# HUDI Workshop

Apache Hudi is a storage abstraction framework that helps distributed organizations build and manage petabyte-scale data lakes. Using primitives such as upserts and incremental pulls, Hudi brings stream style processing to batch-like big data.

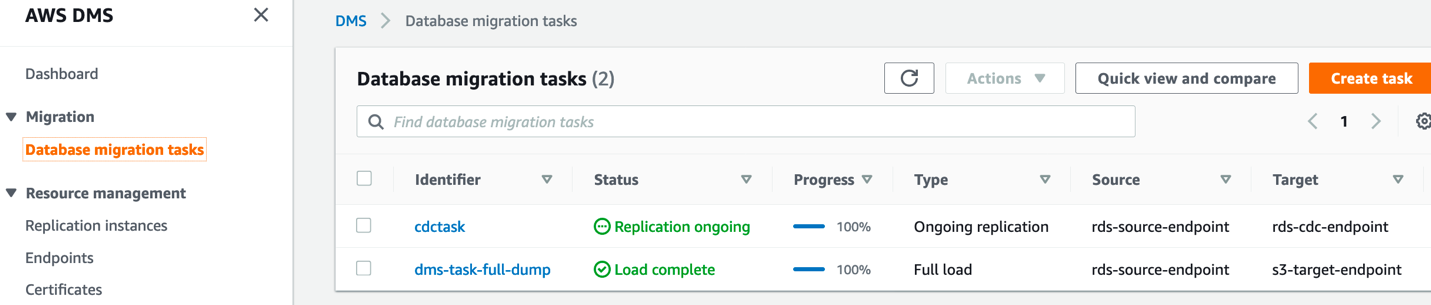
Hudi enables Atomicity, Consistency, Isolation & Durability (ACID) semantics on a data lake.

Here are some good references to read:

1. [Apache Hudi concepts](https://hudi.apache.org/concepts.html)
2. [How Hudi Works](https://docs.aws.amazon.com/emr/latest/ReleaseGuide/emr-hudi-how-it-works.html)

**Step 1: Complete the Prerequisites**

1. Ensure [**Instructor Prelab (not requried in AWS event)**](https://catalog.us-east-1.prod.workshops.aws/v2/workshops/976050cc-0606-4b23-b49f-ca7b8ac4b153/en-US/400/401/410-pre-lab-1) is completed and the source RDS database is fully populated.
2. Ensure Autocomplete DMS Lab is completed and the cdctask status is ‘Replication ongoing’ as shown in the below diagram:



**HUDI Lab:**

In this lab you will learn how to create HUDI tables and run incremental queries in a Glue job.

Here are the steps performed in the order:

Step 1 – Activate Glue Connector from market place

Step 2 – Create glue job and create HUDI table

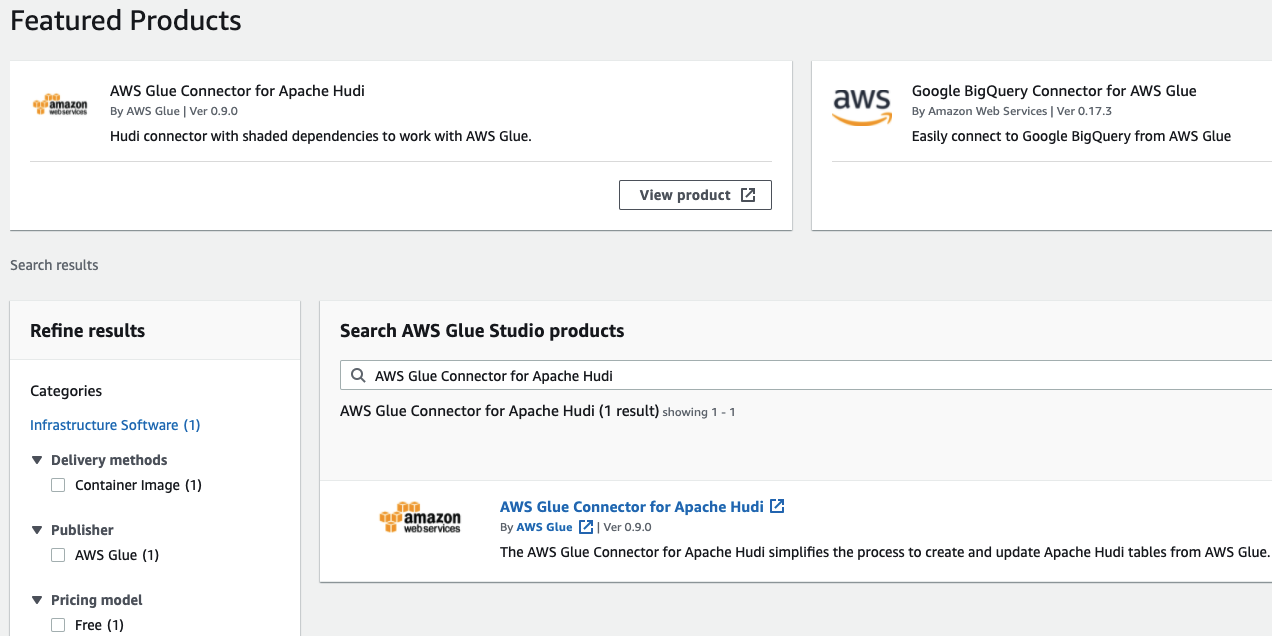
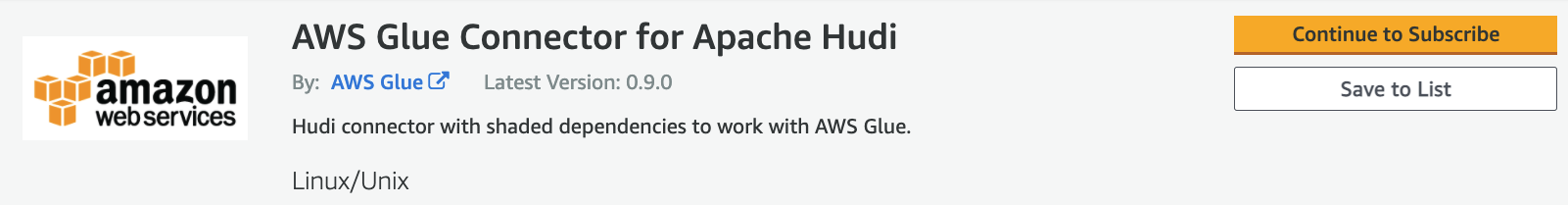
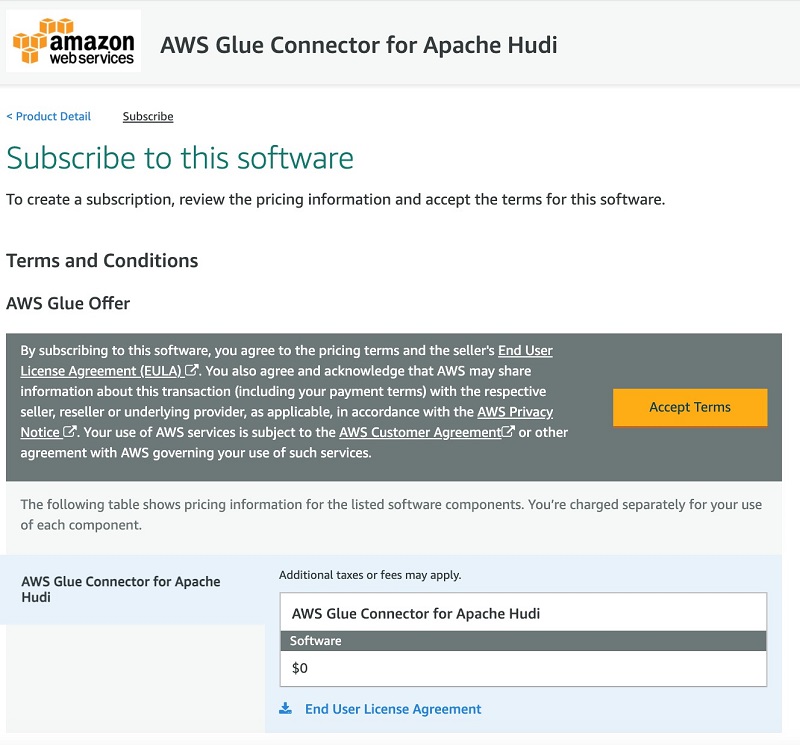
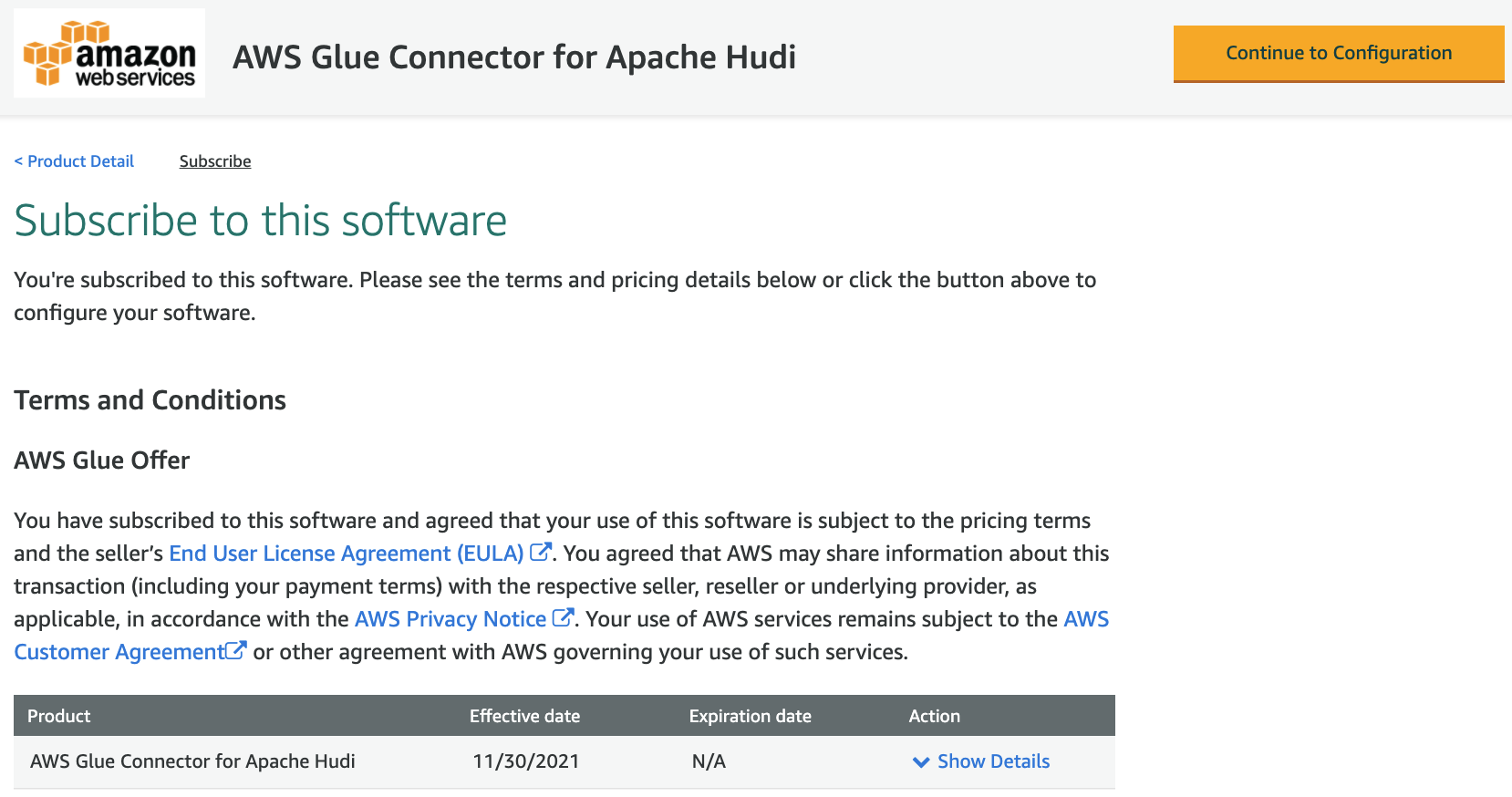
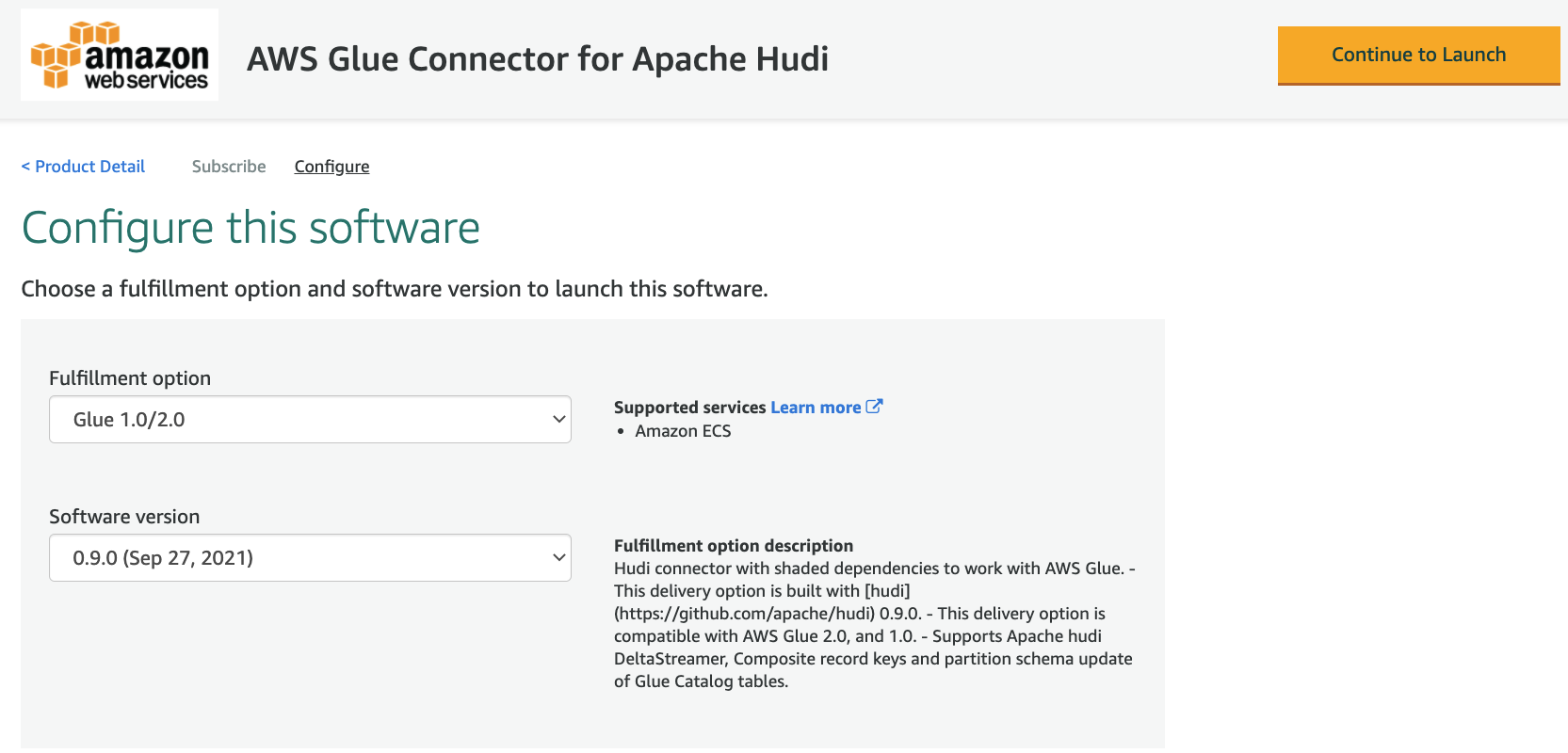
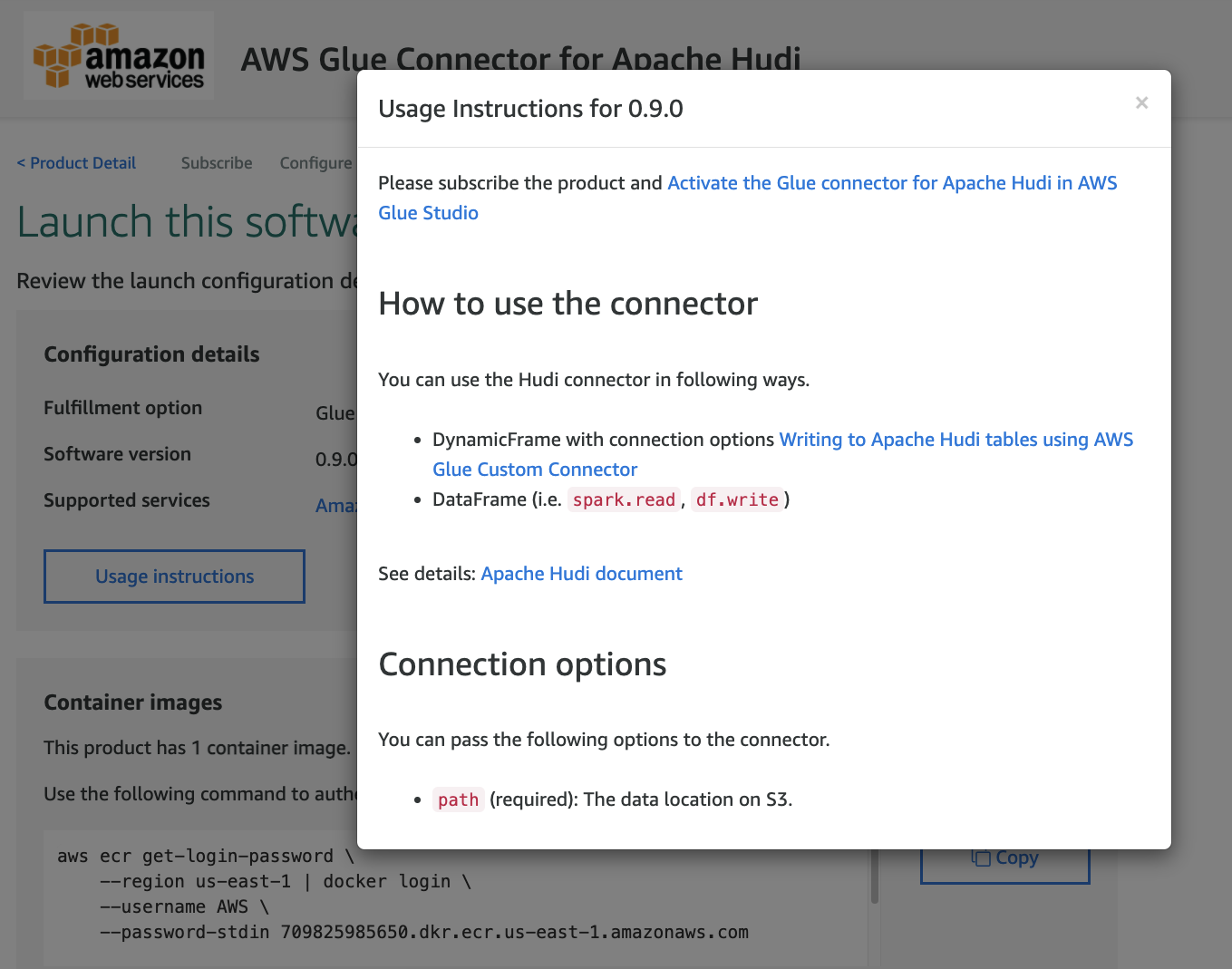
Step 3 – Query the HUDI table in Athena

Step 4 – Understand HUDI configurations

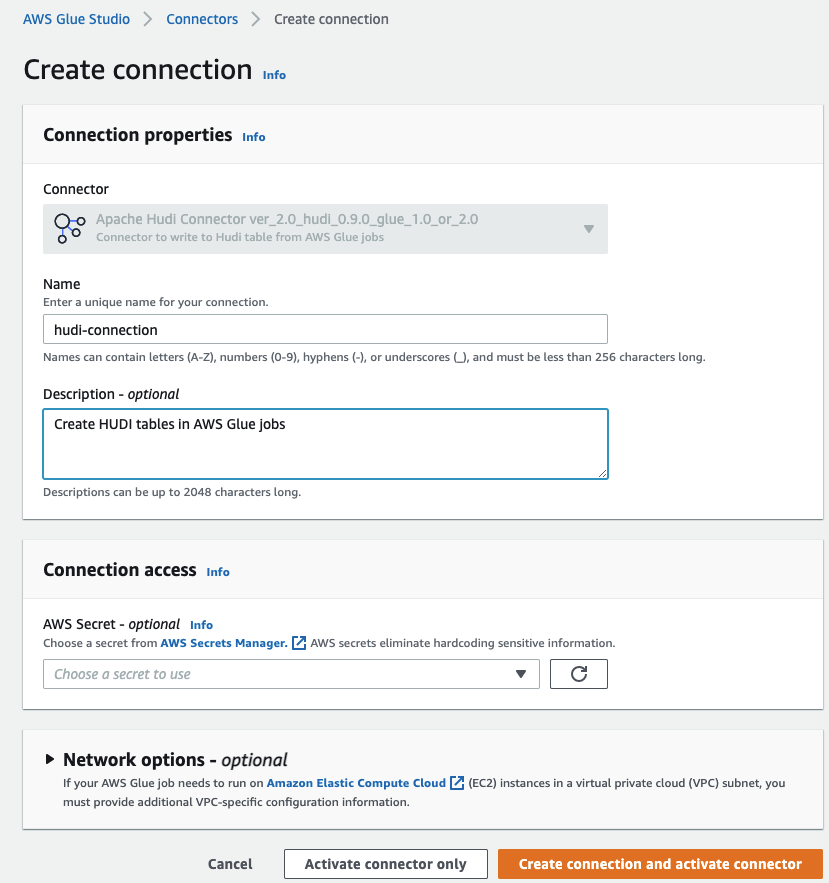
Step 5 – Run Incremental Queries using Spark SQL

**Step 1 - Activate AWS Glue HUDI Connector from the market place**

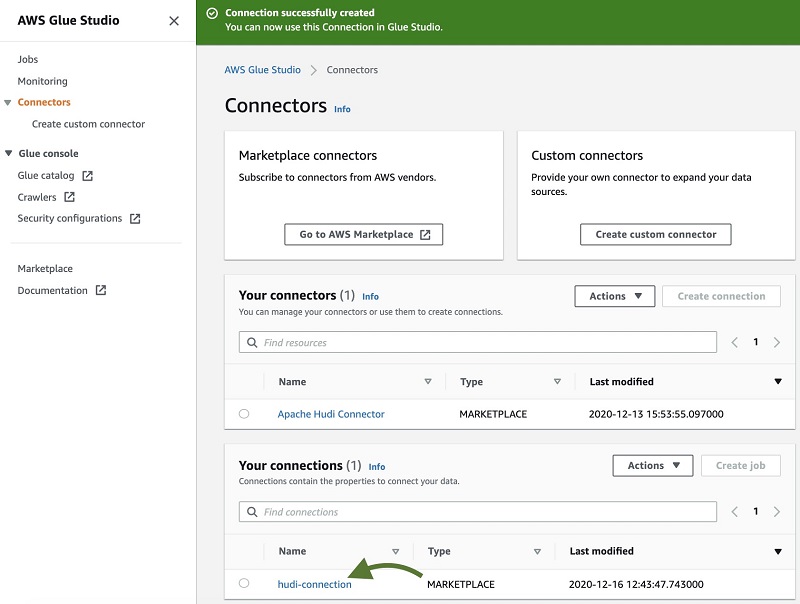
To create your AWS Glue job with an AWS Glue Custom Connector, complete the following steps:

1. Go to the [AWS Glue Studio Console](https://console.aws.amazon.com/gluestudio/home#/marketplace), search for AWS Glue Connector for Apache Hudi and choose AWS Glue Connector for Apache Hudi link.  
   
2. Choose Continue to Subscribe.  
   
3. Review the Terms and Conditions and choose the Accept Terms button to continue.
4. Make sure that the subscription is complete and you see the Effective date populated next to the product and then choose Continue to Configuration button.  
   
5. As of this writing, 0.9.0 is the latest version of the AWS Glue Connector for Apache Hudi. Make sure that *0.9.0 (Sep 27, 2021)* is selected in the Software Version dropdown and *Glue 1.0/2.0* is selected in the Fulfillment option dropdown. Choose Continue to Launch button.  
   
6. Under Launch this software, choose Usage Instructions and then choose Activate the Glue connector for Apache Hudi in AWS Glue Studio.  
   

You’re redirected to AWS Glue Studio.

1. For Name, enter a name for your connection (for example, hudi-connection).
2. For Description, enter a description.  
   
3. Click Create connection and activate connector.

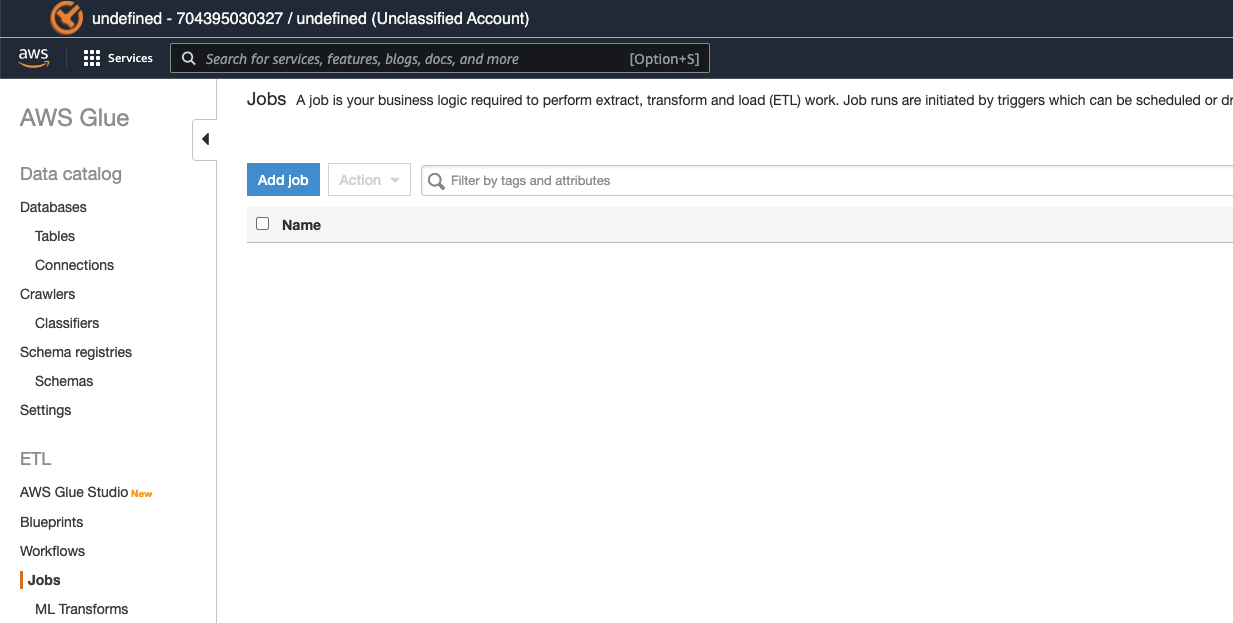
A message appears that the connection was successfully created, and the connection is now visible on the AWS Glue Studio console.



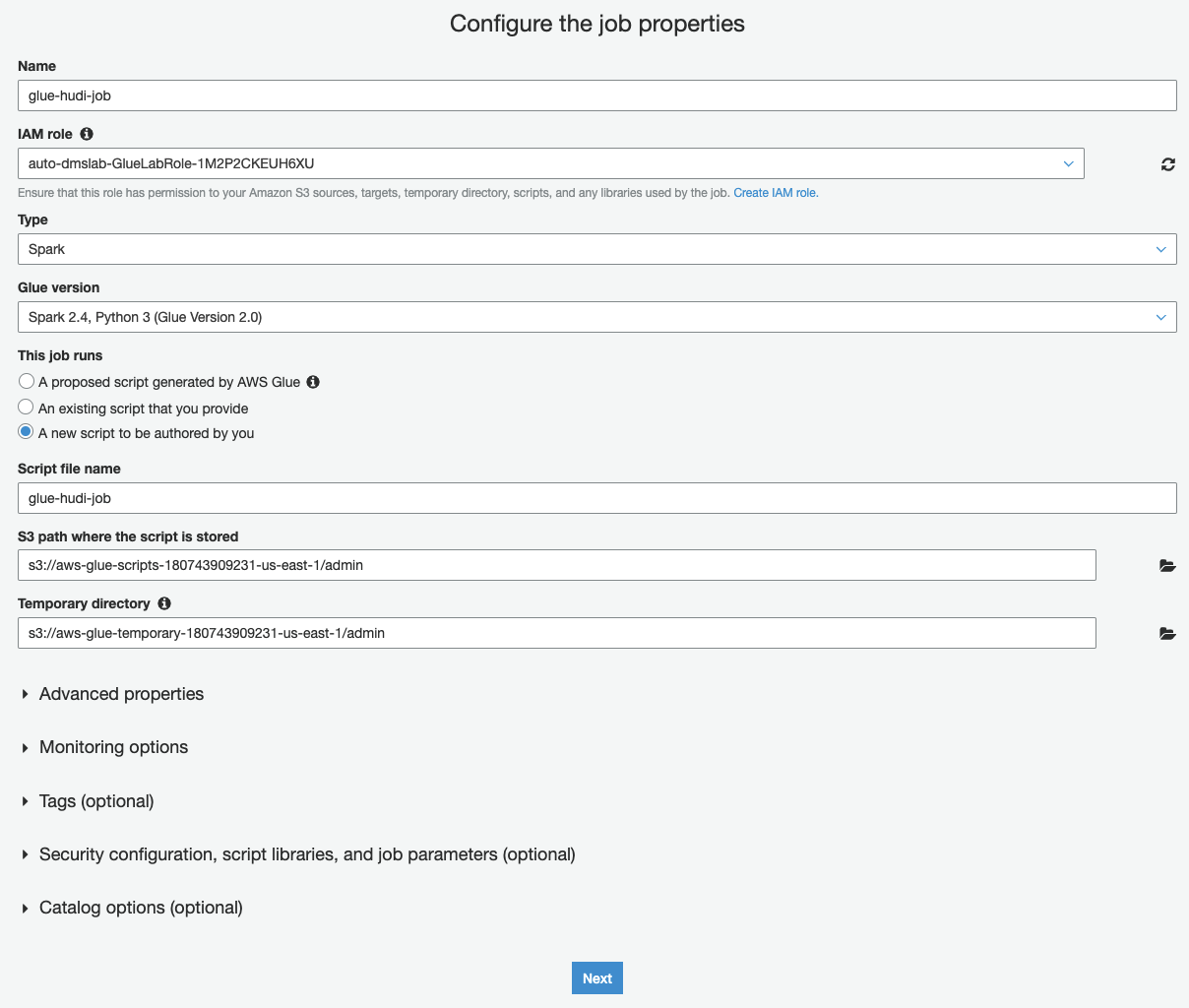
**Step 2 – Create glue job and create HUDI table**

Let’s create a glue job to create a HUDI table for the source table ticket\_purchase\_hist

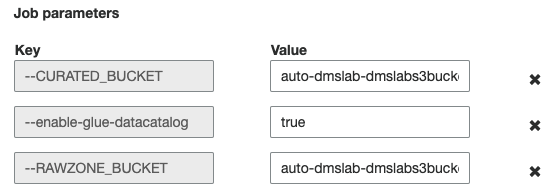
1. Go to [AWS Glue Console](https://console.aws.amazon.com/glue) and click ***Jobs*** in the left navigation pane to create a new job.



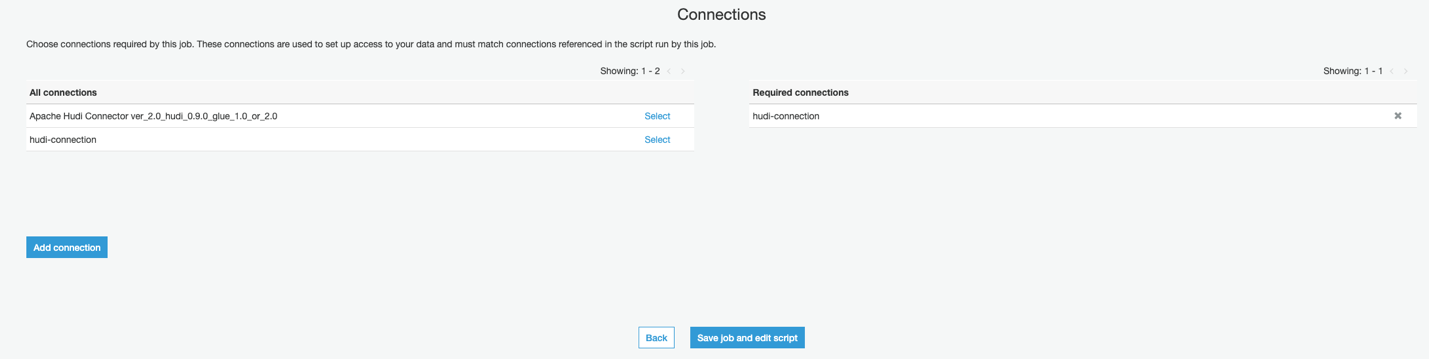
1. Click ***Add Job***, and enter **glue-hudi-job** as the job name.
2. For IAM role, select a role that matches the name like *<stackname>-GlueLabRole-<RandomString>.*
3. Leave default selection for Type.
4. Select Glue version that’s compatible with HUDI Connector you have activated from the marketplace.
5. For This job runs, choose ‘A new script to be authored by you’.



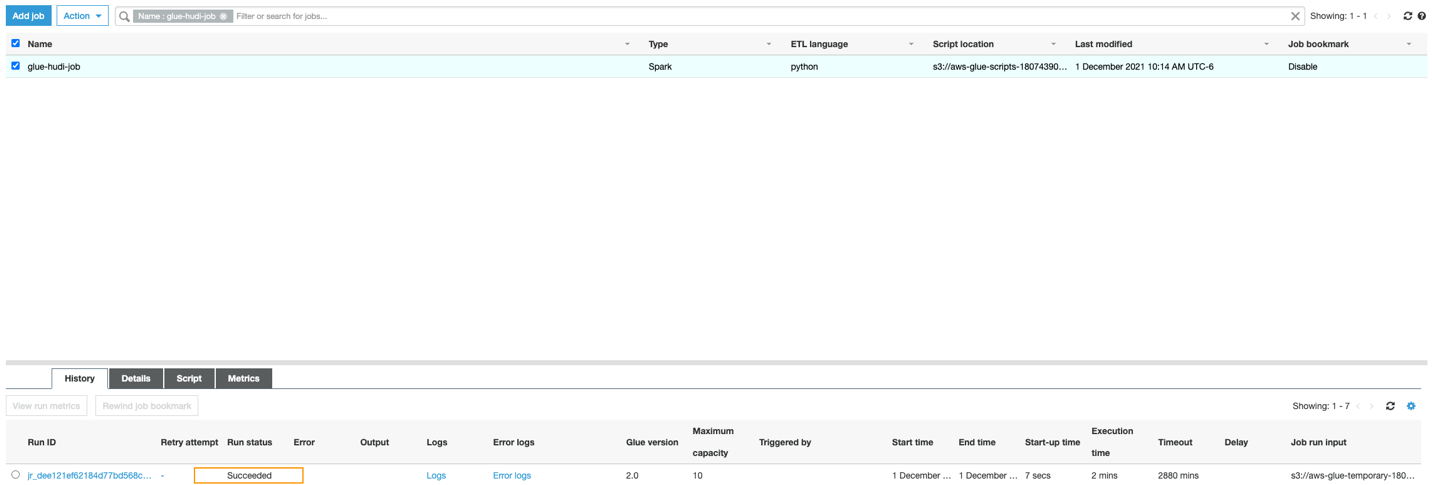
1. Make a note of S3 bucket name from DMS endpoint called s3-target-endpoint, it should look like *auto-dmslab-dmslabs3bucket-<RandomString>*
2. Expand **Security configuration, script libraries, and job parameters**, look for **Job parameters** option and enter following key values
   1. Key: *--enable-glue-datacatalog*, Value: *true*
   2. Key: *--CURATED\_BUCKET*, Value: *Bucket name to hold your HUDI tables*
   3. Key: *--RAWZONE\_BUCKET*, Value: *Bucket name from s3-target-endpoint*



1. Leave the default options for all the remaining fields and click Next
2. Click *Select* next to hudi-connection to add the connection to your Glue job and click *Save job and edit script*

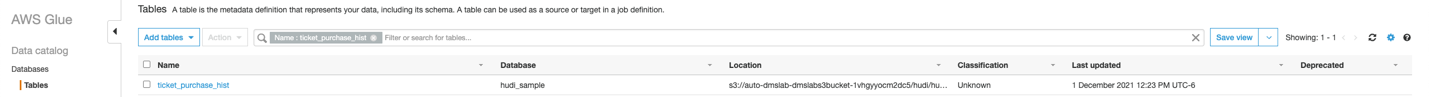


1. Copy the script from <<this location>> and paste it in the Glue script editor
2. Save your changes by clicking Save and click Run job
3. Wait for job to complete and Run status shows *Succeeded*

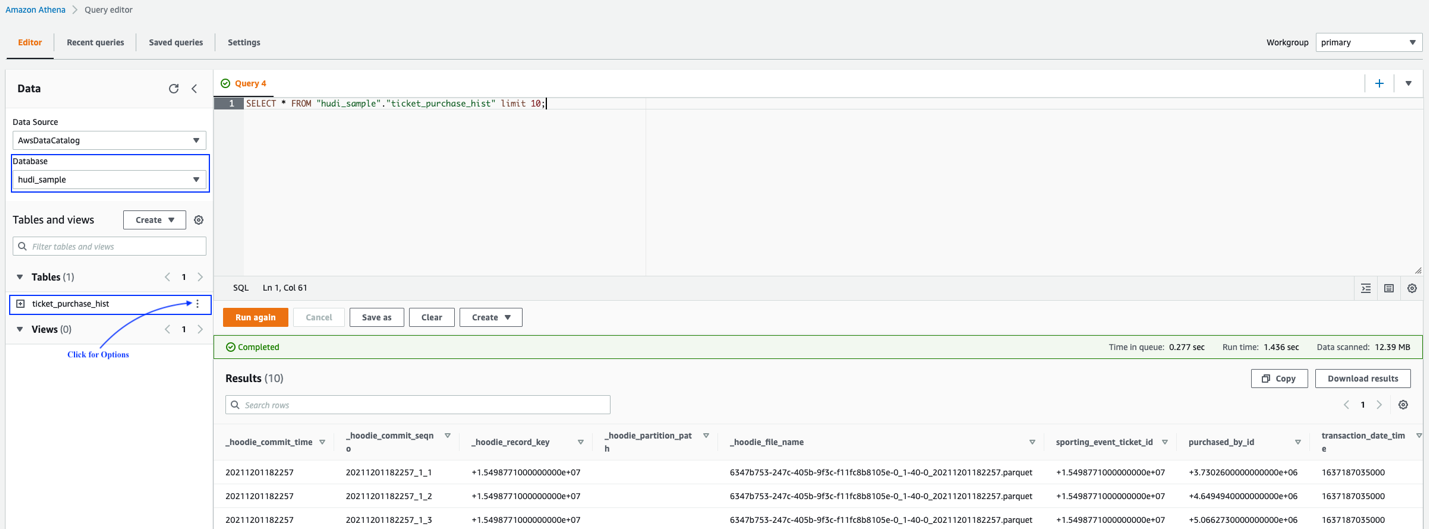


**Step 3 – Query the HUDI table in Athena**

1. Let’s ensure the HUDI table got created successfully by going to [AWS Glue Console](https://console.aws.amazon.com/glue), and click on *Tables* in the left navigation pane
2. You should see a new table called *ticket\_purchase\_hist* and database as *hudi\_sample*

****

1. Go to [Amazon Athena Console](https://console.aws.amazon.com/athena) and choose *hudi\_sample* for Database in the left navigation menu.
2. Click ellipsis and choose *Preview table.*
3. You should see HUDI table results as in the picture below



**Step 4 – Understand HUDI configurations**

HUDI offers different ways to configure your job to write/read HUDI tables. At a high level, you can control behavior at few levels.

Some of the notable configurations are:

* 1. *hoodie.datasource.write.storage.type – Choose* COPY\_ON\_WRITE or MERGE\_ON\_READ as your HUDI table type
  2. *hoodie.datasource.write.recordkey.field –* Similar to primary key of relational database table.
  3. *hoodie.datasource.hive\_sync.partition\_fields -* Field in the table to use for determining hive partition columns.
  4. *hoodie.datasource.write.operation –* Choose **upsert**, **insert** or **bulkinsert** for the write operation. Use bulkinsert to load new data into a table, and there on use upsert/insert. bulk insert uses a disk based write path to scale to load large inputs without need to cache it.
  5. *hoodie.datasource.read.end.instanttime -* Instant time to limit incrementally fetched data to. New data written with an instant\_time <= END\_INSTANTTIME are fetched out.

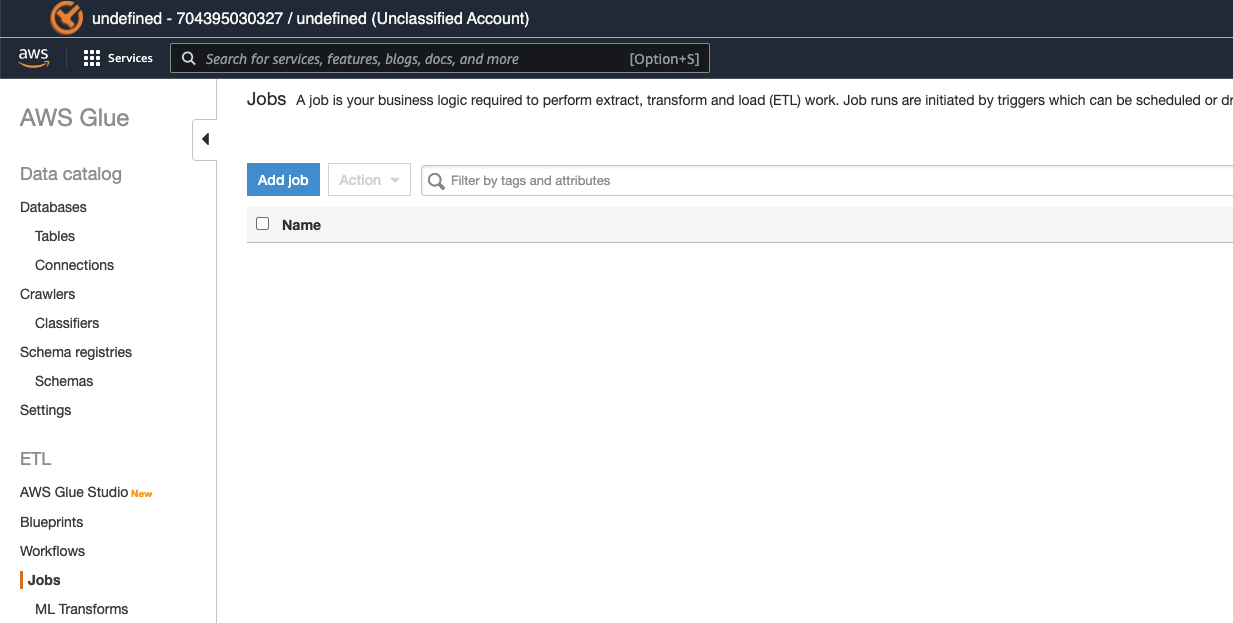
Learn more about all available configurations at [Apache Hudi documentation](https://hudi.apache.org/docs/configurations).

**Step 5 – Run Incremental Queries using Spark SQL**

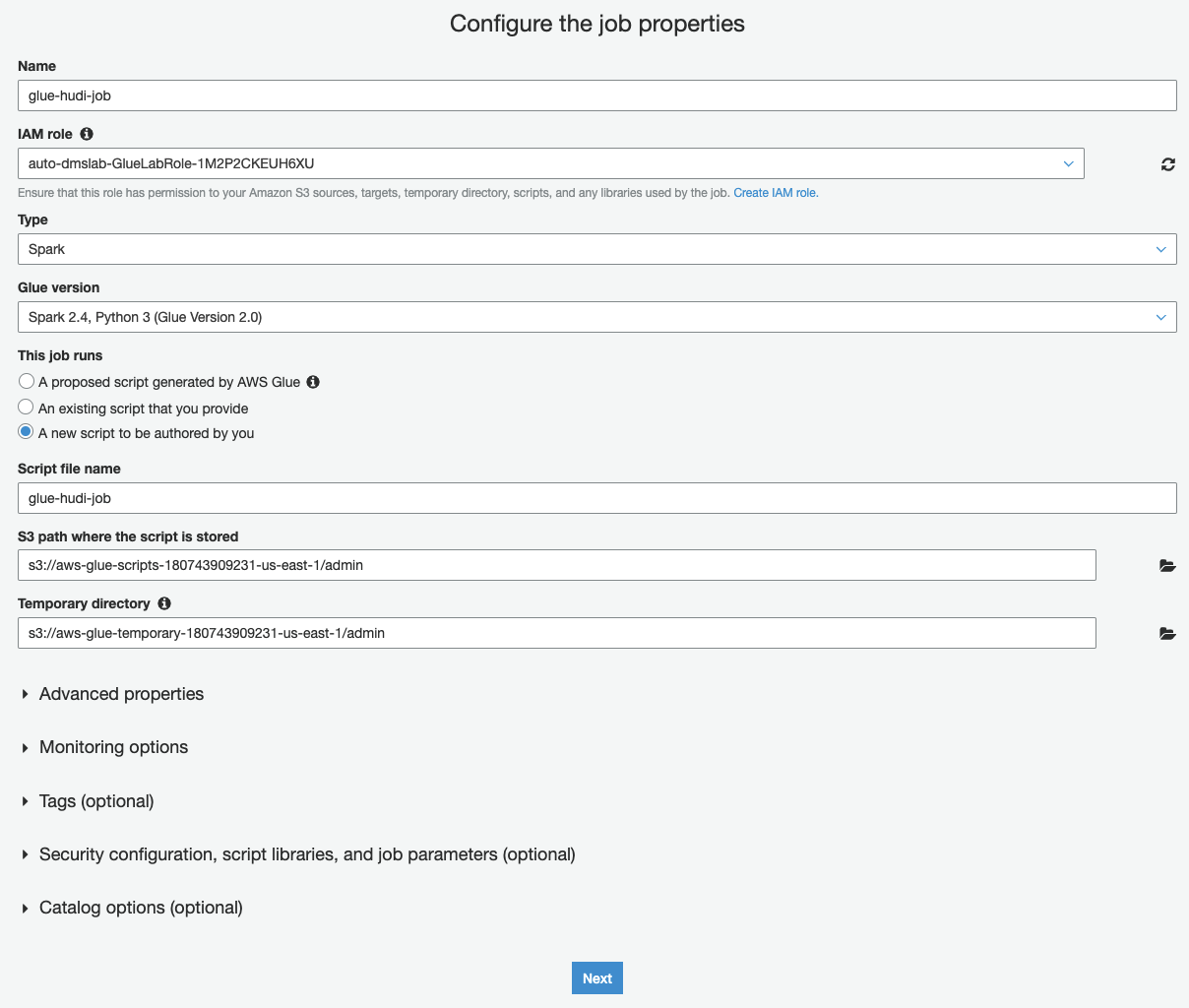
With HUDI, you can run incremental queries to retrieve data *as of* given date within Glue jobs.

Let’s create a glue job to run retrieve just previous day’s rows from the HUDI table *ticket\_purchase\_hist*

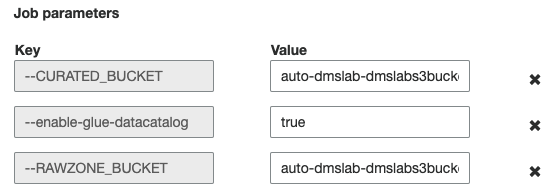
1. Go to [AWS Glue Console](https://console.aws.amazon.com/glue) and click ***Jobs*** in the left navigation pane to create a new job.



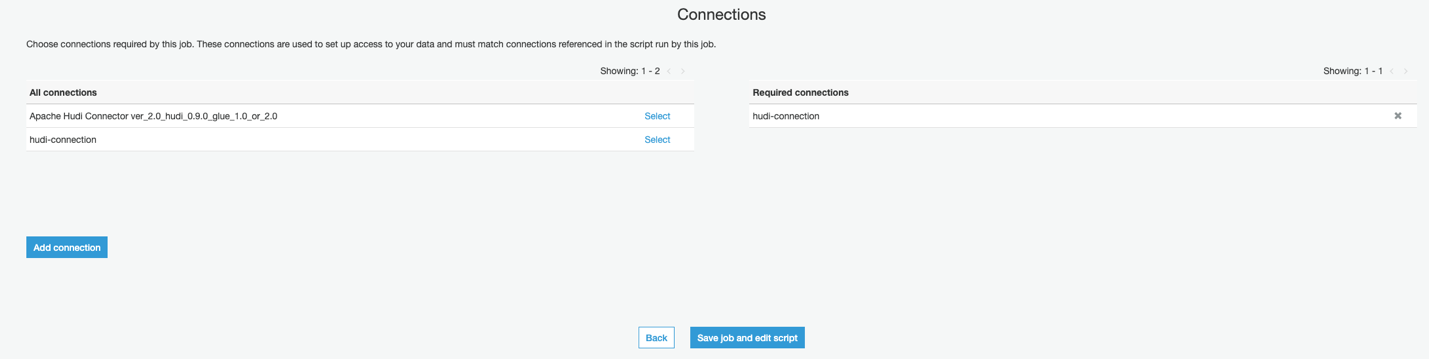
1. Click ***Add Job***, and enter **glue-hudi-job** as the job name.
2. For IAM role, select a role that matches the name like *<stackname>-GlueLabRole-<RandomString>.*
3. Leave default selection for Type.
4. Select Glue version that’s compatible with HUDI Connector you have activated from the marketplace.
5. For This job runs, choose ‘A new script to be authored by you’.



1. Make a note of S3 bucket name from DMS endpoint called s3-target-endpoint, it should look like *auto-dmslab-dmslabs3bucket-<RandomString>*
2. Expand **Security configuration, script libraries, and job parameters**, look for **Job parameters** option and enter following key values
   1. Key: *--enable-glue-datacatalog*, Value: *true*
   2. Key: *--CURATED\_BUCKET*, Value: *Bucket name to hold your HUDI tables*

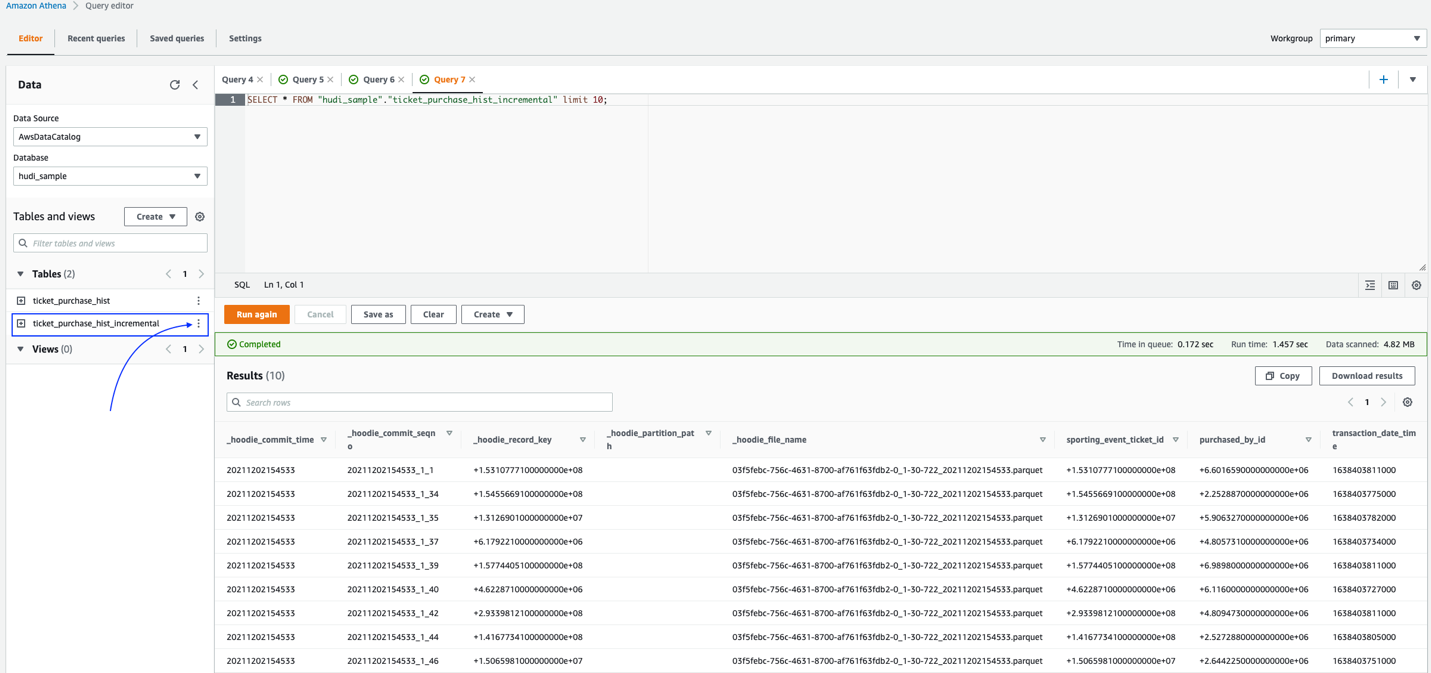


1. Leave the default options for all the remaining fields and click Next
2. Click *Select* next to hudi-connection to add the connection to your Glue job and click *Save job and edit script*



1. Copy the script from <<this location>> and paste it in the Glue script editor
2. Save your changes by clicking Save and click Run job
3. Wait for job to complete and Run status shows *Succeeded*

Go to Amazon Athena console and query the table *ticket\_purchase\_hist\_incremental* and you should have only the incremental changes from previous CDC update.



**Conclusion:**

You have created following resources in this workshop:

1. Activated HUDI connector for Glue from the AWS Marketplace.
2. Glue job to create a HUDI table corresponding to the source table *ticket\_purchase\_hist* performing upserts on the CDC changes.
3. Another glue job to retrieve incremental changes and store the results in a separate table *ticket\_purchase\_hist\_incremental.*
4. Ran queries in Amazon Athena against HUDI tables