**[Instructions] Overview of Glue Components**

As part of this module we will get a high level overview of different Glue Components. We will get into the details of each and every component as part of subsequent modules or sections.

* Glue Catalog
  + Glue Crawlers
  + Glue Databases and Tables
* Glue Jobs
* Glue Triggers
* Glue Workflows

Let us go ahead and build a simple pipeline using the flights dataset to get a quick overview about Glue Components.

* Create Crawler and Catalog table for flights data
* Validate using Athena
* Create Glue Job to convert file format from csv to parquet
* Run and Monitor the job
* Create Catalog table on top of new location with parquet file format
* Validate new table using Athena
* Clean up everything
* Build the workflow for the pipeline
  + Crawl the source folder for flights data to refresh the table
  + Run Glue Job to convert the file format
  + Crawl the target folder to refresh the table
* Run the workflow and monitor
* Validate both source and target tables using Athena

**[Instructions] Create Crawler and Catalog Table**

Let us define and run the crawler to create a catalog table for flights data set.

* Crawler Name: **Flights Data Crawler**
* Database Name: **flights-db**
* Table Name: **flightscsv**

**[Instructions] Analyze Data using Athena**

Let us ensure that we can query the data using a server less query engine called Athena.

* You might have to configure s3 bucket to get started with Athena to store query results.
* Choose appropriate database or use database prefix while running queries.
* Run below queries and ensure that data is copied successfully.

1. SELECT count(1)
2. FROM "flights-db".flightscsv;

[Instructions and Code] Creating S3 Bucket and Role

Here are the details related for creating the s3 bucket and role that are required for our Glue Job to convert file format.

Bucket Name: itv-flights

Policy Name: ITVFlightsS3FullPolicy

Role Name: ITVFlightsGlueRole

Here are the steps we have followed to create s3 Bucket as well as Role for the Glue Job to convert file format will work.

Create s3 Bucket

Create Policy with following Permissions.

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "ListObjectsInBucket",

"Effect": "Allow",

"Action": [

"s3:ListBucket"

],

"Resource": [

"arn:aws:s3:::itv-flights"

]

},

{

"Sid": "AllObjectActions",

"Effect": "Allow",

"Action": "s3:\*Object",

"Resource": [

"arn:aws:s3:::itv-flights/\*"

]

}

]

}

Create Role with a predefined policy AWSGlueServiceRole as well as the newly created custom policy.

**[Instructions] Create and Run the Glue Job**

Let us see how we can create and run our first Glue Job.

* We must have the role with appropriate permissions via policies before creating the Glue Job. In our case we have created it as **ITVFlightsGlueRole**
* We can go to Glue Console and click on Jobs. Enter the name as **flights\_csv\_to\_parquet**.
* Source Table: **flights-db.flightscsv**
* Target Connection type: **s3**
* Target Folder with in bucket: **s3://itv-flights/flightsparquet**
* Make sure to choose all monitoring options and then create the job. It will redirect to Script Editor.
* You just need to click on **Run Job** and monitor the progress.
* Once the job is completely run, we can go to the target folder and see the created files - **ITVFlightsGlueRole**

**[Instructions and Code] Validate using Glue CatalogTable and Athena**

As the job is successfully run, let us create Glue Catalog Table and run queries using Athena against the table.

* Create a Glue Crawler against the s3 URI - **s3://itv-flights/flightsparquet/**
* Make sure to choose **Update a policy in an IAM role** and select **AWSGlueServiceRole-Flights** under IAM section. This will take care updating the role with required permissions to read from **s3://itv-flights/flightsparquet/**
* Run Crawler to create the Glue Catalog Table. It will be created as **flightsparquet**.
* Once the table is created, we can go to Athena Console and run below queries using Query Editor. Make sure to compare against the queries ran earlier using **flightscsv**.

1. SELECT count(1)
2. FROM "flights-db".flightsparquet;
4. SELECT \*
5. FROM "flights-db".flightsparquet
6. LIMIT 10;

**[Instructions and Code] Create and Run Glue Trigger**

Let us create and run the job to convert the file format using Glue Trigger.

* Glue Triggers can be used to schedule Glue Jobs. Here are the steps we are going to follow.
  + Delete **flightsparquet** folder in s3.
  + Create and Start the trigger.
  + Validate whether **flightsparquet** is created or not under s3.
  + Run queries against the Glue Catalog Table **flightsparquet** using Athena.
* We will name it as **flights\_csv\_to\_parquet\_trigger**. You can also name it using plain English **Flights CSV to Parquet Trigger**.
* We can associate the job with the trigger. Once the trigger is created we can select and then click on **Start Trigger** using **Actions Drop Down Menu**.
* You can monitor the status by going to the underlying Glue Job.
* Once it is succeeded you can go to Athena and validate by running the same queries ran earlier.

1. SELECT count(1)
2. FROM "flights-db".flightsparquet;
4. SELECT \*
5. FROM "flights-db".flightsparquet
6. LIMIT 10;

**[Instructions] Create Glue Workflow**

Let us create the Glue Workflow to create a catalog table for source data, run Glue Job and then create a catalog table for target data. Here are the steps to create glue workflow before running it.

* Create Workflow with Name - **Flights CSV to Parquet Workflow**
* Create Trigger to run crawler to create catalog table **flightscsv** - **Run Glue Crawler for Flights CSV Trigger**
* Attach Trigger with appropriate crawler - **Flights Data Crawler**
* Create Trigger to run Glue Job to convert file format from CSV to Parquet - **Run flights\_csv\_to\_parquet Job**
* Attach Trigger with appropriate Glue Job - **flights\_csv\_to\_parquet**
* Create Trigger to run crawler to create catalog table **flightsparquet** - **Run Glue Crawler for Flights Parquet**
* Attach Trigger with appropriate crawler - **Flights Parquet Data Crawler**
* Drop existing tables and delete the target folder so that we can validate the complete Workflow after running it.

**[Instructions] Run Glue Workflow and Validate**

As the Glue Workflow is created with triggers and associated crawlers/jobs, it is time for us to run and validate whether Glue Workflow is successful or not.

* Go to Workflows and start the Workflow using the Actions Drop Down.
* It will trigger the pipeline and you can monitor by going to **History** and select the latest run.
* Once the Glue Workflow is completely run, make sure to validate.
  + Check for the tables in Glue Catalog in **flights-db** database.
  + Also, run queries against the tables using Athena.

1. SELECT count(1)
2. FROM "flights-db".flightscsv;
4. SELECT count(1)
5. FROM "flights-db".flightsparquet;
7. SELECT \*
8. FROM "flights-db".flightsparquet
9. LIMIT 10;