

[Return to "Data Engineering Nanodegree" in the classroom](#)[DISCUSS ON STUDENT HUB](#)

Data Pipelines with Airflow

REVIEW

CODE REVIEW 6

HISTORY

[airflow/plugins/operators/stage_redshift.py](#) 3[airflow/plugins/operators/load_dimension.py](#) 1[airflow/plugins/operators/data_quality.py](#) 1

```
1 from airflow.hooks.postgres_hook import PostgresHook
2 from airflow.models import BaseOperator
3 from airflow.utils.decorators import apply_defaults
4
5 import logging
6
7 class DataQualityOperator(BaseOperator):
8
9     ui_color = '#89DA59'
10
11     @apply_defaults
12     def __init__(self,
13                  redshift_conn_id="",
14                  tables=[],
15                  *args, **kwargs):
16
17         super(DataQualityOperator, self).__init__(*args, **kwargs)
18         self.tables = tables
19         self.redshift_conn_id = redshift_conn_id
20
21     def execute(self, context):
22         redshift_hook = PostgresHook(self.redshift_conn_id)
23         for table in self.tables:
```

AWESOME

Nice work passing an array of checks and runs them in a loop. 🍷

```
24         self.log.info(f"Data Quality checking for {table} table")
25         records = redshift_hook.get_records(f"SELECT COUNT(*) FROM {table}")
26         if len(records) < 1 or len(records[0]) < 1:
27             raise ValueError(f"Data quality check failed. {table} returned no results")
28         num_records = records[0][0]
29         if num_records < 1:
30             raise ValueError(f"Data quality check failed. {table} contained 0 rows")
31         self.log.info(f"Data quality on table {table} check passed with {records[0][0]} records")
```

[airflow/dags/udac_example_dag.py](#) 1[airflow/plugins/operators/load_fact.py](#)[airflow/plugins/operators/_init_.py](#)[airflow/plugins/helpers/sql_queries.py](#)[airflow/plugins/helpers/_init_.py](#)[airflow/plugins/_init_.py](#)[airflow/create_tables.sql](#)[RETURN TO PATH](#)

Rate this review

