# DATA INGESTION

It is because this step was done by upGrad.
All csv files have been uploaded into S3. So in this step I will perform loading data from csv file to Spark data frame.

Firstly, connect PySpark:

```
pyspark --conf "spark.mongodb.read.connection.uri=mongodb://
172.31.67.33:27017/transaction_db.card_transactions?
readPreference=primaryPreferred" --conf
"spark.mongodb.write.connection.uri=mongodb://
172.31.67.33:27017/transaction_db.tb_lookup" --packages
org.mongodb.spark:mongo-spark-connector_2.12:10.1.1
```

Load data from csv file to data frame.

```
df_card =
spark.read.options(inferSchema='True',header='True').csv('s3:
//history-transactions/card_member.csv')

df_score =
spark.read.options(inferSchema=‘True',header='True').csv('s3:
//history-transactions/member_score.csv')
```

Check if the data from csv files have been loaded correctly.

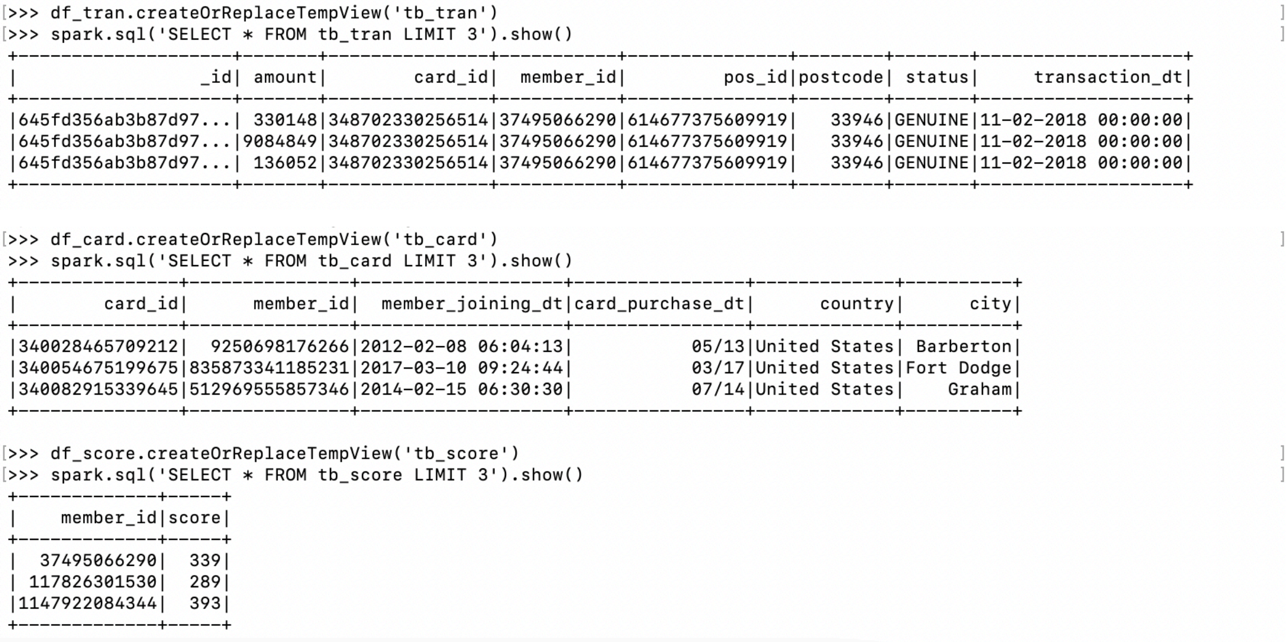
```
>>> df_card.printSchema()
root
 |-- card_id: long (nullable = true)
 |-- member_id: long (nullable = true)
 |-- member_joining_dt: timestamp (nullable = true)
 |-- card_purchase_dt: string (nullable = true)
 |-- country: string (nullable = true)
 |-- city: string (nullable = true)

>>> df_score.printSchema()
root
 |-- member_id: long (nullable = true)
 |-- score: integer (nullable = true)
```

We also need to load data from MongoDB to Spark data frame.

```
df_tran = spark.read.format("mongodb").load()

>>> df_tran.printSchema()
root
 |-- _id: string (nullable = true)
 |-- amount: integer (nullable = true)
 |-- card_id: long (nullable = true)
 |-- member_id: long (nullable = true)
 |-- pos_id: long (nullable = true)
 |-- postcode: integer (nullable = true)
 |-- status: string (nullable = true)
 |-- transaction_dt: string (nullable = true)
```

After that, convert data frame to table.

```
>>> df_tran.createOrReplaceTempView('tb_tran')
>>> spark.sql('SELECT * FROM tb_tran LIMIT 3').show()

>>> df_card.createOrReplaceTempView('tb_card')
>>> spark.sql('SELECT * FROM tb_card LIMIT 3').show()

>>> df_score.createOrReplaceTempView('tb_score')
>>> spark.sql('SELECT * FROM tb_score LIMIT 3').show()
```



```
[>>> df_tran.createOrReplaceTempView('tb_tran')
[>>> spark.sql('SELECT * FROM tb_tran LIMIT 3').show()
+--------------------+-------+---------------+-----------+---------------+--------+-------+-------------------+
|                 _id| amount|        card_id|  member_id|         pos_id|postcode| status|     transaction_dt|
+--------------------+-------+---------------+-----------+---------------+--------+-------+-------------------+
|645fd356ab3b87d97...| 330148|348702330256514|37495066290|614677375609919|   33946|GENUINE|11-02-2018 00:00:00|
|645fd356ab3b87d97...|9084849|348702330256514|37495066290|614677375609919|   33946|GENUINE|11-02-2018 00:00:00|
|645fd356ab3b87d97...| 136052|348702330256514|37495066290|614677375609919|   33946|GENUINE|11-02-2018 00:00:00|
+--------------------+-------+---------------+-----------+---------------+--------+-------+-------------------+

[>>> df_card.createOrReplaceTempView('tb_card')
>>> spark.sql('SELECT * FROM tb_card LIMIT 3').show()
+---------------+---------------+-------------------+----------------+-------------+----------+
|        card_id|      member_id|  member_joining_dt|card_purchase_dt|      country|      city|
+---------------+---------------+-------------------+----------------+-------------+----------+
|340028465709212|  9250698176266|2012-02-08 06:04:13|           05/13|United States| Barberton|
|340054675199675|835873341185231|2017-03-10 09:24:44|           03/17|United States|Fort Dodge|
|340082915339645|512969555857346|2014-02-15 06:30:30|           07/14|United States|    Graham|
+---------------+---------------+-------------------+----------------+-------------+----------+

[>>> df_score.createOrReplaceTempView('tb_score')
[>>> spark.sql('SELECT * FROM tb_score LIMIT 3').show()
+-------------+-----+
|    member_id|score|
+-------------+-----+
|  37495066290|  339|
| 117826301530|  289|
|1147922084344|  393|
+-------------+-----+
```