UDACITY

PROJECT SPECIFICATION

# Data Pipelines with Airflow

## General

| CRITERIA | MEETS SPECIFICATIONS |
| --- | --- |
| The dag and plugins do not give an error when imported to Airflow | DAG can be browsed without issues in the Airflow UI |
| All tasks have correct dependencies | The dag follows the data flow provided in the instructions, all the tasks have a dependency and DAG begins with a start_execution task and ends with a end_execution task. |

## Dag configuration

| CRITERIA | MEETS SPECIFICATIONS |
| --- | --- |
| Default_args object is | DAG contains default_args dict, with the following keys:<br><br>• Owner |

| CRITERIA | MEETS SPECIFICATIONS |
| --- | --- |
| used in the DAG | <ul><li>Depends_on_past</li><li>Start_date</li><li>Retries</li><li>Retry_delay</li><li>Catchup</li></ul> |
| Defaults_args are bind to the DAG | The DAG object has default args set |
| The DAG has a correct schedule | The DAG should be scheduled to run once an hour |

## Staging the data

| CRITERIA | MEETS SPECIFICATIONS |
| --- | --- |
| Task to stage JSON data is included in the DAG and uses the RedshiftStage operator | There is a task that to stages data from S3 to Redshift. (Runs a Redshift copy statement) |
| Task uses params | Instead of running a static SQL statement to stage the data, the task uses params to generate the copy statement dynamically |

| CRITERIA | MEETS SPECIFICATIONS |
|---|---|
| Logging used | The operator contains logging in different steps of the execution |
| The database connection is created by using a hook and a connection | The SQL statements are executed by using a Airflow hook |

## Loading dimensions and facts

| CRITERIA | MEETS SPECIFICATIONS |
|---|---|
| Set of tasks using the dimension load operator is in the DAG | Dimensions are loaded with on the LoadDimension operator |
| A task using the fact load operator is in the DAG | Facts are loaded with on the LoadFact operator |
| Both operators use params | Instead of running a static SQL statement to stage the data, the task uses params to generate the copy statement dynamically |

| CRITERIA | MEETS SPECIFICATIONS |
|---|---|
| | |
| The dimension task contains a param to allow switch between append and insert-delete functionality | The DAG allows to switch between append-only and delete-load functionality |

## Data Quality Checks

| CRITERIA | MEETS SPECIFICATIONS |
|---|---|
| A task using the data quality operator is in the DAG and at least one data quality check is done | Data quality check is done with correct operator |
| The operator raises an error if the check fails pass | The DAG either fails or retries n times |
| | |

| CRITERIA | MEETS SPECIFICATIONS |
| --- | --- |
| The operator is parametrized | Operator uses params to get the tests and the results, tests are not hard coded to the operator |

## Suggestions to Make Your Project Stand Out!

- Simple and dynamic operators, as little hard coding as possible
- Effective use of parameters in tasks
- Clean formatting of values in SQL strings
- Load dimensions with a subdag