SparkSubmitOperator Demonstration

This document will guide you through the demonstration of the SparkSubmitOperator in Session 2 of the Airflow module.

Prerequisites:

- Hive table created in the HiveOperator demonstration (credit_card.filtered_transaction).
- sample_spark.py(the code explained in the video)
- transactions_per_item.py (The spark application)
- The version of the enabled JDK is 8.

What are we doing?
In this demonstration, we need to create a DAG with one Spark task.

- transactions_per_item - It will execute a spark job to analyze how many transactions a particular item has been a part of.

Please follow the instructions below:

1. Login to your EMR instance.

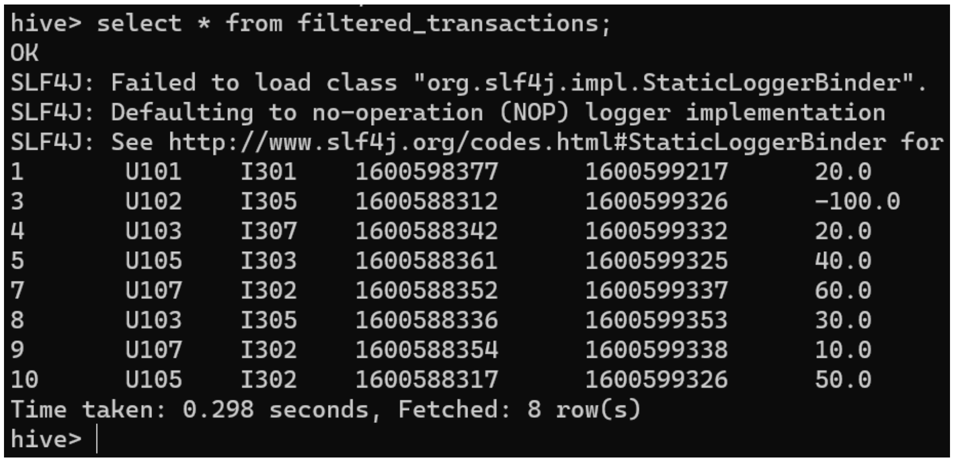
2. Activate the Python virtual environment using the following command:

   source /home/hadoop/airflow/bin/activate

3. Now in this demonstration, we will be using the hive tables we created in the HiveOperator demonstration.

   You can use the following command to check it:

   hive -e “select * from credir_card.filtered_transactions;”



```
hive> select * from filtered_transactions;
OK
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for
1       U101    I301    1600598377      1600599217      20.0
3       U102    I305    1600588312      1600599326      -100.0
4       U103    I307    1600588342      1600599332      20.0
5       U105    I303    1600588361      1600599325      40.0
7       U107    I302    1600588352      1600599337      60.0
8       U103    I305    1600588336      1600599353      30.0
9       U107    I302    1600588354      1600599338      10.0
10      U105    I302    1600588317      1600599326      50.0
Time taken: 0.298 seconds, Fetched: 8 row(s)
hive>
```

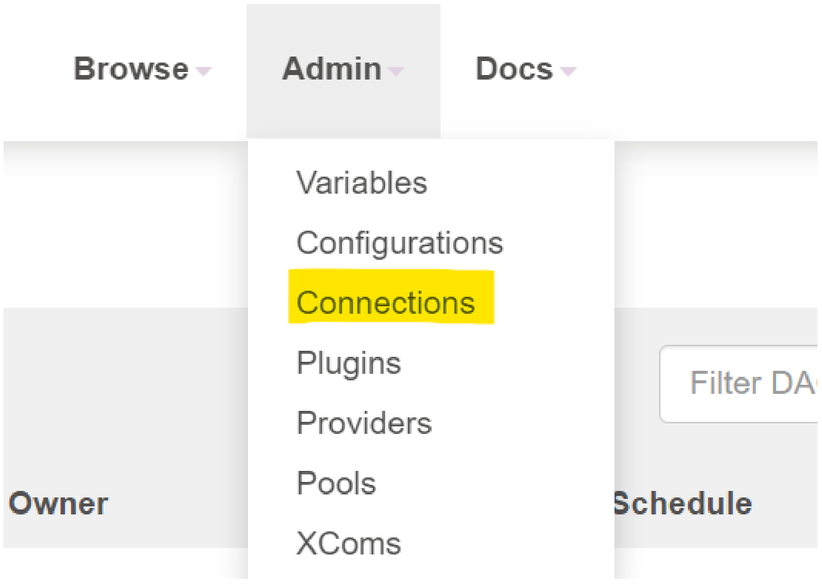
If you don't have these tables, you will have to finish the HiveOperator demonstration first and then you can continue to the next step

4. In the /home/hadoop directory, we need to place the transactions_per_item.py file. (You can use WinSCP or create a new file and paste the code in that file)

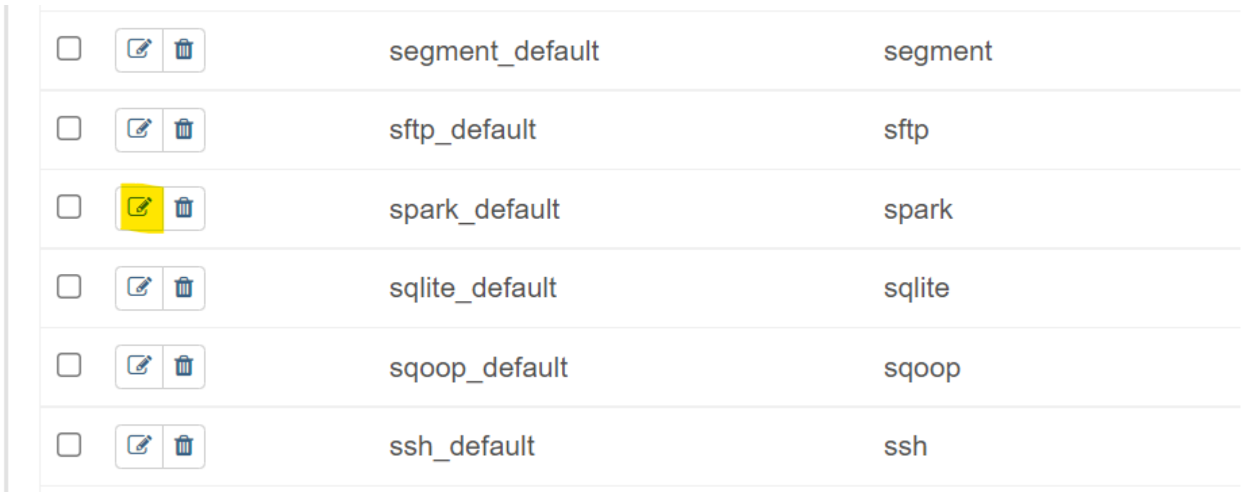
5. Next, you need to set up the Spark connection from the Airflow UI which is hosted in the URL : **your_public_dns:8082**

   Note: You can find you_publin_ip in your AWS EMR dashboard (IPv4 Public DNS))

   Go to the Admin tab and click on Connections

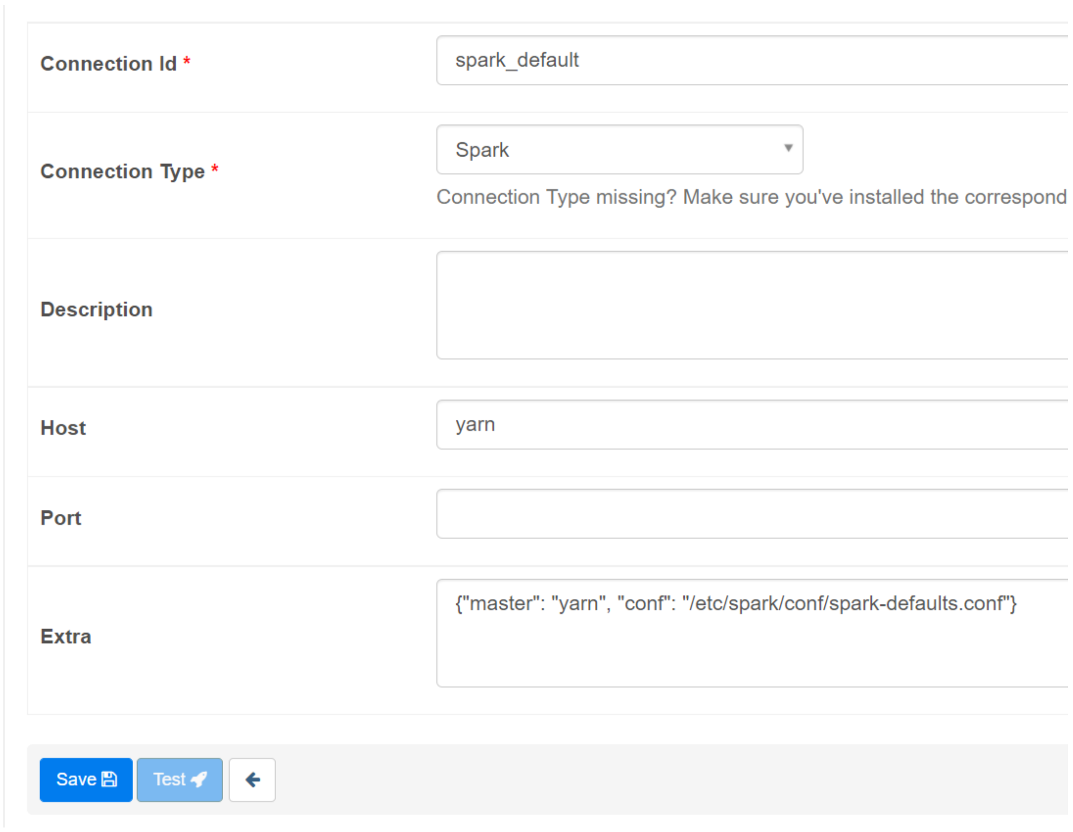


Now click on the edit button to the left of the **spark_default** connection.



☐		segment_default	segment
☐		sftp_default	sftp
☐		spark_default	spark
☐		sqlite_default	sqlite
☐		sqoop_default	sqoop
☐		ssh_default	ssh

Next, fill in the following details and click on Save



Connection Id *	spark_default
Connection Type *	Spark Connection Type missing? Make sure you've installed the correspond
Description	
Host	yarn
Port	
Extra	{"master": "yarn", "conf": "/etc/spark/conf/spark-defaults.conf"}

Conn Id: spark_default

Conn Type: Spark (Select from the drop-down)

Host: yarn

Extra: **{"master": "yarn", "conf": "/etc/spark/conf/spark-defaults.conf"}**

6. Now you need to place the **sample_spark.py** file in the **/home/hadoop/airflow/dags** directory. (You can use WinSCP or create a new file and paste the code in that file)

7. To ensure that the file there are no issues/errors with the file is it considered good practice to compile the program using the following command:

   python sample_spark.py

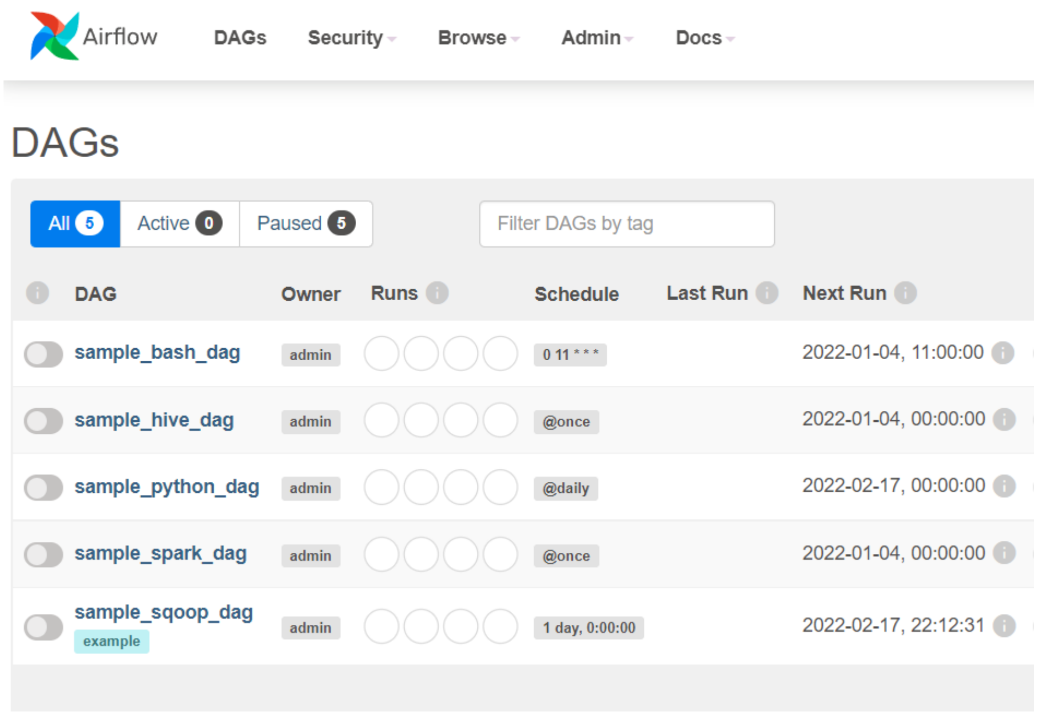
8. You can also use the following command to list the dags in your instance:

   airflow dags list

9. Once you have made sure that your dag file has no issues you can go to the Airflow UI which is hosted in the URL : **your_public_dns:8082**

   Note: You can find you_publin_ip in your AWS EMR dashboard (IPv4 Public DNS)

10. Switch ON the DAG(sample_spark_dag)



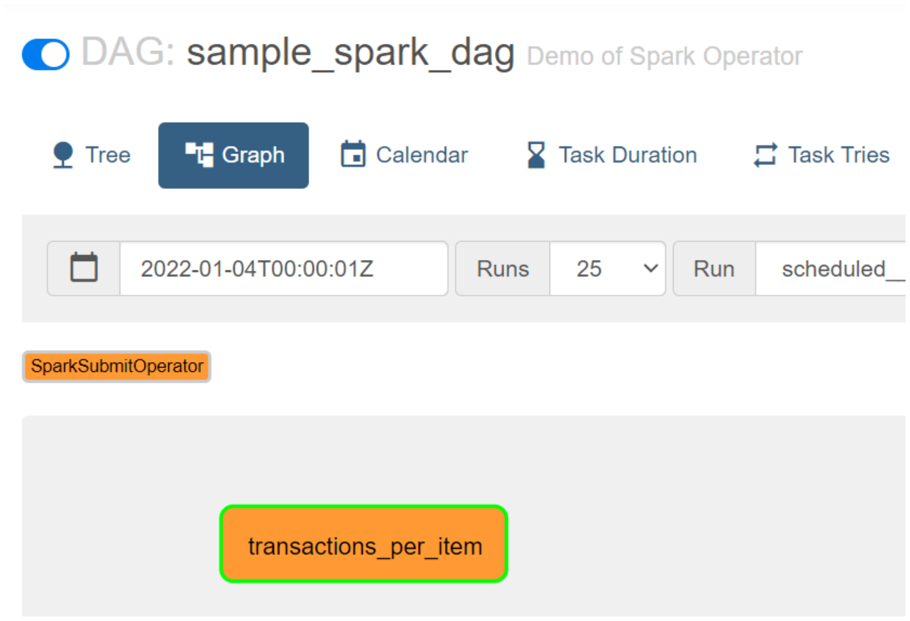
DAGs

DAG	Owner	Runs	Schedule	Last Run	Next Run
sample_bash_dag	admin		0 11 * * *		2022-01-04, 11:00:00
sample_hive_dag	admin		@once		2022-01-04, 00:00:00
sample_python_dag	admin		@daily		2022-02-17, 00:00:00
sample_spark_dag	admin		@once		2022-01-04, 00:00:00
sample_sqoop_dag example	admin		1 day, 0:00:00		2022-02-17, 22:12:31

(Note: The DAG might take a while to show up on the UI. Keep refreshing and wait patiently)

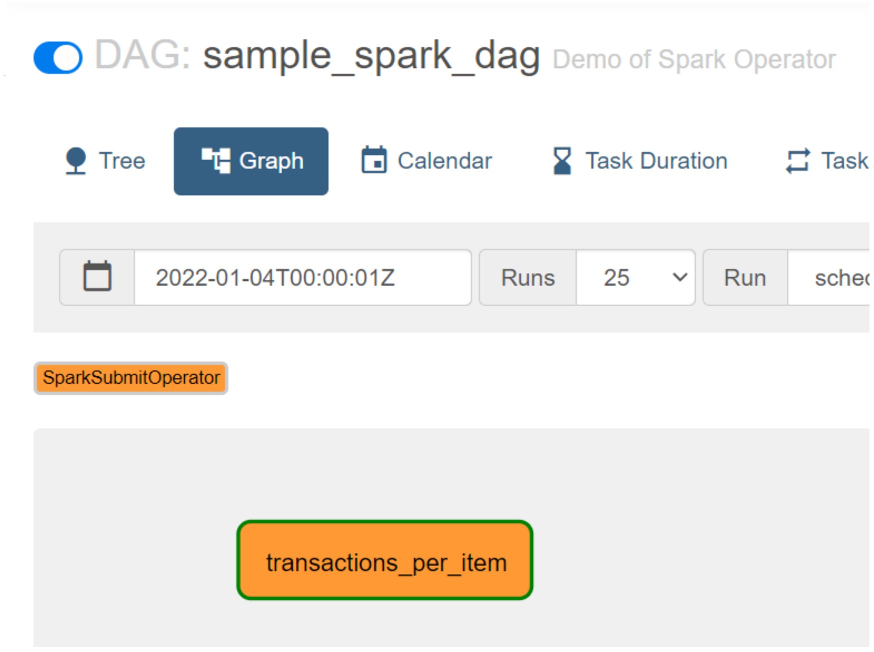
11. Click on the sample_spark_dag and go to the graph view

    You will see the task is running



DAG: sample_spark_dag Demo of Spark Operator

Click on refresh and eventually, it will have successfully completed



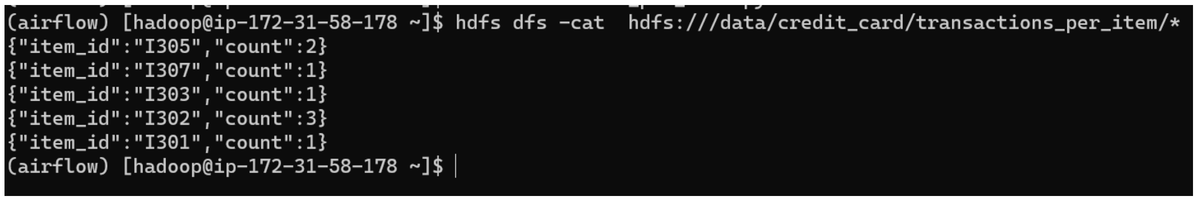
DAG: sample_spark_dag Demo of Spark Operator

12. Once the DAG has completed execution, the output will be generated in the following HDFS location:

    hdfs:///data/credit_card/transactions_per_item

13. You can view the results in the CLI by using the following command:

    hdfs dfs -cat hdfs:///data/credit_card/transactions_per_item/*



```
(airflow) [hadoop@ip-172-31-58-178 ~]$ hdfs dfs -cat  hdfs:///data/credit_card/transactions_per_item/*
{"item_id":"I305","count":2}
{"item_id":"I307","count":1}
{"item_id":"I303","count":1}
{"item_id":"I302","count":3}
{"item_id":"I301","count":1}
(airflow) [hadoop@ip-172-31-58-178 ~]$ 
```

14. You can switch off your DAG if you don't want it to run anymore.