**Lab 1: Setup Glue DataCatalog**

**Pre-requisites:**

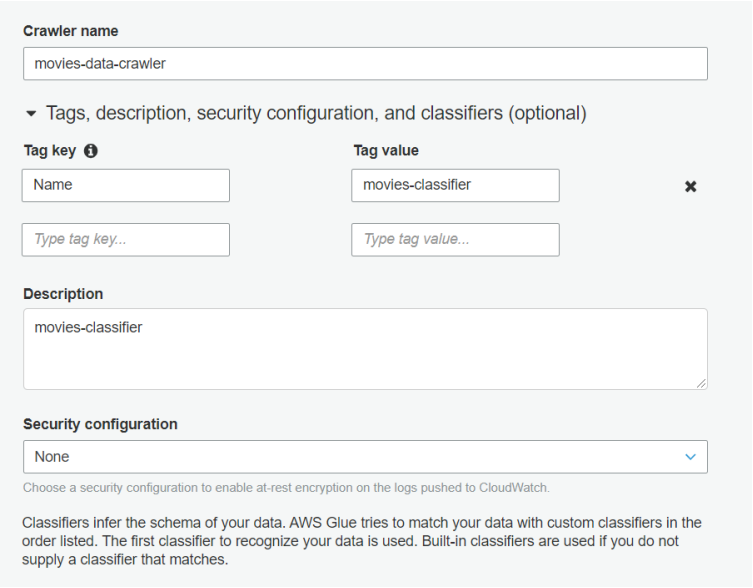
1. Data setup in S3
2. We will do this exercise in us-east-1 region.

**TO DO:**

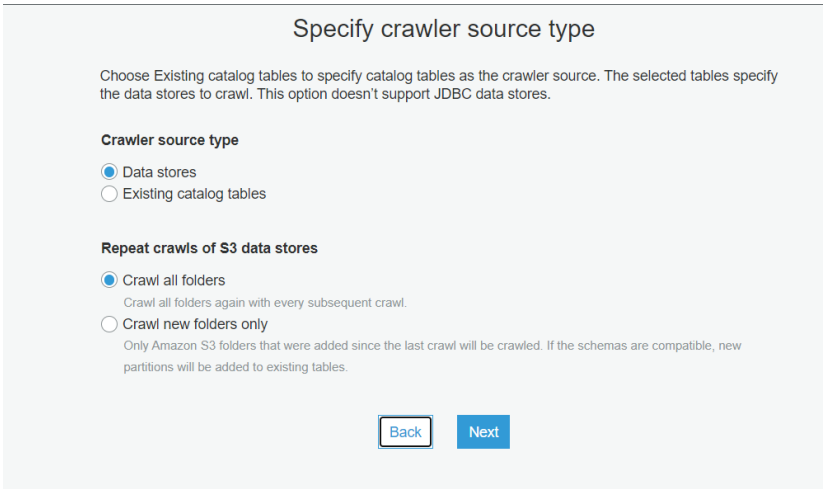
1. Setup a crawler that crawls over the data in S3 which would then setup a table for us in AWS Glue Data Catalog

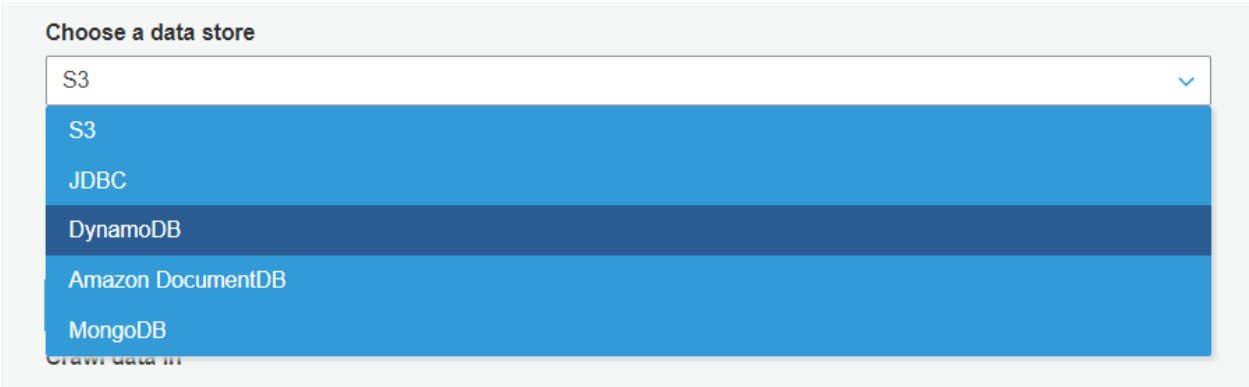
**Steps:**

1. Create a bucket ***moviestabledata*** in S3 and Upload **movies.json** file.
2. Navigate to Glue
3. Click on Crawlers. Click on Add Crawler.
    a. Give Crawler a name *movies-data-crawler.* Expand Tags,Description,… and go to the bottom and add movies-classifier. Click **Next**

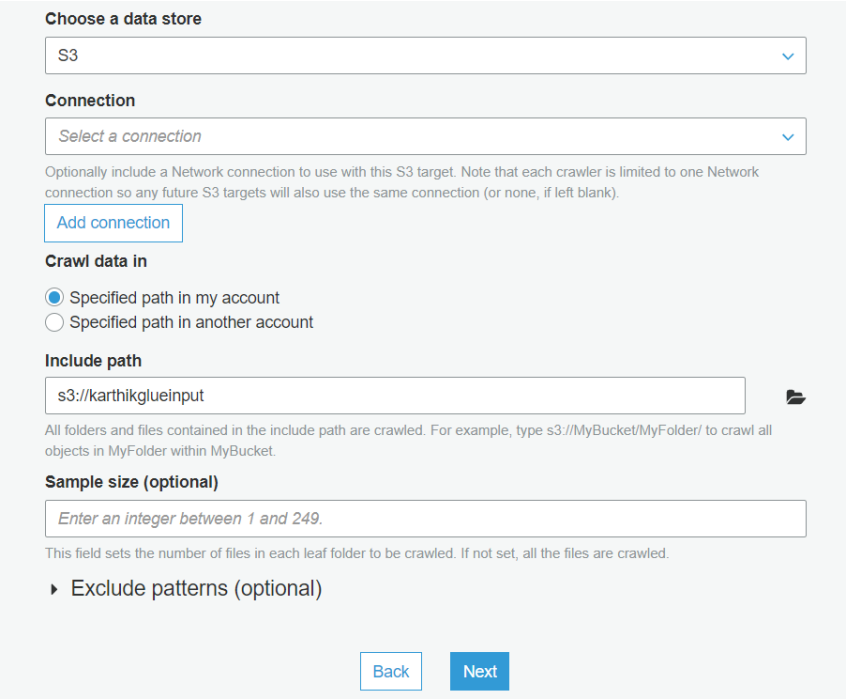



    b. Select the **Data Store** that we want this crawler to crawl. Click Next.
    We have various options as indicated below. We can select JDB Connection, DynamoDB or Amazon Document DB which is a MongoDB managed service that AWS provides and we can also crawl a Mongo DB database.

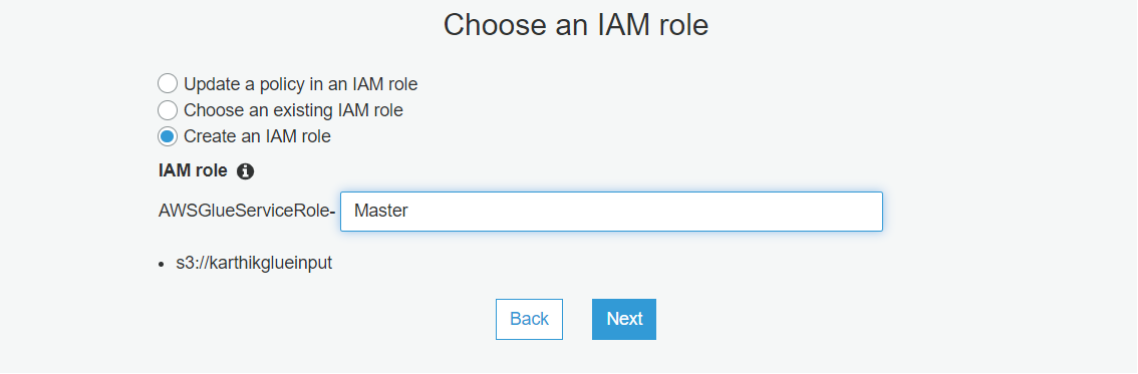


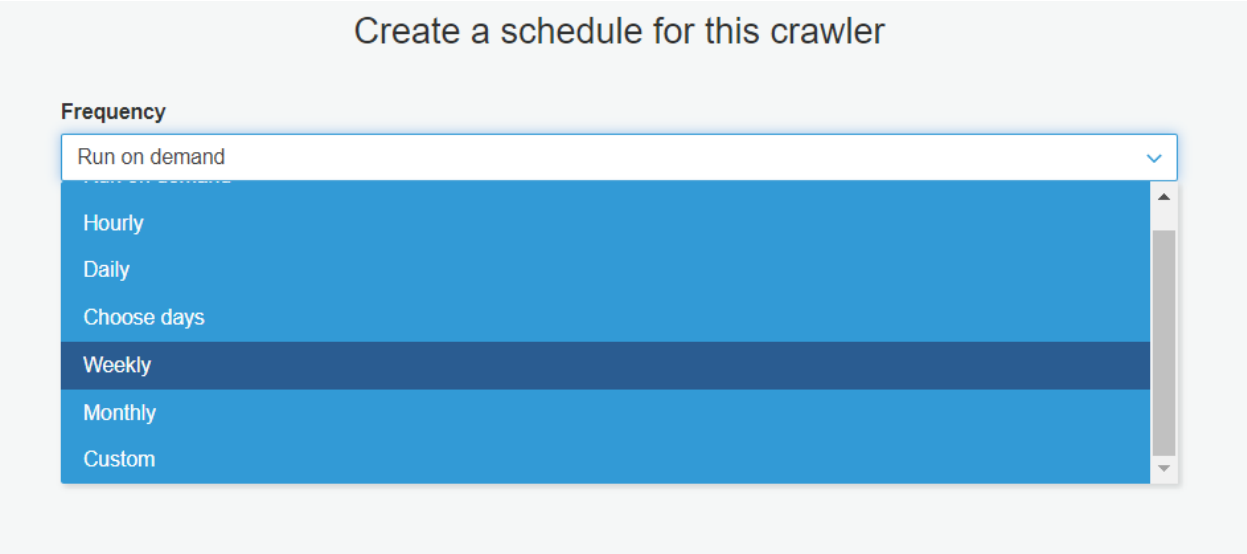

c. Select Data Store as S3. Select Crawl data in "Specified path in my account".



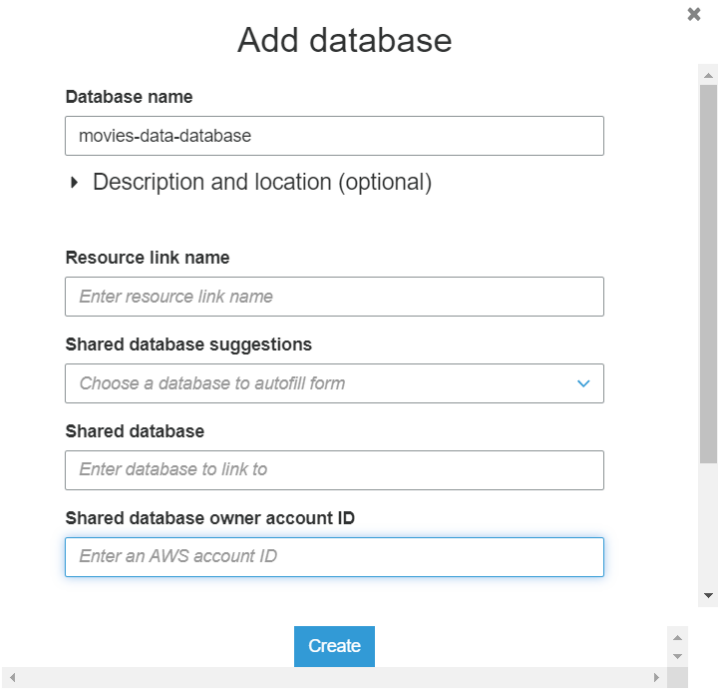
d. We can also exclude certain folders from crawling, expand “Exclude Patterns”. Just observe, we are not excluding anything for this lab exercise. Click Next.
e. We can add more Data Stores. For this lab exercise its not needed. Select “No”. Click Next.
f. We need to create movies-role - IAM Role. This is needed as we need to give read access to our bucket. Give the name as “Master” and Click on “Next”.




g. Setup a schedule to specify when this crawler is going to run. We can setup hourly, weekly, monthly or choose days or you can setup a custom cron expression. For this lab exercise, select “**Run on Demand”.** And click next



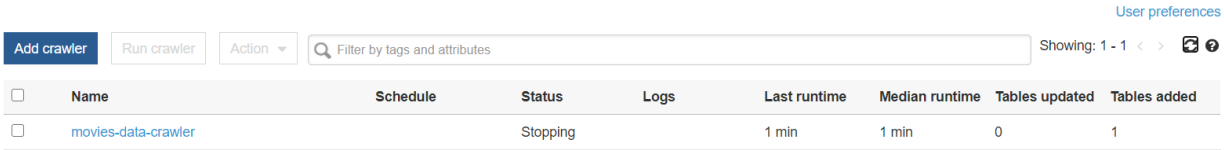

h. Setup a Database, this is  where our tables will live. Click on Add Database. Give the name as movies-data-database. Click on Create. All the tables will now be created in this database within AWS Glue. Click Next.




i. Review all the settings and click Finish.
j. Now our AWS Glue crawler *job-data-crawler* is ready.

4. Select the *movies-data-crawler* and click on Run Crawler.

   A crawler connects to a data store, progresses through a prioritized list of classifiers to determine the schema for your data, and then creates metadata tables in your data catalog.




   You can see that crawler has finished running and it has added 1 table and it took runtime of 1 min.

5. Click on Databases. Select *movies-data-database* and click on it. Now, click on " Tables in movies-data-database". Click on table **karthikglueinput** and have a look at details highlighted in yellow.

| | |
|---|---|
| **Classification** | json |
| **Location** | s3://karthikglueinput/ |
| **Connection** | |
| **Deprecated** | No |
| **Last updated** | Fri Jul 30 14:50:42 GMT+530 2021 |
| **Input format** | org.apache.hadoop.mapred.TextInputFormat |
| **Output format** | org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat |
| **Serde serialization lib** | org.openx.data.jsonserde.JsonSerDe |
| **Serde parameters** | paths **cast,genres,title,year** |
| **Table properties** | sizeKey **10118** objectCount **1** UPDATED_BY_CRAWLER **movies-data-crawler** CrawlerSchemaSerializerVersion **1.0** recordCount **126** averageRecordSize **80** CrawlerSchemaDeserializerVersion **1.0** compressionType **none** typeOfData **file** |

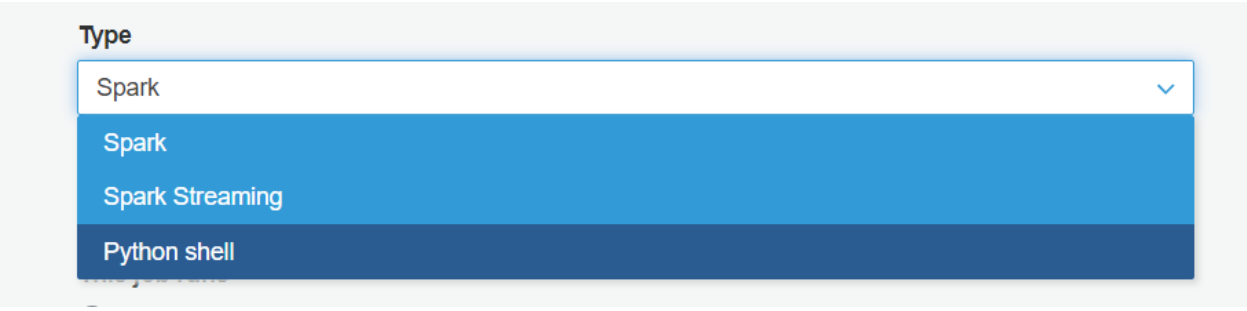
Schema

Showing: 1 - 4 of 4

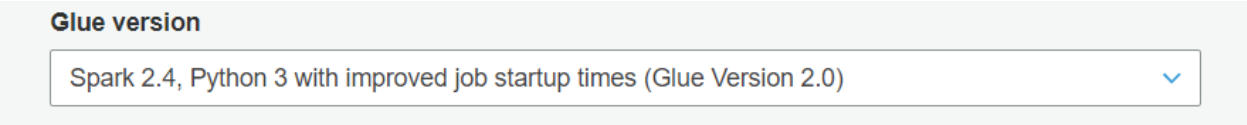
| | Column name | Data type | Partition key | Comment |
|---|---|---|---|---|
| 1 | title | string | | |
| 2 | year | int | | |
| 3 | cast | array | | |
| 4 | genres | array | | |

**Lab2 : Setup Glue Job**

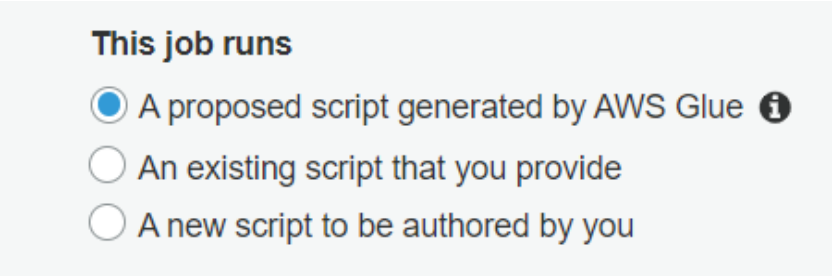
1. Go to AWS Glue.
2. We have already setup Data Catalog from our previous lab. It is data related to movies from Wikipedia.
3. Click on the table and have a look at the schema.
4. Click on Jobs under ETL category. Click on Add Job.
5. Give the job name – *movies-data-job*
6. Select IAM Role that we created in previous lab – AWSGlueServiceRole-MasterKarthik
7. Select the processing environment, we can select Spark out of the below mentioned options.



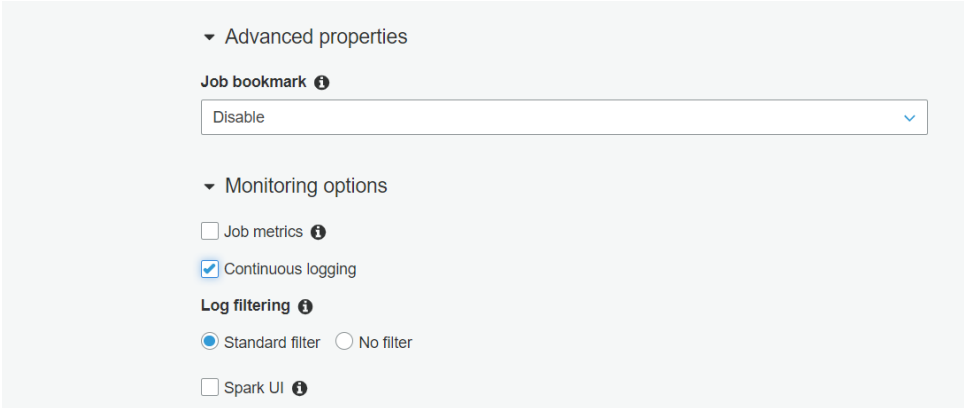

8. Select appropriate Glue Version. For this exercise, lets go ahead with below mentioned version:



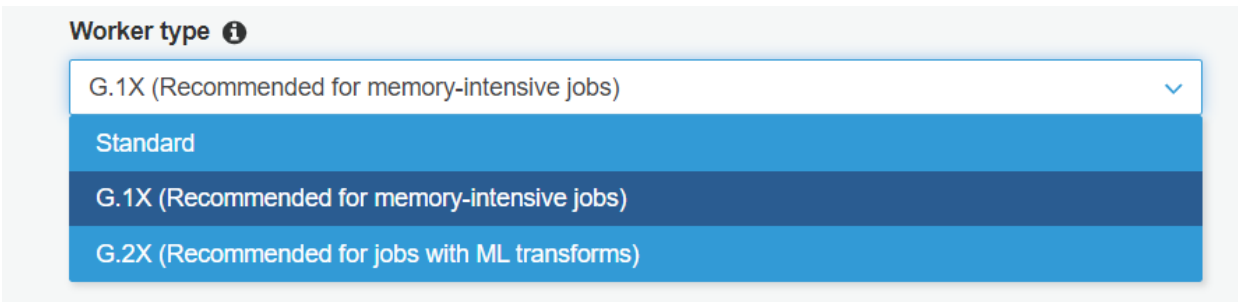

9. Select how do you want to run the job. Lets go ahead with the first option wherein Glue will construct a script for us.



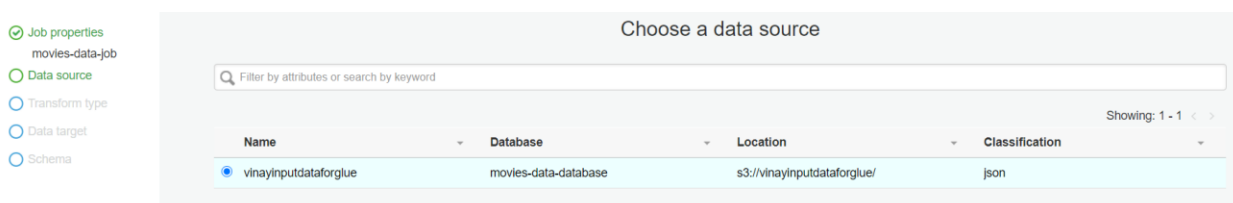

10. Examine the name of the script and where the scripts are going to get stored on S3.
11. Go to Advanced Properties, we will talk about job bookmarks later. Leave it Disabled.
12. Go to Monitoring options and we would like to monitor Job Metrics. This will load all of the job metrics into CloudWatch. Also, select continuous logging.



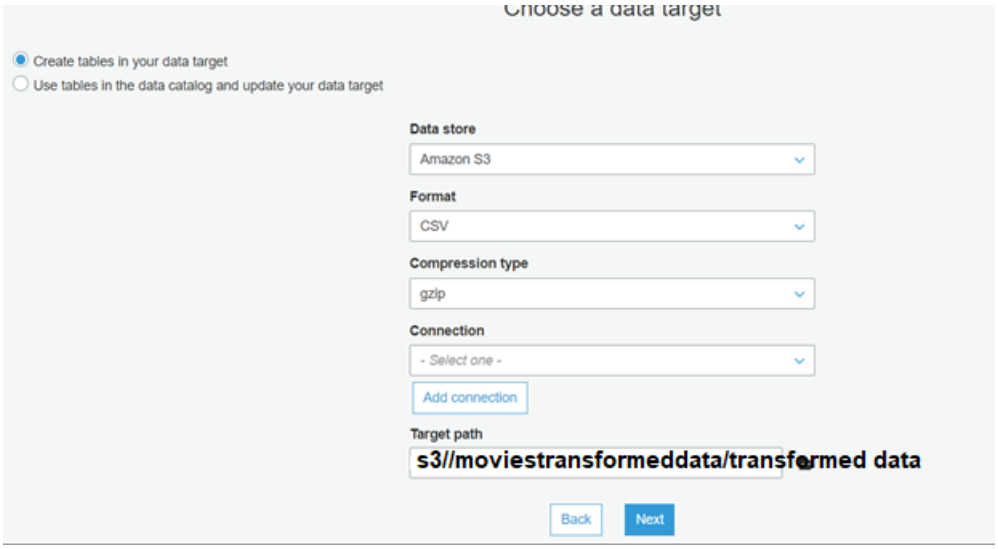

13. Go to security configurations and select the worker type and number of worker nodes. Worker type can be any one of the below mentioned options. Select G.1X for our exercise.



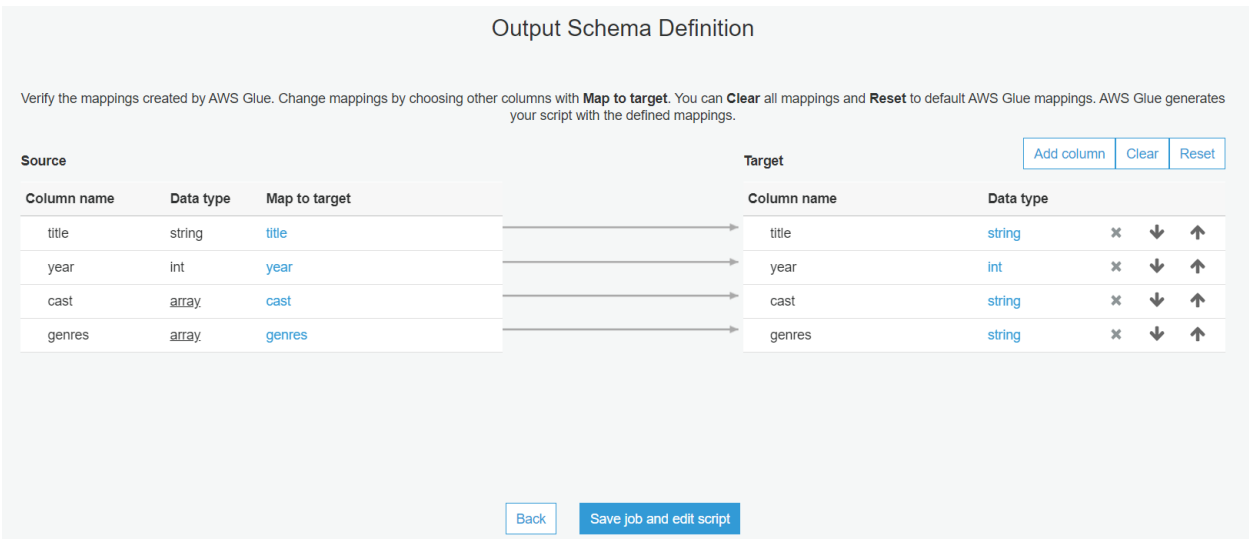

14. No Of workers or DPU's is 10 by default for spark jobs. Leave it to 10 for our exercise.
15. Click on Next.
16. We need to now select the data source. We have already setup the metadata catalog. Select "***moviestabledata***" and click on Next.




17. You will see that ML is not supported with Glue 2.0. If you want to use ML then you need to go ahead with Glue 1.0. Select Change Schema and click on Next. This will create a new schema for the data and load it to a target dataset.
18. We can now update our Glue Data Catalog or we can create tables in target dataset. Select "Create tables in your data target" and select as shown in below image. Ensure that you have created "**transformeddata**" folder in the same S3 bucket –. Mention **transformeddata** as the target path where data in CSV zipped format will be stored on S3.



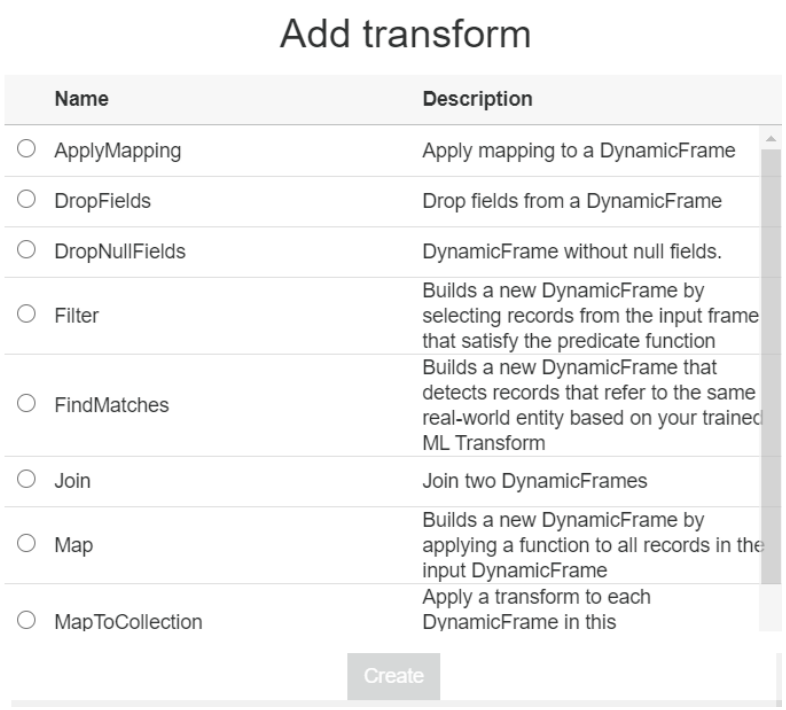

19. Click on Next. JSON data will now be converted into CSV format and You will now land up as shown below:




20. If any of the columns are not important then you can drop them. Click on “Save job and edit script”.
21. You will see that AWS Glue has setup a script for us. It will be a PySpark code. This code will connect to the datascource, apply transformation and store the output in the target.


On the left hand side, select the item and the code that does that job will be selected automatically.

22. Select “Transform” at the top-right-hand-side. It will show all the transformations that u can do.

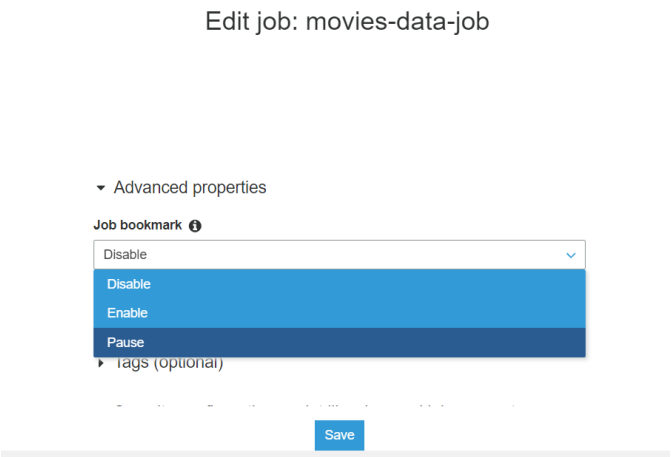



23. Select DropNullFields and you will see that code will be inserted into the script automatically. Get rid of that line of code. For our exercise, we don’t need.
24. Click on save and Run Job. It will allow you to override any parameters that you must have set earlier. You can do those changes and click on Run Job. For our exercise no changes are required at this step.
25. You can see the progress in “Continuous Logs” tab below.

26. What will happen as part of Running the job? Glue will spin up a cluster servers which will run this PySpark code against our input data, will make the transformations, it is going to compress the data and store that compressed data on S3 in our target folder.
27. Examine the target folder. You should have .gz file created. Download it. Unzip using 7-Zip and open the unzipped file using Notepad++. Observe the column headings and comma separated data. It will be a csv file.

**Lab 3: Job Bookmarks (observation)**

1. Go to AWS Glue -> Jobs -> select movies-data-job. Go to Actions and click on Edit Job
2. Expand Advanced Properties. You can select either Disable (by default), Enable or Pause. Select Pause and click on save.




3. In case of Pause, you need to specify from-value and to-value. To specify this, Go to Actions and click on Run Job, go to advanced properties and select pause again. Now you will be able to specify from and to values based on the run id's of jobs.