this chapter is best to use [databricks community cluster (https://community.cloud.databricks.com)](https://community.cloud.databricks.com) to practise

see `../SparkSQL.sql` and `../SparkSQL.html`

one can use `spark-sql` CLI to enter `SQL` command directly

- [Table Types in Spark: External or Managed? (http://www.gatorsmile.io/table-types-in-spark-external-or-managed/)](http://www.gatorsmile.io/table-types-in-spark-external-or-managed/)

In [1]:

```python
from pyspark.sql import SparkSession
import pyspark.sql.functions as F
from pyspark.sql.types import *

spark = SparkSession\
    .builder\
    .appName("chapter-10-data-src")\
    .enableHiveSupport()\
    .getOrCreate()

import os
SPARK_BOOK_DATA_PATH = os.environ['SPARK_BOOK_DATA_PATH']
```

In [2]:

```python
spark
```

Out[2]:

SparkSession - hive

SparkContext

[Spark UI (http://192.168.0.207:4040)](http://192.168.0.207:4040)

Version

`v3.0.1`

Master

`local[*]`

AppName

`PySparkShell`

In [3]:

```python
file_path = SPARK_BOOK_DATA_PATH + "/data/flight-data/json/2015-summ
file_path
```

Out[3]:

```
'/home/wengong/spark_data//data/flight-data/json/2015-summary.json'
```

In [4]:

```python
spark.read.json(file_path)\
  .createOrReplaceTempView("flight_data") # DF => SQL
```

In [5]:

```
df = spark.sql("""
SELECT DEST_COUNTRY_NAME, sum(count)
FROM flight_data GROUP BY DEST_COUNTRY_NAME
""")\
  .where("DEST_COUNTRY_NAME like 'S%'")\
  .where("`sum(count)` > 10")
# SQL => DF


df.show(5)
```

```
+--------------------+----------+
|   DEST_COUNTRY_NAME|sum(count)|
+--------------------+----------+
|             Senegal|        40|
|              Sweden|       118|
|               Spain|       420|
|    Saint Barthelemy|        39|
|Saint Kitts and N...|       139|
+--------------------+----------+
only showing top 5 rows
```

Database

In [15]:

```
spark.sql("show databases;").show()
```

```
+---------+
|namespace|
+---------+
|  default|
+---------+
```

In [12]:

```
spark.sql("use default").show()
```

```
++
||
++
++
```

In [13]:

```
spark.sql("SELECT current_database()").show()
```

```
+------------------+
|current_database()|
+------------------+
|           default|
+------------------+
```

In [8]:

```
spark.sql("create database my_db").show()
```

```
++
||
++
++
```

In [10]:

```
spark.sql("use my_db").show()
```

```
++
||
++
++
```

In [11]:

```
spark.sql("SELECT current_database()").show()
```

```
+------------------+
|current_database()|
+------------------+
|             my_db|
+------------------+
```

In [14]:

```
spark.sql("drop database if exists my_db").show()
```

```
++
||
++
++
```

Table

In [16]:

```
spark.sql("use default")

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

```
+--------+-------------------+-----------+
|database|          tableName|isTemporary|
+--------+-------------------+-----------+
| default|            flights|      false|
| default|        flights_csv|      false|
| default|flights_from_select|      false|
| default|       hive_flights|      false|
| default|     hive_flights_2|      false|
| default|      just_usa_view|      false|
| default|partitioned_flights|      false|
|        |        flight_data|       true|
+--------+-------------------+-----------+
```

In [9]:

```
spark.sql("drop table flights")
```

Out[9]:

```
DataFrame[]
```

In [10]:

```
spark.sql("drop table hive_flights")
```

Out[10]:

```
DataFrame[]
```

In [11]:

```
spark.sql("drop table hive_flights_2")
```

Out[11]:

```
DataFrame[]
```

In [13]:

```
spark.sql("""
CREATE TABLE flights (
  DEST_COUNTRY_NAME STRING, ORIGIN_COUNTRY_NAME STRING, count LONG)
USING JSON OPTIONS (path '/home/wengong/spark_data//data/flight-data
""").show()
```

```
++
||
++
++
```

In [15]:

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

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|    United States|            Romania|   15|
|    United States|            Croatia|    1|
|    United States|            Ireland|  344|
|            Egypt|      United States|   15|
|    United States|              India|   62|
+-----------------+-------------------+-----+
```

In [16]:

```
sql_stmt = f"""CREATE TABLE flights_csv (
  DEST_COUNTRY_NAME STRING,
  ORIGIN_COUNTRY_NAME STRING COMMENT "remember, the US will be most
  count LONG)
USING csv OPTIONS (header true, path '{SPARK_BOOK_DATA_PATH}data/fl
"""
```

In [17]:
```
sql_stmt
```

Out[17]:
```
'CREATE TABLE flights_csv (\n  DEST_COUNTRY_NAME STRING,\n  ORIGIN_COU
NTRY_NAME STRING COMMENT "remember, the US will be most prevalent",\n  
count LONG)\nUSING csv OPTIONS (header true, path \'/home/wengong/spar
k_data/data/flight-data/csv/2015-summary.csv\')\n'
```

In [18]:
```
spark.sql(sql_stmt).show()
```

```
++
||
++
++
```

In [19]:
```
spark.sql("CREATE TABLE if not exists flights_from_select USING parq
```

Out[19]:
```
DataFrame[]
```

In [30]:
```
spark.sql("CREATE TABLE if not exists flights_from_select2 USING pa
```

Out[30]:
```
DataFrame[]
```

In [20]:
```
spark.sql("""
CREATE TABLE partitioned_flights USING parquet PARTITIONED BY (DEST_
AS SELECT DEST_COUNTRY_NAME, ORIGIN_COUNTRY_NAME, count FROM flights
""")
```

Out[20]:
```
DataFrame[]
```

In [22]:
```
spark.sql("describe table partitioned_flights").show()
```

```
+--------------------+---------+-------+
|            col_name|data_type|comment|
+--------------------+---------+-------+
| ORIGIN_COUNTRY_NAME|   string|   null|
|               count|   bigint|   null|
|   DEST_COUNTRY_NAME|   string|   null|
|# Partition Infor...|         |       |
|          # col_name|data_type|comment|
|   DEST_COUNTRY_NAME|   string|   null|
+--------------------+---------+-------+
```

In [23]:

```
spark.sql("describe table flights").show()
```

```
+-------------------+---------+-------+
|           col_name|data_type|comment|
+-------------------+---------+-------+
|  DEST_COUNTRY_NAME|   string|   null|
|ORIGIN_COUNTRY_NAME|   string|   null|
|              count|   bigint|   null|
+-------------------+---------+-------+
```

```
spark.sql("""
INSERT INTO partitioned_flights(count,ORIGIN_COUNTRY_NAME)
  PARTITION (DEST_COUNTRY_NAME="UNITED STATES")
  SELECT count, ORIGIN_COUNTRY_NAME FROM flights
  WHERE DEST_COUNTRY_NAME='UNITED STATES' LIMIT 12
""").show()
```

In [25]:

```
spark.sql("select * from partitioned_flights").show()
```

```
+-------------------+-----+-----------------+
|ORIGIN_COUNTRY_NAME|count|DEST_COUNTRY_NAME|
+-------------------+-----+-----------------+
|      United States|   15|            Egypt|
|            Romania|   15|    United States|
|            Croatia|    1|    United States|
|            Ireland|  344|    United States|
|              India|   62|    United States|
+-------------------+-----+-----------------+
```

In [26]:

```
spark.sql("SHOW PARTITIONS partitioned_flights").show(truncate=False
```

```
+--------------------------------+
|partition                       |
+--------------------------------+
|DEST_COUNTRY_NAME=Egypt         |
|DEST_COUNTRY_NAME=United States|
+--------------------------------+
```

In [27]:

```
sql_stmt=f"""
CREATE EXTERNAL TABLE hive_flights (
  DEST_COUNTRY_NAME STRING, ORIGIN_COUNTRY_NAME STRING, count LONG)
ROW FORMAT DELIMITED FIELDS TERMINATED BY ',' LOCATION '{SPARK_BOOK_
"""
sql_stmt
```

Out[27]:

```
"\nCREATE EXTERNAL TABLE hive_flights (\n  DEST_COUNTRY_NAME STRING, O
RIGIN_COUNTRY_NAME STRING, count LONG)\nROW FORMAT DELIMITED FIELDS TE
RMINATED BY ',' LOCATION '/home/wengong/spark_data/data/flight-data-hi
ve/'\n"
```

In [28]:

```
spark.sql(sql_stmt)
```

Out[28]:

```
DataFrame[]
```

In [29]:

```
df = spark.sql("select * from hive_flights")
df.show()
```

```
+--------------------+-------------------+-----+
|   DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+--------------------+-------------------+-----+
|       United States|            Romania|   15|
|       United States|            Croatia|    1|
|       United States|            Ireland|  344|
|               Egypt|      United States|   15|
|       United States|              India|   62|
|       United States|          Singapore|    1|
|       United States|            Grenada|   62|
|          Costa Rica|      United States|  588|
|             Senegal|      United States|   40|
|             Moldova|      United States|    1|
|       United States|       Sint Maarten|  325|
|       United States|   Marshall Islands|   39|
|              Guyana|      United States|   64|
|               Malta|      United States|    1|
|            Anguilla|      United States|   41|
|             Bolivia|      United States|   30|
|       United States|           Paraguay|    6|
|             Algeria|      United States|    4|
|Turks and Caicos ...|      United States|  230|
|       United States|          Gibraltar|    1|
+--------------------+-------------------+-----+
only showing top 20 rows
```

In [30]:

```
spark.sql("DESCRIBE TABLE hive_flights").show()
```

```
+-------------------+---------+-------+
|           col_name|data_type|comment|
+-------------------+---------+-------+
|  DEST_COUNTRY_NAME|   string|   null|
|ORIGIN_COUNTRY_NAME|   string|   null|
|              count|   bigint|   null|
+-------------------+---------+-------+
```

In [31]:

```
sql_stmt = f"""
CREATE EXTERNAL TABLE hive_flights_2
ROW FORMAT DELIMITED FIELDS TERMINATED BY ','
LOCATION '{SPARK_BOOK_DATA_PATH}/data/flight-data-hive/' AS SELECT
"""
spark.sql(sql_stmt)
```

Out[31]:

```
DataFrame[]
```

In [32]:

```
df = spark.sql("select * from hive_flights_2 limit 5")
df.show()
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|    United States|            Romania|   15|
|    United States|            Croatia|    1|
|    United States|            Ireland|  344|
|            Egypt|      United States|   15|
|    United States|              India|   62|
+-----------------+-------------------+-----+
```

In [33]:

```
spark.sql("select * from flights_from_select").show()
```

```
+--------------------+-------------------+-----+
|   DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+--------------------+-------------------+-----+
|       United States|            Romania|   15|
|       United States|            Croatia|    1|
|       United States|            Ireland|  344|
|               Egypt|      United States|   15|
|       United States|              India|   62|
|       United States|          Singapore|    1|
|       United States|            Grenada|   62|
|          Costa Rica|      United States|  588|
|             Senegal|      United States|   40|
|             Moldova|      United States|    1|
|       United States|       Sint Maarten|  325|
|       United States|   Marshall Islands|   39|
|              Guyana|      United States|   64|
|               Malta|      United States|    1|
|            Anguilla|      United States|   41|
|             Bolivia|      United States|   30|
|       United States|           Paraguay|    6|
|             Algeria|      United States|    4|
|Turks and Caicos ...|      United States|  230|
|       United States|          Gibraltar|    1|
+--------------------+-------------------+-----+
only showing top 20 rows
```

```
INSERT INTO flights_from_select
  SELECT DEST_COUNTRY_NAME, ORIGIN_COUNTRY_NAME, count FROM flights LIMIT 20
```

In [34]:

```
spark.sql("REFRESH table partitioned_flights")
```

Out[34]:

```
DataFrame[]
```

In [ ]:

```
-- COMMAND ----------

MSCK REPAIR TABLE partitioned_flights


-- COMMAND ----------

DROP TABLE flights_csv;


-- COMMAND ----------

DROP TABLE IF EXISTS flights_csv;


-- COMMAND ----------

CACHE TABLE flights


-- COMMAND ----------

UNCACHE TABLE FLIGHTS


-- COMMAND ----------
```

In [35]:

```
spark.sql("""CREATE VIEW just_usa_view AS
  SELECT * FROM flights WHERE dest_country_name = 'United States'
""")
df = spark.sql("select * from just_usa_view limit 5")
df.show()
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|    United States|            Romania|   15|
|    United States|            Croatia|    1|
|    United States|            Ireland|  344|
|    United States|              India|   62|
|    United States|          Singapore|    1|
+-----------------+-------------------+-----+
```

View

In [17]:

```
spark.sql("""
CREATE TEMP VIEW just_usa_view_temp AS
  SELECT * FROM flights WHERE dest_country_name = 'United States'
""")

spark.sql("""
CREATE GLOBAL TEMP VIEW just_usa_global_view_temp AS
  SELECT * FROM flights WHERE dest_country_name = 'United States'
""")
```

Out[17]:

```
DataFrame[]
```

In [18]:

```
spark.sql("SHOW TABLES").show()
```

```
+--------+-------------------+-----------+
|database|          tableName|isTemporary|
+--------+-------------------+-----------+
| default|            flights|      false|
| default|        flights_csv|      false|
| default|flights_from_select|      false|
| default|       hive_flights|      false|
| default|     hive_flights_2|      false|
| default|      just_usa_view|      false|
| default|partitioned_flights|      false|
|        |        flight_data|       true|
|        | just_usa_view_temp|       true|
+--------+-------------------+-----------+
```

In [19]:

```
spark.sql("select * from flights limit 3").show()
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|    United States|            Romania|   15|
|    United States|            Croatia|    1|
|    United States|            Ireland|  344|
+-----------------+-------------------+-----+
```

In [21]:

```
spark.sql("""
CREATE OR REPLACE TEMP VIEW just_usa_view_temp AS
  SELECT * FROM flights WHERE dest_country_name = 'United States'
""")
```

Out[21]:

```
DataFrame[]
```

In [22]:

```
spark.sql("SELECT * FROM just_usa_view_temp").show()
```

```
+-----------------+--------------------+-----+
|DEST_COUNTRY_NAME| ORIGIN_COUNTRY_NAME|count|
+-----------------+--------------------+-----+
|    United States|             Romania|   15|
|    United States|             Croatia|    1|
|    United States|             Ireland|  344|
|    United States|               India|   62|
|    United States|           Singapore|    1|
|    United States|             Grenada|   62|
|    United States|        Sint Maarten|  325|
|    United States|    Marshall Islands|   39|
|    United States|            Paraguay|    6|
|    United States|           Gibraltar|    1|
|    United States|Federated States ...|   69|
|    United States|              Russia|  161|
|    United States|         Netherlands|  660|
|    United States|             Senegal|   42|
|    United States|              Angola|   13|
|    United States|            Anguilla|   38|
|    United States|             Ecuador|  300|
|    United States|              Cyprus|    1|
|    United States|            Portugal|  134|
|    United States|          Costa Rica|  608|
+-----------------+--------------------+-----+
only showing top 20 rows
```

In [24]:

```python
spark.sql("EXPLAIN SELECT * FROM just_usa_view").show(truncate=False
```

```
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
|plan
|
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
|== Physical Plan ==
*(1) Project [DEST_COUNTRY_NAME#134, ORIGIN_COUNTRY_NAME#135, count#13
6L]
+- *(1) Filter (isnotnull(dest_country_name#134) AND (dest_country_nam
e#134 = United States))
   +- FileScan json default.flights[DEST_COUNTRY_NAME#134,ORIGIN_COUNT
RY_NAME#135,count#136L] Batched: false, DataFilters: [isnotnull(DEST_C
OUNTRY_NAME#134), (DEST_COUNTRY_NAME#134 = United States)], Format: JS
ON, Location: InMemoryFileIndex[file:/home/wengong/spark_data/data/fli
ght-data/json/2015-summary.json], PartitionFilters: [], PushedFilters:
[IsNotNull(DEST_COUNTRY_NAME), EqualTo(DEST_COUNTRY_NAME,United State
s)], ReadSchema: struct<DEST_COUNTRY_NAME:string,ORIGIN_COUNTRY_NAME:s
tring,count:bigint>

|
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
```

In [25]:

```
spark.sql("EXPLAIN SELECT * FROM flights WHERE dest_country_name =
```

```
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
|plan
|
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
|== Physical Plan ==
*(1) Project [DEST_COUNTRY_NAME#134, ORIGIN_COUNTRY_NAME#135, count#13
6L]
+- *(1) Filter (isnotnull(dest_country_name#134) AND (dest_country_nam
e#134 = United States))
   +- FileScan json default.flights[DEST_COUNTRY_NAME#134,ORIGIN_COUNT
RY_NAME#135,count#136L] Batched: false, DataFilters: [isnotnull(DEST_C
OUNTRY_NAME#134), (DEST_COUNTRY_NAME#134 = United States)], Format: JS
ON, Location: InMemoryFileIndex[file:/home/wengong/spark_data/data/fli
ght-data/json/2015-summary.json], PartitionFilters: [], PushedFilters:
[IsNotNull(DEST_COUNTRY_NAME), EqualTo(DEST_COUNTRY_NAME,United State
s)], ReadSchema: struct<DEST_COUNTRY_NAME:string,ORIGIN_COUNTRY_NAME:s
tring,count:bigint>

|
+---------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
----------------------------------------------------------------------
+
```

In [28]:

```
spark.sql("DROP VIEW IF EXISTS just_usa_view;")
spark.sql("show tables").show()
```

```
+--------+------------------+-----------+
|database|         tableName|isTemporary|
+--------+------------------+-----------+
| default|           flights|      false|
| default|       flights_csv|      false|
| default|flights_from_select|      false|
| default|      hive_flights|      false|
| default|    hive_flights_2|      false|
| default|partitioned_flights|      false|
|        |       flight_data|       true|
|        | just_usa_view_temp|       true|
+--------+------------------+-----------+
```

SELECT Syntax

```

SELECT [ALL|DISTINCT] named_expression[, named_expression, ...]
    FROM relation[, relation, ...]
    [lateral_view[, lateral_view, ...]]
    [WHERE boolean_expression]
    [aggregation [HAVING boolean_expression]]
    [ORDER BY sort_expressions]
    [CLUSTER BY expressions]
    [DISTRIBUTE BY expressions]
    [SORT BY sort_expressions]
    [WINDOW named_window[, WINDOW named_window, ...]]
    [LIMIT num_rows]

named_expression:
    : expression [AS alias]

relation:
    | join_relation
    | (table_name|query|relation) [sample] [AS alias]
    : VALUES (expressions)[, (expressions), ...]
          [AS (column_name[, column_name, ...])]

expressions:
    : expression[, expression, ...]

sort_expressions:
    : expression [ASC|DESC][, expression [ASC|DESC], ...]


```

```
spark.sql("""
<add SQL Statement here>
""").show(5)
```

In [33]:

```
spark.sql("""
SELECT
 *
FROM flights
""").show(5)
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|    United States|            Romania|   15|
|    United States|            Croatia|    1|
|    United States|            Ireland|  344|
|            Egypt|      United States|   15|
|    United States|              India|   62|
+-----------------+-------------------+-----+
only showing top 5 rows
```

In [37]:

```
spark.sql("""
SELECT
    *,
  CASE WHEN upper(DEST_COUNTRY_NAME) = 'UNITED STATES' THEN 1
       WHEN DEST_COUNTRY_NAME = 'Egypt' THEN 0
       ELSE -1 END as dest_tag
FROM flights
""").show(10)
```

```
+-----------------+-------------------+-----+--------+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|dest_tag|
+-----------------+-------------------+-----+--------+
|    United States|            Romania|   15|       1|
|    United States|            Croatia|    1|       1|
|    United States|            Ireland|  344|       1|
|            Egypt|      United States|   15|       0|
|    United States|              India|   62|       1|
|    United States|          Singapore|    1|       1|
|    United States|            Grenada|   62|       1|
|       Costa Rica|      United States|  588|      -1|
|          Senegal|      United States|   40|      -1|
|          Moldova|      United States|    1|      -1|
+-----------------+-------------------+-----+--------+
only showing top 10 rows
```

complex type

In [39]:

```
spark.sql("""
CREATE VIEW IF NOT EXISTS nested_data AS
  SELECT (DEST_COUNTRY_NAME, ORIGIN_COUNTRY_NAME) as country, count
""")

spark.sql("""
SELECT * FROM nested_data
""").show(5, False)
```

```
+-----------------------+-----+
|country                |count|
+-----------------------+-----+
|[United States, Romania]|15   |
|[United States, Croatia]|1    |
|[United States, Ireland]|344  |
|[Egypt, United States]  |15   |
|[United States, India]  |62   |
+-----------------------+-----+
only showing top 5 rows
```

In [40]:

```
spark.sql("""
SELECT country.DEST_COUNTRY_NAME, count FROM nested_data
""").show(5, False)

spark.sql("""
SELECT country.*, count FROM nested_data
""").show(5, False)
```

```
+-----------------+-----+
|DEST_COUNTRY_NAME|count|
+-----------------+-----+
|United States    |15   |
|United States    |1    |
|United States    |344  |
|Egypt            |15   |
|United States    |62   |
+-----------------+-----+
only showing top 5 rows

+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|United States    |Romania            |15   |
|United States    |Croatia            |1    |
|United States    |Ireland            |344  |
|Egypt            |United States      |15   |
|United States    |India              |62   |
+-----------------+-------------------+-----+
only showing top 5 rows
```

collect_set(), collect_list()

In [41]:

```
spark.sql("""
SELECT DEST_COUNTRY_NAME as new_name, collect_list(count) as flight_
  collect_set(ORIGIN_COUNTRY_NAME) as origin_set
FROM flights GROUP BY DEST_COUNTRY_NAME
""").show(5, False)

spark.sql("""
SELECT DEST_COUNTRY_NAME, ARRAY(1, 2, 3) FROM flights
""").show(5, False)

spark.sql("""
SELECT DEST_COUNTRY_NAME as new_name, collect_list(count)[0]
FROM flights GROUP BY DEST_COUNTRY_NAME
""").show(5, False)
```

```
+--------+-------------+---------------+
|new_name|flight_counts|origin_set     |
+--------+-------------+---------------+
|Anguilla|[41]         |[United States]|
|Paraguay|[60]         |[United States]|
|Russia  |[176]        |[United States]|
|Senegal |[40]         |[United States]|
|Sweden  |[118]        |[United States]|
+--------+-------------+---------------+
only showing top 5 rows

+-----------------+--------------+
|DEST_COUNTRY_NAME|array(1, 2, 3)|
+-----------------+--------------+
|United States    |[1, 2, 3]     |
|United States    |[1, 2, 3]     |
|United States    |[1, 2, 3]     |
|Egypt            |[1, 2, 3]     |
|United States    |[1, 2, 3]     |
+-----------------+--------------+
only showing top 5 rows

+--------+---------------------+
|new_name|collect_list(count)[0]|
+--------+---------------------+
|Anguilla|41                   |
|Paraguay|60                   |
|Russia  |176                  |
|Senegal |40                   |
|Sweden  |118                  |
+--------+---------------------+
only showing top 5 rows
```

explode()

In [44]:

```

spark.sql("""
CREATE OR REPLACE TEMP VIEW flights_agg AS
  SELECT DEST_COUNTRY_NAME, collect_list(count) as collected_counts
  FROM flights GROUP BY DEST_COUNTRY_NAME
""")


spark.sql("""
SELECT explode(collected_counts), DEST_COUNTRY_NAME FROM flights_agg
""").show(5, False)


```

```
+---+-----------------+
|col|DEST_COUNTRY_NAME|
+---+-----------------+
|41 |Anguilla         |
|60 |Paraguay         |
|176|Russia           |
|40 |Senegal          |
|118|Sweden           |
+---+-----------------+
only showing top 5 rows
```

Function

In [46]:

```
spark.sql("""
SHOW FUNCTIONS
""").show(15, False)

spark.sql("""
SHOW SYSTEM FUNCTIONS
""").show(15, False)

spark.sql("""
SHOW USER FUNCTIONS
""").show(5, False)

spark.sql("""
SHOW FUNCTIONS "s*";
""").show(5, False)

spark.sql("""
SHOW FUNCTIONS LIKE "collect*";
""").show(5, False)
```

```
+--------+
|function|
+--------+
|!       |
|!=      |
|%       |
|&       |
|*       |
|+       |
|-       |
|/       |
|<       |
|<=      |
|<=>     |
|<>      |
|=       |
|==      |
|>       |
+--------+
only showing top 15 rows

+--------+
|function|
+--------+
|!       |
|!=      |
|%       |
|&       |
|*       |
|+       |
|-       |
|/       |
|<       |
|<=      |
|<=>     |
|<>      |
|=       |
|==       |
|>        |
+--------+
only showing top 15 rows

+--------+
|function|
+--------+
+--------+

+--------------+
|function      |
+--------------+
|schema_of_csv |
|schema_of_json|
|second        |
|sentences     |
|sequence      |
+--------------+
only showing top 5 rows

+------------+
|function    |
+------------+
|collect_list|
|collect_set |
+------------+
```

In [47]:

```
def power3(x):
    return x*x*x
```

In [48]:

```
power3(10)
```

Out[48]:

```
1000
```

In [59]:

```
udf_power3 = F.udf(lambda x: x*x*x, LongType())
```

In [69]:

```
udf_power3 = F.udf(lambda x: x*x*x)
```

In [70]:

```python
df = spark.range(10).select("id")
df.show()
```

```
+---+
| id|
+---+
|  0|
|  1|
|  2|
|  3|
|  4|
|  5|
|  6|
|  7|
|  8|
|  9|
+---+
```

In [78]:

```python
df = df.withColumn("id_p3", udf_power3(F.col("id")))\
    .withColumn("id_p1", udf_power3(F.col("id").cast("double")))\
    .withColumn("id_str", F.col("id").cast("string"))
df.show()
df.printSchema()
```

```
+---+-----+-----+------+
| id|id_p3|id_p1|id_str|
+---+-----+-----+------+
|  0|    0|  0.0|     0|
|  1|    1|  1.0|     1|
|  2|    8|  8.0|     2|
|  3|   27| 27.0|     3|
|  4|   64| 64.0|     4|
|  5|  125|125.0|     5|
|  6|  216|216.0|     6|
|  7|  343|343.0|     7|
|  8|  512|512.0|     8|
|  9|  729|729.0|     9|
+---+-----+-----+------+

root
 |-- id: long (nullable = false)
 |-- id_p3: string (nullable = true)
 |-- id_p1: string (nullable = true)
 |-- id_str: string (nullable = false)
```

In [79]:

```python
df.createOrReplaceTempView("id_data")
```

In [80]:

```
spark.sql("select * from id_data").show()
```

```
+---+-----+-----+------+
| id|id_p3|id_p1|id_str|
+---+-----+-----+------+
|  0|    0|  0.0|     0|
|  1|    1|  1.0|     1|
|  2|    8|  8.0|     2|
|  3|   27| 27.0|     3|
|  4|   64| 64.0|     4|
|  5|  125|125.0|     5|
|  6|  216|216.0|     6|
|  7|  343|343.0|     7|
|  8|  512|512.0|     8|
|  9|  729|729.0|     9|
+---+-----+-----+------+
```

In [81]:

```
spark.sql("describe table id_data").show()
```

```
+--------+---------+-------+
|col_name|data_type|comment|
+--------+---------+-------+
|      id|   bigint|   null|
|   id_p3|   string|   null|
|   id_p1|   string|   null|
|  id_str|   string|   null|
+--------+---------+-------+
```

register UDF for SQL use

In [65]:

```
spark.udf.register("udf_power3", udf_power3)
```

Out[65]:

```
<function __main__.<lambda>(x)>
```

In [66]:

```
spark.sql("""
SHOW USER FUNCTIONS
""").show(5, False)
```

```
+----------+
|function  |
+----------+
|udf_power3|
+----------+
```

In [68]:

```
spark.sql("""
SELECT count, udf_power3(count) as count_3 FROM flights
""").show(5, False)
```

```
+-----+--------+
|count|count_3 |
+-----+--------+
|15   |3375    |
|1    |1       |
|344  |40707584|
|15   |3375    |
|62   |238328  |
+-----+--------+
only showing top 5 rows
```

In [83]:

```
spark.sql("""
SELECT dest_country_name FROM flights
GROUP BY dest_country_name ORDER BY sum(count) DESC LIMIT 5
""").show(5, False)
```

```
+-----------------+
|dest_country_name|
+-----------------+
|United States    |
|Canada           |
|Mexico           |
|United Kingdom   |
|Japan            |
+-----------------+
```

In [84]:

```
spark.sql("""
SELECT * FROM flights
WHERE origin_country_name IN (SELECT dest_country_name FROM flights
      GROUP BY dest_country_name ORDER BY sum(count) DESC LIMIT 5)
""").show(5, False)
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|Egypt            |United States      |15   |
|Costa Rica       |United States      |588  |
|Senegal          |United States      |40   |
|Moldova          |United States      |1    |
|Guyana           |United States      |64   |
+-----------------+-------------------+-----+
only showing top 5 rows
```

In [85]:

```
spark.sql("""
SELECT * FROM flights f1
WHERE EXISTS (SELECT 1 FROM flights f2
            WHERE f1.dest_country_name = f2.origin_country_name)
AND EXISTS (SELECT 1 FROM flights f2
            WHERE f2.dest_country_name = f1.origin_country_name)
""").show(5, False)
```

```
+-----------------+-------------------+-----+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|
+-----------------+-------------------+-----+
|United States    |Romania            |15   |
|United States    |Croatia            |1    |
|United States    |Ireland            |344  |
|Egypt            |United States      |15   |
|United States    |India              |62   |
+-----------------+-------------------+-----+
only showing top 5 rows
```

In [86]:

```
spark.sql("""
SELECT *, (SELECT max(count) FROM flights) AS maximum FROM flights
""").show(5, False)
```

```
+-----------------+-------------------+-----+-------+
|DEST_COUNTRY_NAME|ORIGIN_COUNTRY_NAME|count|maximum|
+-----------------+-------------------+-----+-------+
|United States    |Romania            |15   |370002 |
|United States    |Croatia            |1    |370002 |
|United States    |Ireland            |344  |370002 |
|Egypt            |United States      |15   |370002 |
|United States    |India              |62   |370002 |
+-----------------+-------------------+-----+-------+
only showing top 5 rows
```

```
SET spark.sql.shuffle.partitions=20
```