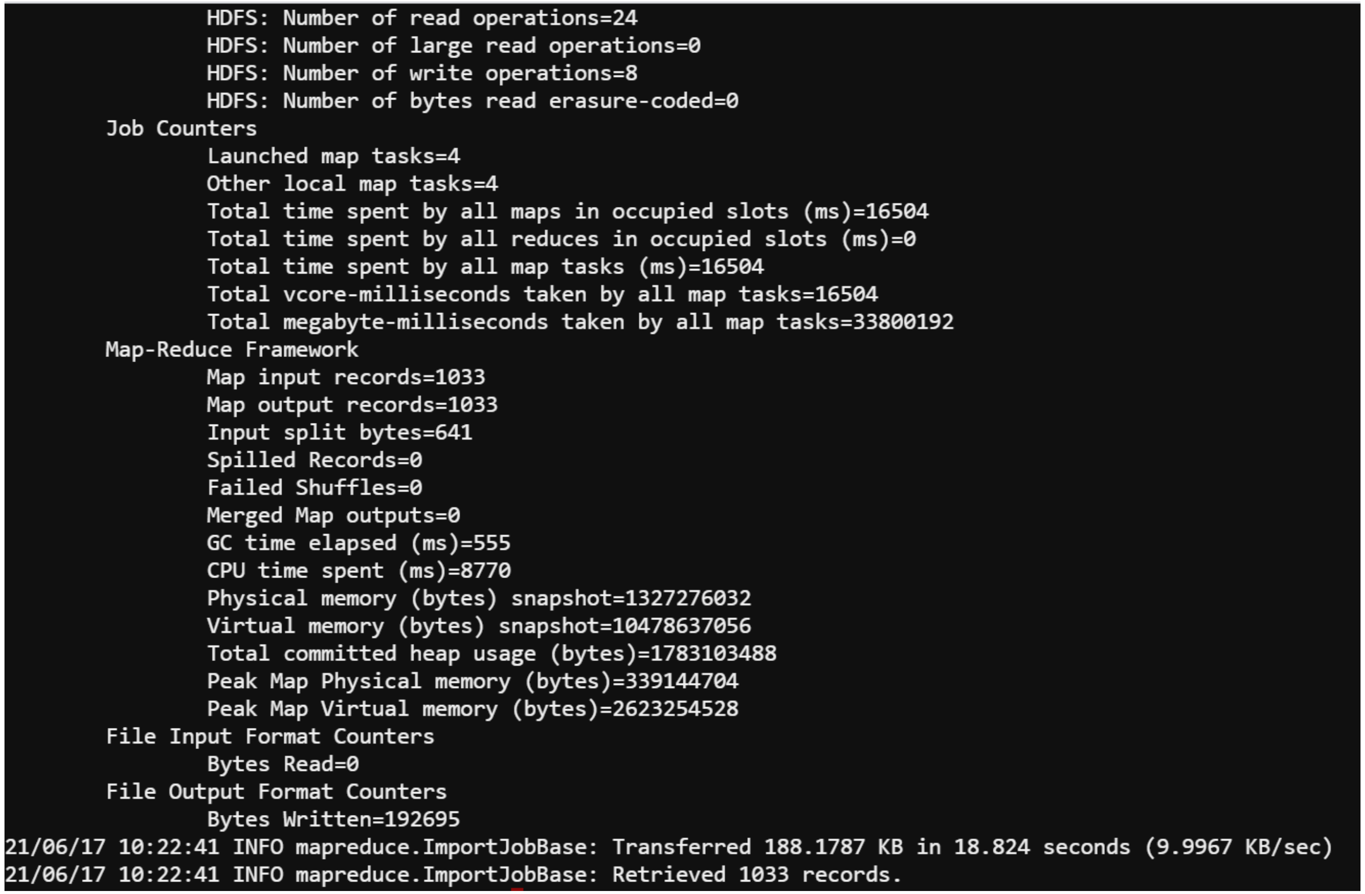
Create table in Hive



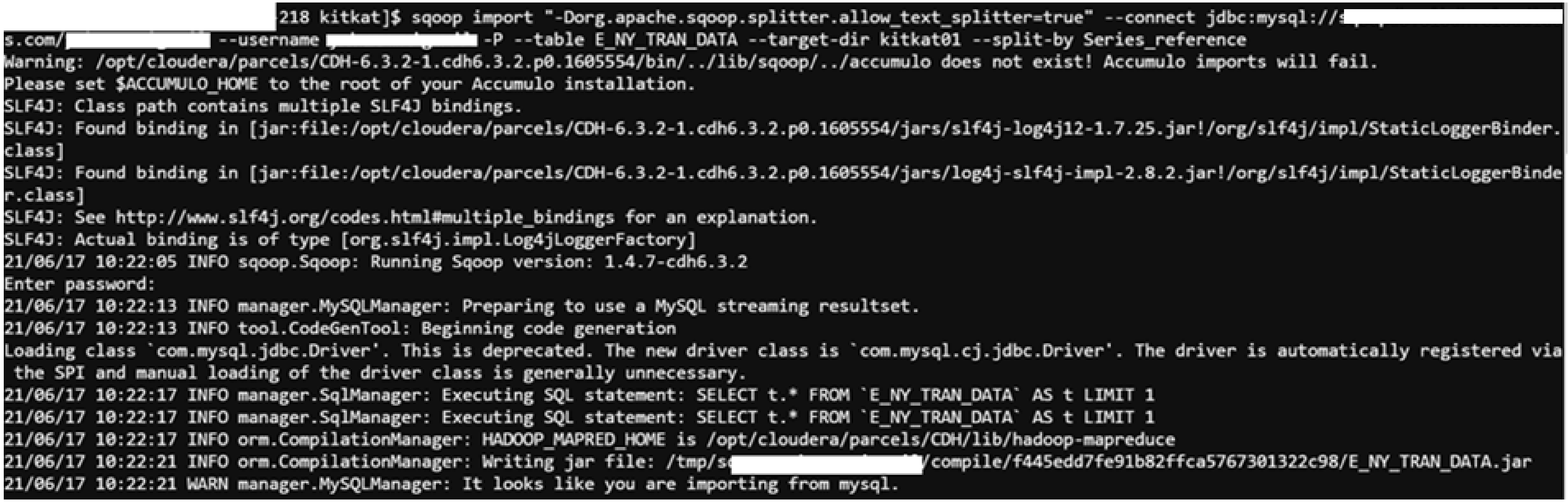
```
[redacted] log]$ hive -e "show create table kitkat_db.E_NY_TRAN_DATA"
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinde
r.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.
class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/hive-common-2.1.1-cdh6.3.2.jar!/hive-log4j
2.properties Async: false
OK
CREATE EXTERNAL TABLE `kitkat_db.E_NY_TRAN_DATA`(
  `series_reference` string,
  `period` string,
  `data_value` string,
  `suppressed` string,
  `status` string,
  `units` string,
  `magnitude` string,
  `subject` string,
  `group_` string,
  `series_title_1` string,
  `series_title_2` string)
ROW FORMAT SERDE
  'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe'
WITH SERDEPROPERTIES (
  'field.delim'=',',
  'serialization.format'=',')
STORED AS INPUTFORMAT
  'org.apache.hadoop.mapred.TextInputFormat'
OUTPUTFORMAT
  'org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat'
LOCATION
  'hdfs://nameservice1/user/[redacted]/kitkat01'
TBLPROPERTIES (
  'transient_lastDdlTime'='1623931664')
Time taken: 1.849 seconds, Fetched: 25 row(s)
```

SQOOP

1) Sqoop import the mysql data ; please note – in order for Sqoop to import the data, we need to ensure there is a PK column defined in MYSQL Table we are importing, however in this case – deliberately we have avoided having PK. So, to solve this will will ask Sqoop to use String Column for Splitting the data.. . the property for this is as below.

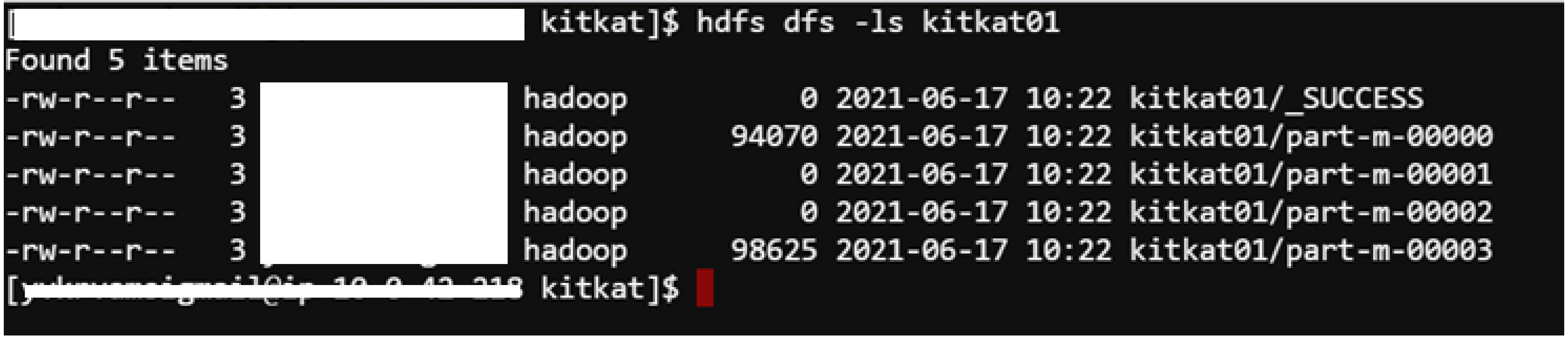


```
                HDFS: Number of read operations=24
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=8
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=4
                Other local map tasks=4
                Total time spent by all maps in occupied slots (ms)=16504
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=16504
                Total vcore-milliseconds taken by all map tasks=16504
                Total megabyte-milliseconds taken by all map tasks=33800192
        Map-Reduce Framework
                Map input records=1033
                Map output records=1033
                Input split bytes=641
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=555
                CPU time spent (ms)=8770
                Physical memory (bytes) snapshot=1327276032
                Virtual memory (bytes) snapshot=10478637056
                Total committed heap usage (bytes)=1783103488
                Peak Map Physical memory (bytes)=339144704
                Peak Map Virtual memory (bytes)=2623254528
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=192695
21/06/17 10:22:41 INFO mapreduce.ImportJobBase: Transferred 188.1787 KB in 18.824 seconds (9.9967 KB/sec)
21/06/17 10:22:41 INFO mapreduce.ImportJobBase: Retrieved 1033 records.
```

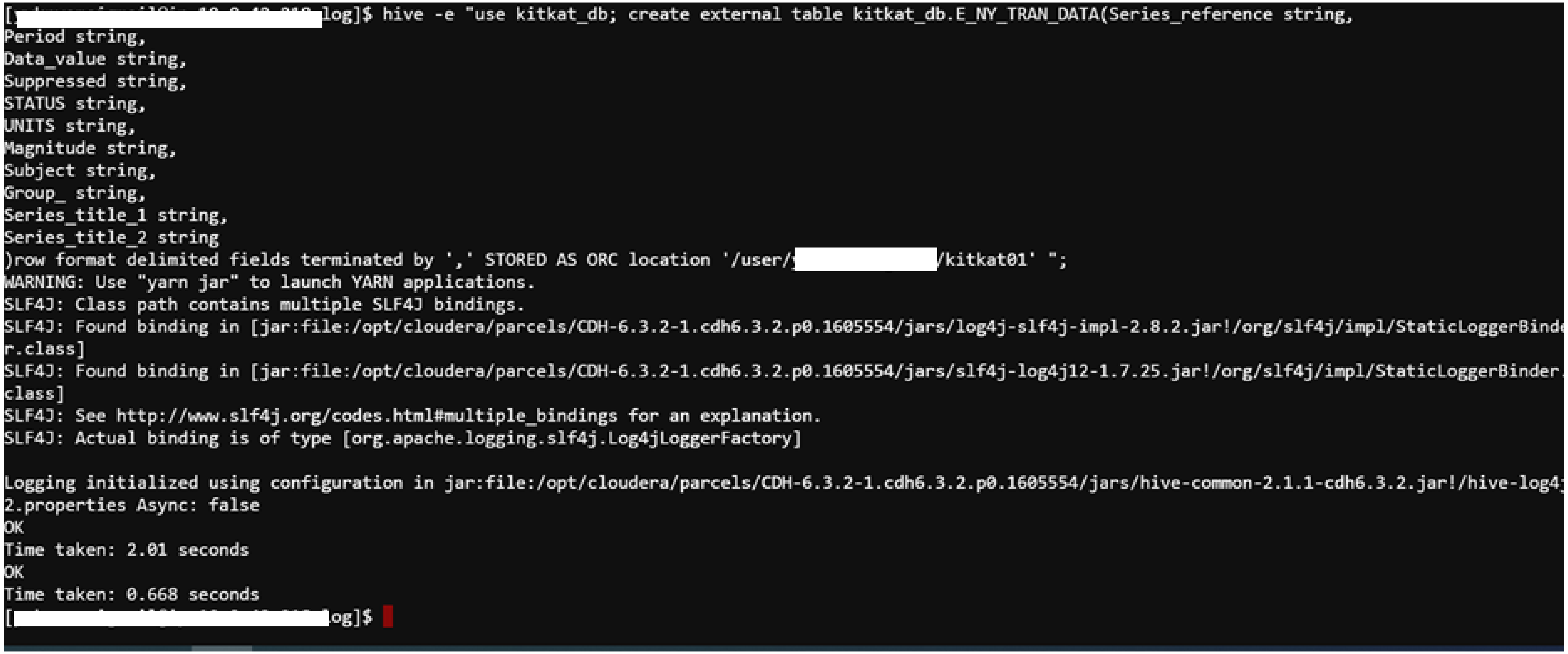


```
[redacted]218 kitkat]$ sqoop import "-Dorg.apache.sqoop.splitter.allow_text_splitter=true" --connect jdbc:mysql://[redacted]
s.com/[redacted] --username [redacted] -P --table E_NY_TRAN_DATA --target-dir kitkat01 --split-by Series_reference
Warning: /opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/bin/../lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.
class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinde
r.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.slf4j.impl.Log4jLoggerFactory]
21/06/17 10:22:05 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7-cdh6.3.2
Enter password:
21/06/17 10:22:13 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
21/06/17 10:22:13 INFO tool.CodeGenTool: Beginning code generation
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via
 the SPI and manual loading of the driver class is generally unnecessary.
21/06/17 10:22:17 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `E_NY_TRAN_DATA` AS t LIMIT 1
21/06/17 10:22:17 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `E_NY_TRAN_DATA` AS t LIMIT 1
21/06/17 10:22:17 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /opt/cloudera/parcels/CDH/lib/hadoop-mapreduce
21/06/17 10:22:21 INFO orm.CompilationManager: Writing jar file: /tmp/s[redacted]/compile/f445edd7fe91b82ffca5767301322c98/E_NY_TRAN_DATA.jar
21/06/17 10:22:21 WARN manager.MySQLManager: It looks like you are importing from mysql.
```

Verify data in HDFS

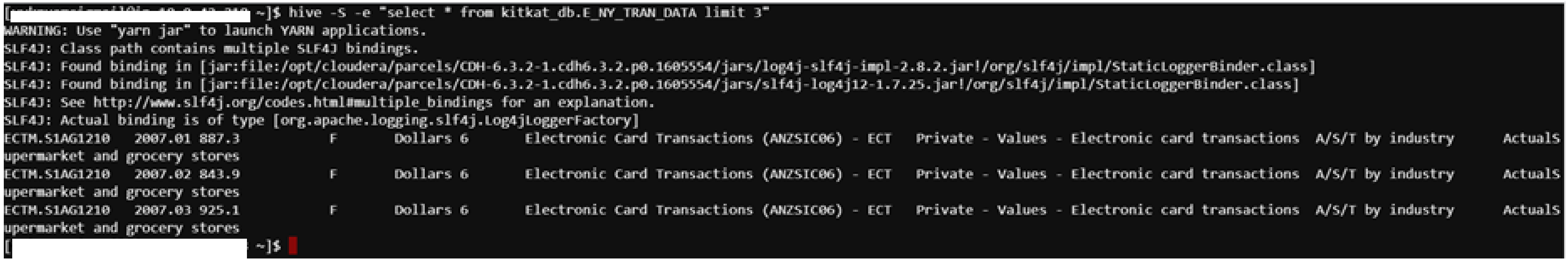


```
[[redacted] kitkat]$ hdfs dfs -ls kitkat01
Found 5 items
-rw-r--r--   3 [redacted] hadoop          0 2021-06-17 10:22 kitkat01/_SUCCESS
-rw-r--r--   3            hadoop      94070 2021-06-17 10:22 kitkat01/part-m-00000
-rw-r--r--   3            hadoop          0 2021-06-17 10:22 kitkat01/part-m-00001
-rw-r--r--   3            hadoop          0 2021-06-17 10:22 kitkat01/part-m-00002
-rw-r--r--   3            hadoop      98625 2021-06-17 10:22 kitkat01/part-m-00003
[[redacted] kitkat]$
```

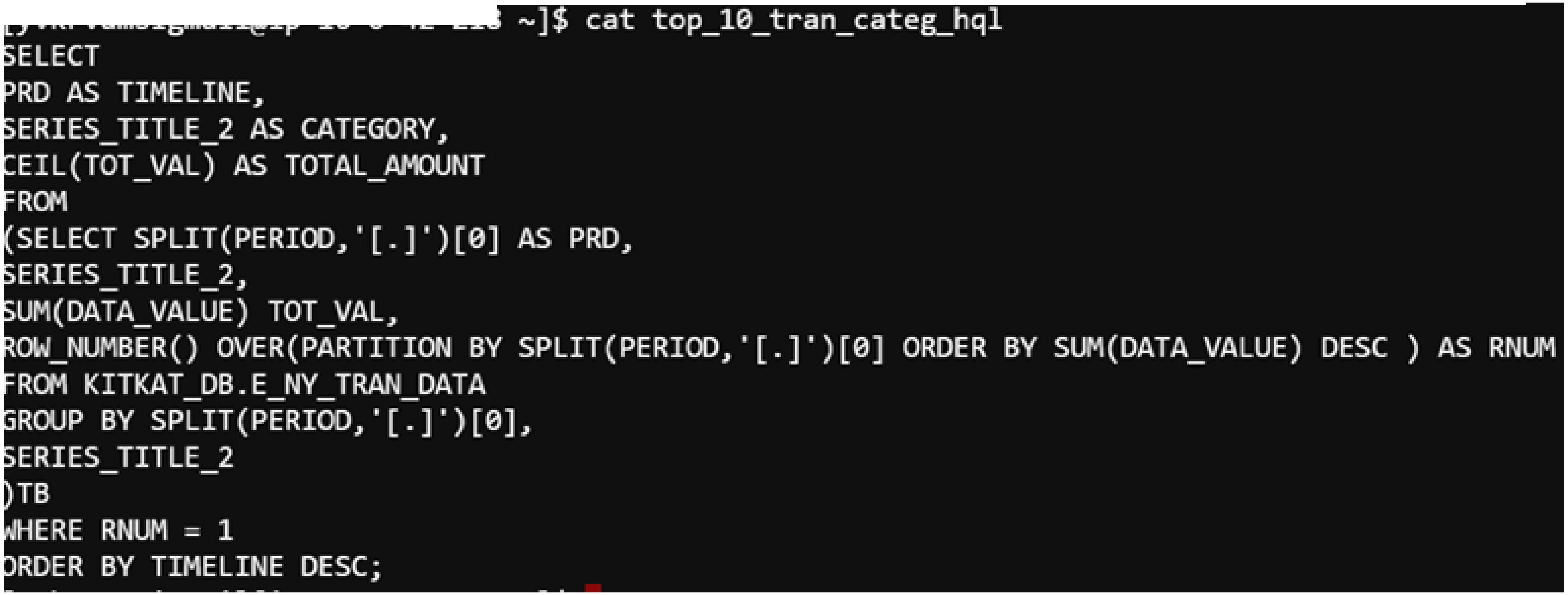


```
[[redacted]log]$ hive -e "use kitkat_db; create external table kitkat_db.E_NY_TRAN_DATA(Series_reference string,
Period string,
Data_value string,
Suppressed string,
STATUS string,
UNITS string,
Magnitude string,
Subject string,
Group_ string,
Series_title_1 string,
Series_title_2 string
)row format delimited fields terminated by ',' STORED AS ORC location '/user/[redacted]/kitkat01' ";
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinde
r.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.
class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/hive-common-2.1.1-cdh6.3.2.jar!/hive-log4j
2.properties Async: false
OK
Time taken: 2.01 seconds
OK
Time taken: 0.668 seconds
[[redacted]og]$
```

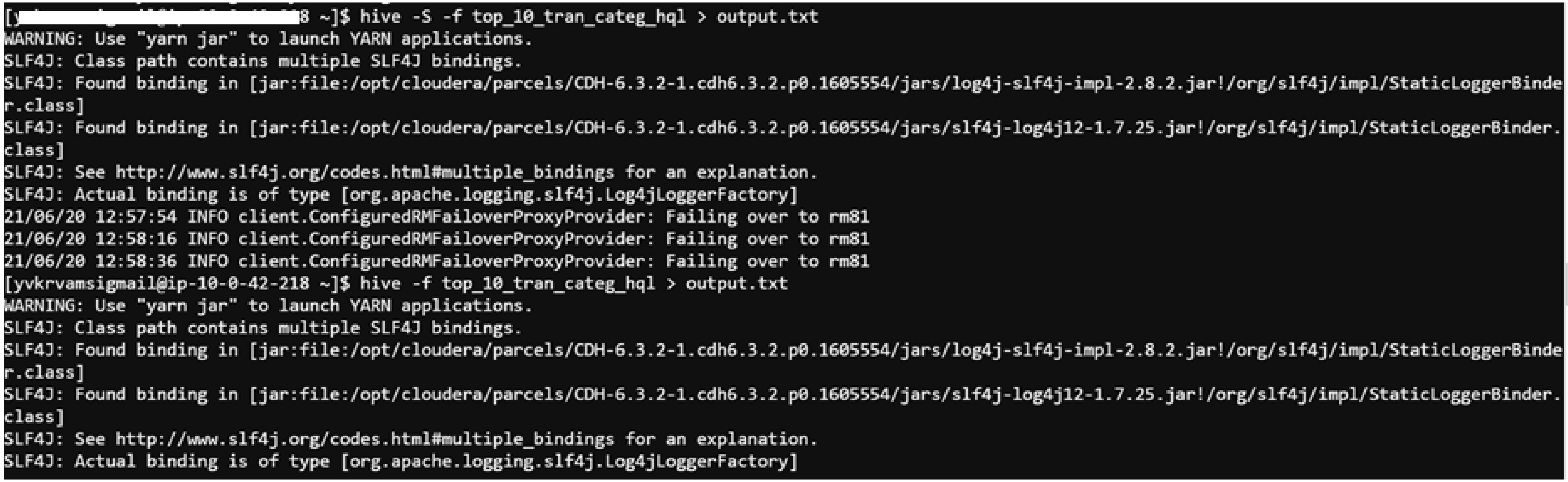
Validate data in hive table:



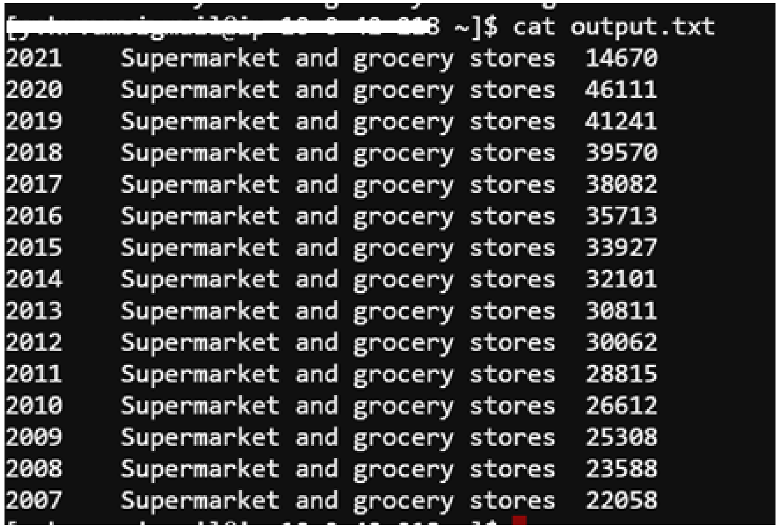
```
[[redacted] ~]$ hive -S -e "select * from kitkat_db.E_NY_TRAN_DATA limit 3"
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
ECTM.S1AG1210   2007.01 887.3           F       Dollars 6       Electronic Card Transactions (ANZSIC06) - ECT   Private - Values - Electronic card transactions  A/S/T by industry     ActualS
upermarket and grocery stores
ECTM.S1AG1210   2007.02 843.9           F       Dollars 6       Electronic Card Transactions (ANZSIC06) - ECT   Private - Values - Electronic card transactions  A/S/T by industry     ActualS
upermarket and grocery stores
ECTM.S1AG1210   2007.03 925.1           F       Dollars 6       Electronic Card Transactions (ANZSIC06) - ECT   Private - Values - Electronic card transactions  A/S/T by industry     ActualS
upermarket and grocery stores
[[redacted] ~]$
```



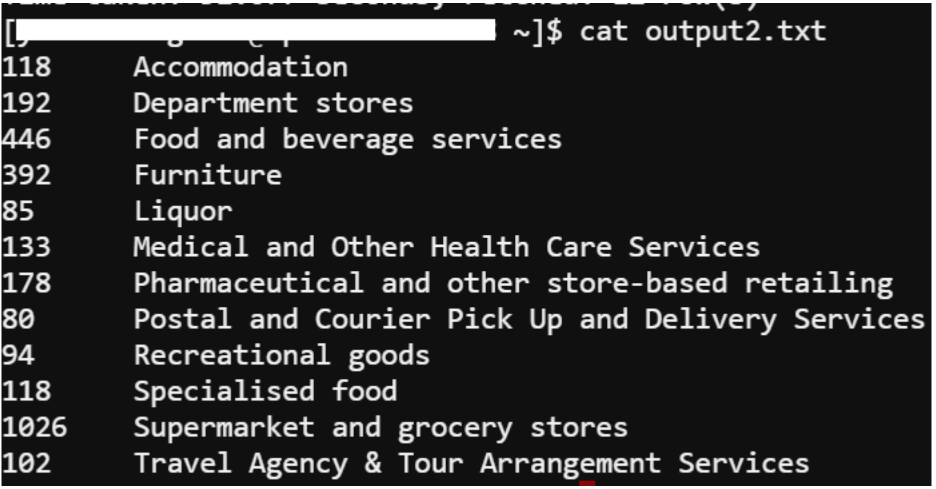
```
[redacted] ~]$ cat top_10_tran_categ_hql
SELECT
PRD AS TIMELINE,
SERIES_TITLE_2 AS CATEGORY,
CEIL(TOT_VAL) AS TOTAL_AMOUNT
FROM
(SELECT SPLIT(PERIOD,'[.]')[0] AS PRD,
SERIES_TITLE_2,
SUM(DATA_VALUE) TOT_VAL,
ROW_NUMBER() OVER(PARTITION BY SPLIT(PERIOD,'[.]')[0] ORDER BY SUM(DATA_VALUE) DESC ) AS RNUM
FROM KITKAT_DB.E_NY_TRAN_DATA
GROUP BY SPLIT(PERIOD,'[.]')[0],
SERIES_TITLE_2
)TB
WHERE RNUM = 1
ORDER BY TIMELINE DESC;
```



```
[[redacted]8 ~]$ hive -S -f top_10_tran_categ_hql > output.txt
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinde
r.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.
class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
21/06/20 12:57:54 INFO client.ConfiguredRMFailoverProxyProvider: Failing over to rm81
21/06/20 12:58:16 INFO client.ConfiguredRMFailoverProxyProvider: Failing over to rm81
21/06/20 12:58:36 INFO client.ConfiguredRMFailoverProxyProvider: Failing over to rm81
[yvkrvamsigmail@ip-10-0-42-218 ~]$ hive -f top_10_tran_categ_hql > output.txt
WARNING: Use "yarn jar" to launch YARN applications.
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/log4j-slf4j-impl-2.8.2.jar!/org/slf4j/impl/StaticLoggerBinde
r.class]
SLF4J: Found binding in [jar:file:/opt/cloudera/parcels/CDH-6.3.2-1.cdh6.3.2.p0.1605554/jars/slf4j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.
class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
```



```
[[redacted] ~]$ cat output.txt
2021    Supermarket and grocery stores  14670
2020    Supermarket and grocery stores  46111
2019    Supermarket and grocery stores  41241
2018    Supermarket and grocery stores  39570
2017    Supermarket and grocery stores  38082
2016    Supermarket and grocery stores  35713
2015    Supermarket and grocery stores  33927
2014    Supermarket and grocery stores  32101
2013    Supermarket and grocery stores  30811
2012    Supermarket and grocery stores  30062
2011    Supermarket and grocery stores  28815
2010    Supermarket and grocery stores  26612
2009    Supermarket and grocery stores  25308
2008    Supermarket and grocery stores  23588
2007    Supermarket and grocery stores  22058
```



```
[[redacted] ~]$ cat output2.txt
118     Accommodation
192     Department stores
446     Food and beverage services
392     Furniture
85      Liquor
133     Medical and Other Health Care Services
178     Pharmaceutical and other store-based retailing
80      Postal and Courier Pick Up and Delivery Services
94      Recreational goods
118     Specialised food
1026    Supermarket and grocery stores
102     Travel Agency & Tour Arrangement Services
```

Conclusion:

- **KPI 1**: The Expenditure of ***Supermarket and grocery stores*** remains the top highest spend in each year since 2007 to 2020. Considering 2020 – only 5 months data is available. There has been 100% increase in the expenditure. Total expense in Supermarket has more than doubled since 2007.
- **KPI 2**: Average Expenses remains the second highest for “Food and beverage services” and the lowest is “Postal and Courier.”. Overall, the average gives us an idea of consumer expenditure behaviour.