

Big Data: Amazon Athena, Glue, EMR

Table of Contents

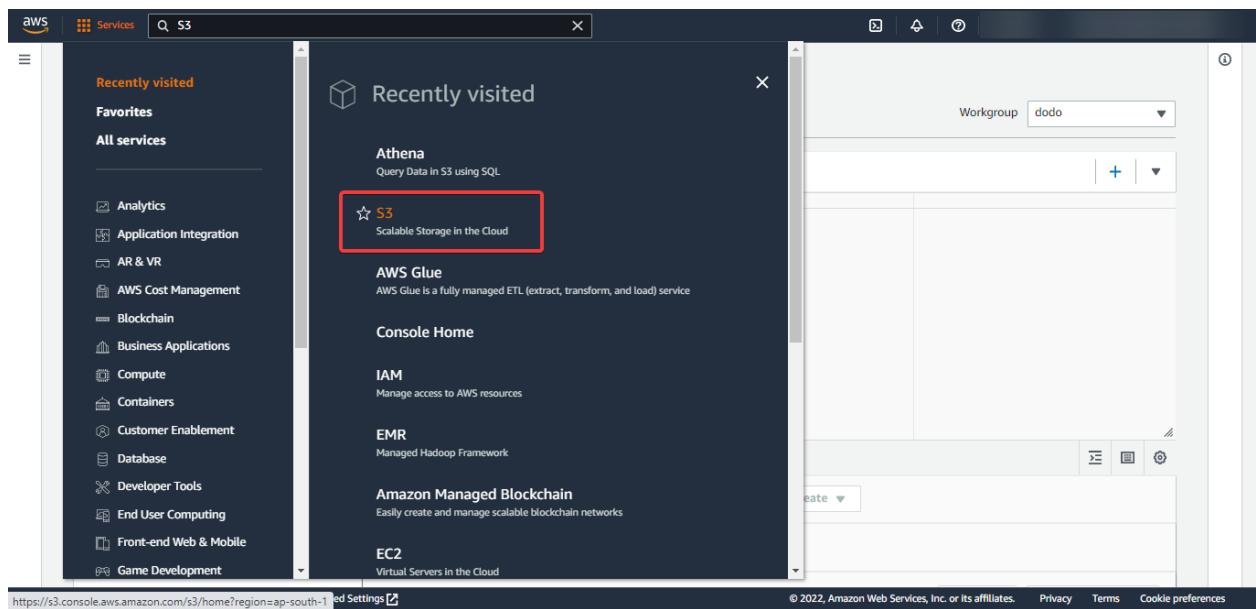
Query data from S3 files using Amazon Athena	2
Create a S3 bucket.....	2
AWS Athena: Querying S3 data	10
Table creation from S3 bucket	11
Workgroups Creation.....	14
Command SQL query.....	16
Query data on Amazon Athena from AWS Glue crawler	19
Create AWS Glue crawler.....	19
Run Crawler	24
SQL query on Athena Editor.....	27
Run Notebook on AWS Glue	28
Add Dev Endpoints.....	28
Create SageMaker notebook	32
Run PySpark Job in AWS Glue.....	39
Using Amazon EMR with AWS Glue Catalog	45
Create IAM Role for Glue.....	45
Create Glue database and glue cluster	48
Launch EMR Cluster and Run Jupyter Notebook.....	50
IAM ROLE for Glue: Sagemaker.....	54
Create IAM role for Glue: Crawler	59
CleanUp	61

Big Data: Amazon Athena, Glue, EMR

Query data from S3 files using Amazon Athena

Create a S3 bucket

- Log in to **AWS Console**
- Select **S3**.



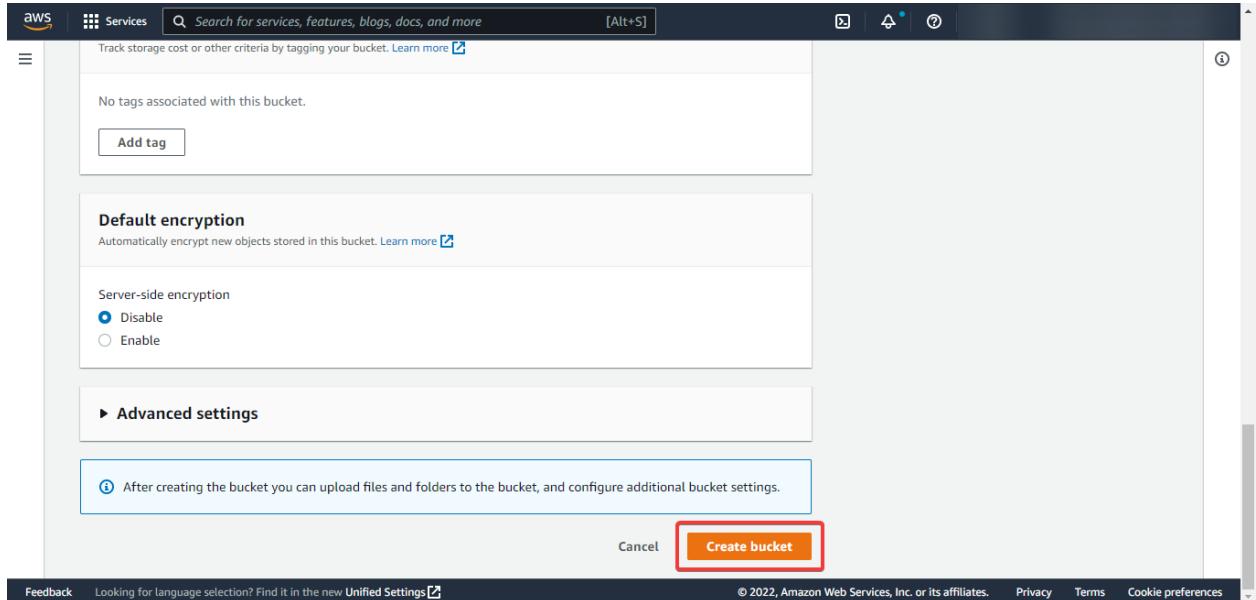
- Click **Create bucket**

The screenshot shows the AWS S3 service page. On the left, there's a sidebar with links for Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, and Access analyzer for S3. Below that are links for Block Public Access settings and Storage Lens (Dashboards and AWS Organizations settings). A Feature spotlight section is also present. The main content area displays an 'Account snapshot' with metrics: Total storage (507.6 GB), Object count (10.6 M), and Avg. object size (50.4 KB). It also includes a note about enabling advanced metrics. Below this is a table titled 'Buckets (292) Info' with columns for Name, AWS Region, Access, and Creation date. The table lists three buckets: '031342435657-templates-multibranch-deployment' (US East (N. Virginia)), 'ae-s3bucket-aknijixe6b1w' (US East (Ohio)), and 'ak5050' (US East (N. Virginia)). At the bottom, there are links for Feedback, Unified Settings, and various AWS terms like Privacy, Terms, and Cookie preferences.

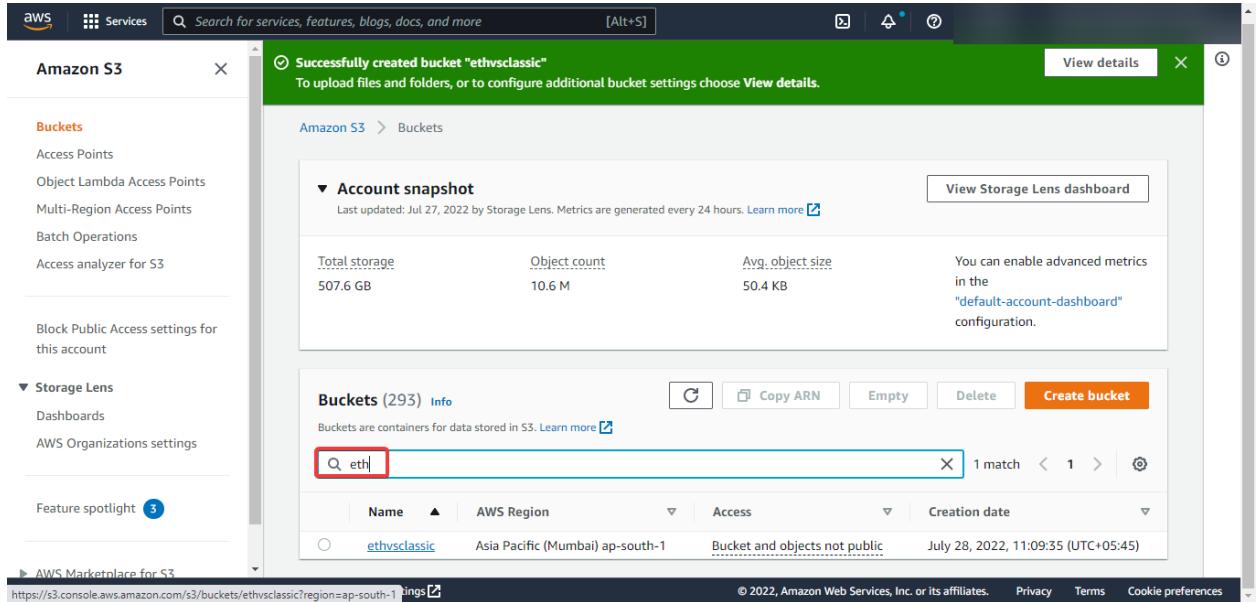
- Enter **Bucket name** ethvsclassic and select **AWS Region Asia Pacific (Mumbai)ap-south-1**

The screenshot shows the 'Create bucket' wizard. In the 'General configuration' step, the 'Bucket name' field contains 'ethvsclassic' and the 'AWS Region' dropdown is set to 'Asia Pacific (Mumbai) ap-south-1'. Below this, there's a section for 'Copy settings from existing bucket - optional' with a 'Choose bucket' button. In the 'Object Ownership' step, it says 'Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.' There are two options: 'ACLs disabled (recommended)' (selected) and 'ACLs enabled'. Both options have descriptive text below them. At the bottom, there are links for Feedback, Unified Settings, and various AWS terms like Privacy, Terms, and Cookie preferences.

- Keep others default option.
- Once completed **Create bucket**.



- Finally, you will see successfully bucket create at the top of the page.



Click on your bucket **ethvsclassic**.

The screenshot shows the AWS S3 service page. A green banner at the top indicates that the bucket 'ethvsclassic' has been successfully created. Below the banner, the 'Account snapshot' section displays basic statistics: Total storage (507.6 GB), Object count (10.6 M), and Avg. object size (50.4 KB). A note says you can enable advanced metrics. The main area shows a list of buckets, with 'ethvsclassic' highlighted by a red box. The bucket details show it was created on July 28, 2022, at 11:09:35 (UTC+05:45) and is located in the Asia Pacific (Mumbai) region (ap-south-1). Buttons for 'Copy ARN', 'Empty', 'Delete', and 'Create bucket' are visible.

Click **create folder**.

The screenshot shows the 'ethvsclassic' bucket details page. The 'Objects' tab is selected. At the top of the objects list, there is a 'Actions' menu with a 'Create folder' button highlighted by a red box. Below the actions, there is a search bar and a table header for 'Name', 'Type', 'Last modified', 'Size', and 'Storage class'. A message at the bottom states 'No objects' and 'You don't have any objects in this bucket.'

If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.

Folder

Folder name
ethereum_collection /

Folder names can't contain "/". See rules for naming [\[link\]](#)

Server-side encryption

The following settings apply only to the new folder object and not to the objects contained within it.

Server-side encryption
 Disable
 Enable

Cancel **Create folder**

- Click **create folder**

If your bucket policy prevents uploading objects without specific tags, metadata, or access control list (ACL) grantees, you will not be able to create a folder using this configuration. Instead, you can use the [upload configuration](#) to upload an empty folder and specify the appropriate settings.

Folder

Folder name
ethereum_collection /

Folder names can't contain "/". See rules for naming [\[link\]](#)

Server-side encryption

The following settings apply only to the new folder object and not to the objects contained within it.

Server-side encryption
 Disable
 Enable

Cancel **Create folder**

Open your folder **ethereum_collection**

The screenshot shows the AWS S3 console interface. A green success message at the top states: "Successfully created folder \"ethereum_collection\". Operation successfully completed." The left sidebar shows navigation options like Buckets, Storage Lens, and Feature spotlight. The main area is titled "ethvsclassic" and shows an "Objects (1)" section. A table lists one object: "ethereum_collection/" which is a Folder. The "Upload" button is highlighted with a red box.

Upload files from your PC.

The screenshot shows the AWS S3 console interface. The left sidebar shows Buckets, Storage Lens, and Feature spotlight. The main area is titled "ethereum_collection/" and shows an "Objects (0)" section. The "Upload" button is highlighted with a red box. A message at the bottom says "No objects" and "You don't have any objects in this folder." The "Upload" button is also highlighted with a red box.

- **Add files**

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more [\[Link\]](#)

Drag and drop files and folders you want to upload here, or choose **Add files**, or **Add folders**.

Files and folders (0)		Add files	Add folder
All files and folders in this table will be uploaded.			
<input type="text" value="Find by name"/> < 1 >			
Name	Folder	Type	Size
No files or folders			
You have not chosen any files or folders to upload.			

- Tick the uploaded files.
- Click **upload**.

Files and folders (2 Total, 75.9 MB)

<input checked="" type="checkbox"/>	Name	Folder	Type	Size
<input checked="" type="checkbox"/>	ethereum_classic_ether	-	text/csv	37.9 MB
<input checked="" type="checkbox"/>	eum-000000000000.csv	-	text/csv	37.9 MB
<input checked="" type="checkbox"/>	ethereum_classic_ether	-	text/csv	37.9 MB
<input checked="" type="checkbox"/>	eum-000000000001.csv	-	text/csv	37.9 MB

Destination

Destination
s3://ethvsclassic/ethereum_collection/

▶ **Destination details**
Bucket settings that impact new objects stored in the specified destination.

▶ **Permissions**
Grant public access and access to other AWS accounts.

▶ **Properties**
Specify storage class, encryption settings, tags, and more.

Cancel **Upload**

It makes take some time depend on file size.

- Finally, files have been uploaded in s3 bucket.

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

Upload succeeded View details below.

Upload: status

The information below will no longer be available after you navigate away from this page.

Summary

Destination	Succeeded	Failed
s3://ethvsclassic/ethereum_collection/	2 files, 75.9 MB (100.00%)	0 files, 0 B (0%)

Files and folders Configuration

Files and folders (2 Total, 75.9 MB)

Name	Folder	Type	Size	Status	Error
ethereum_classic_ethereum-000000000000.csv	-	text/csv	37.9 MB	Succeeded	-
ethereum_classic_ethereum-000000000001.csv	-	text/csv	37.9 MB	Succeeded	-

Feedback Looking for language selection? Find it in the new Unified Settings. © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The screenshot shows the AWS S3 console after a file upload. At the top, a green banner indicates 'Upload succeeded' with a link to 'View details below'. Below this, the title 'Upload: status' is displayed. A note says 'The information below will no longer be available after you navigate away from this page.' Under the 'Summary' section, there's a table comparing 'Destination' (s3://ethvsclassic/ethereum_collection/) against 'Succeeded' (2 files, 75.9 MB, 100.00%) and 'Failed' (0 files, 0 B, 0%). The main area shows a table of 'Files and folders' with two entries: 'ethereum_classic_ethereum-000000000000.csv' and 'ethereum_classic_ethereum-000000000001.csv', both listed under 'Status' as 'Succeeded'. A red box highlights the first row of this table. At the bottom, there are links for 'Feedback', 'Unified Settings', copyright notice, and privacy/terms/cookie preferences.

AWS Athena: Querying S3 data

- Go to **Amazon Athena** on AWS Console
- Select **Services** and click **Analytics**
- Select **Athena**

The screenshot shows the AWS Services menu on the left with 'Analytics' selected. A modal window titled 'Analytics' is open, featuring a section for 'Athena' which is highlighted with a red box. The 'Athena' section contains the text 'Query Data in S3 using SQL'. To the right of the modal, there is a preview of an S3 bucket listing with two objects, both 37.9 MB in size and stored in the 'Standard' storage class. The URL in the browser bar is <https://ap-south-1.console.aws.amazon.com/athena/home?region=ap-south-1>.

- Go to Query editor
- Click **Create** and select **S3 bucket data**

The screenshot shows the Amazon Athena Query editor interface. In the top navigation bar, 'Amazon Athena' and 'Query editor' are visible. The main area has tabs for 'Editor', 'Recent queries', 'Saved queries', and 'Settings'. On the right, a 'Workgroup' dropdown is set to 'dodo'. On the left, a sidebar under 'Data' shows options like 'Create a table from data source', 'Data source' (set to 'AwsDataCatalog'), and 'Database' (set to 'Choose a database'). A dropdown menu is open over the 'Create' button, with 'S3 bucket data' highlighted with a red box. Below the dropdown, other options include 'AWS Glue Crawler' and 'Create with SQL'. The bottom part of the screen shows a query editor with an SQL input field and a 'Run' button.

Table creation from S3 bucket

Create a Table name : ethereum

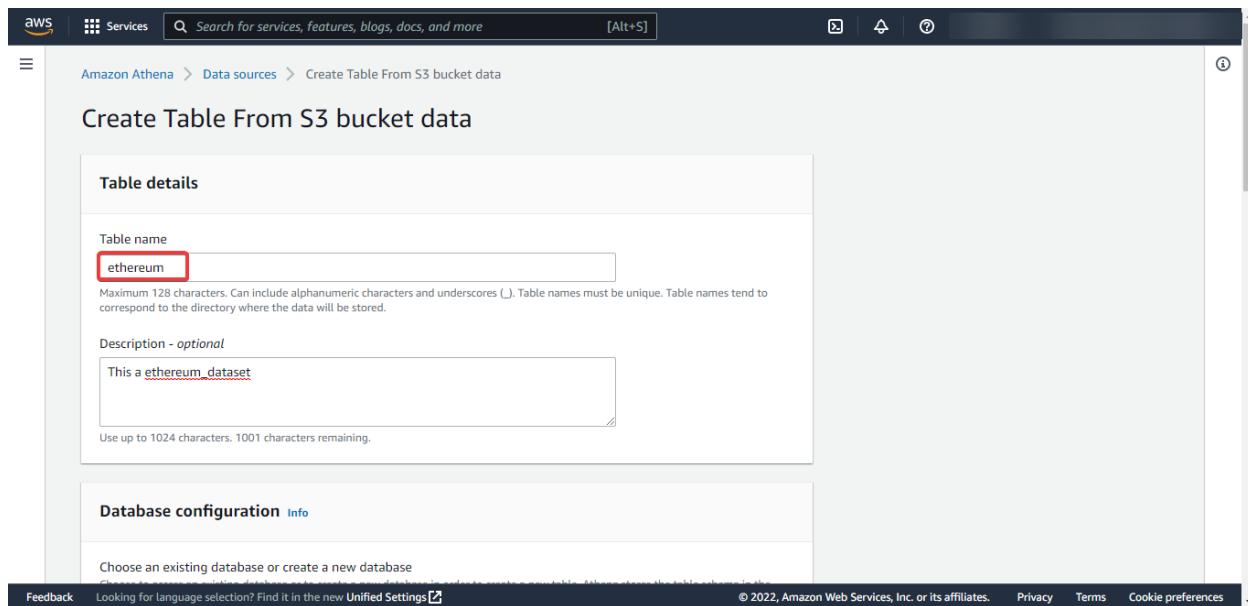


Table details

Table name: ethereum

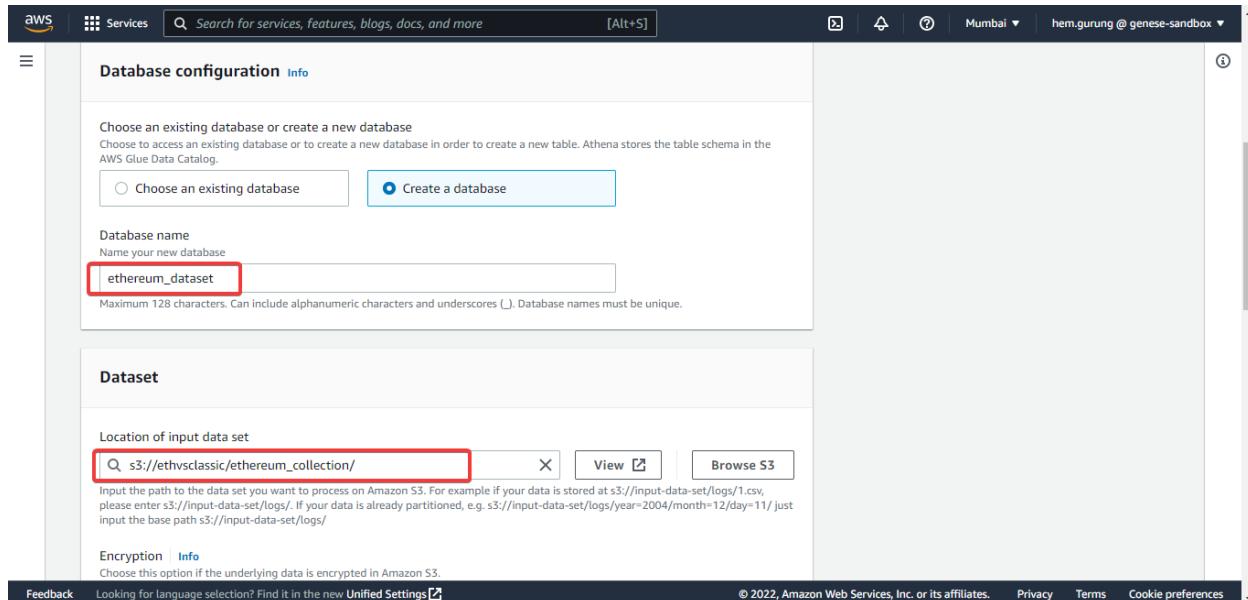
Description - optional: This is a ethereum dataset

Database configuration

Choose an existing database or create a new database

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Create **Database name**: ethereum_dataset and
- Input s3 dataset location s3://ethvclassic/ethereum_collection/



Database configuration

Choose an existing database or create a new database

Choose an existing database Create a database

Database name: ethereum_dataset

Dataset

Location of input data set: s3://ethvclassic/ethereum_collection/

Encryption Info

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Add **Data format** as CSV file
- Choose Bulk add columns

The screenshot shows the AWS Data Format configuration interface. At the top, the 'Data format' section has 'CSV' selected. Below it, the 'Column details' section lists two columns: 'address' (string type) and 'eth_balance' (string type). At the bottom of this section are 'Add a column' and 'Bulk add columns' buttons, with 'Bulk add columns' highlighted by a red box.

- Add Bulk add columns : address string, eth_balance string

The screenshot shows the 'Bulk add columns' dialog box. It contains instructions for defining columns in name value pairs separated by commas. Below this is a text input field containing 'address string, eth_balance string'. At the bottom right of the dialog are 'Cancel' and 'Add' buttons, with 'Add' highlighted by a red box.

- Once completed **Create table**

The screenshot shows the 'Create table' dialog box in the AWS Glue Data Catalog. At the top, there's a search bar and a 'Services' dropdown. Below the search bar, there are fields for 'Column name' (with placeholder 'Enter a column name') and 'Column type' (with placeholder 'Select a column type'). A 'Remove' button is also present. A large 'Add column' button is located below these fields. In the center, there's a section titled 'Preview table query' containing the following SQL code:

```
1 CREATE EXTERNAL TABLE IF NOT EXISTS `ethereum_dataset`.`ethereum` (
2   `address` string,
3   `eth_balance` string
4 )
5 ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe'
6 WITH SERDEPROPERTIES (
7   'serialization.format' = ',',
8   'field.delim' = ','
9 ) LOCATION 's3://ethvsclassic/ethereum_collection/'
10 TBLPROPERTIES ('has_encrypted_data'= 'false');
```

At the bottom right of the dialog is a red 'Create table' button.

Now Query will successfully run if there is already a workplace.

Note: If you have workplace issue setting follow below command.

The screenshot shows the AWS Glue Data Catalog Editor interface. The top navigation bar includes 'Services', a search bar, and a 'Workgroup' dropdown set to 'dodo'. The main area has tabs for 'Data', 'Editor', 'Recent queries', 'Saved queries', and 'Settings'. On the left, there are sections for 'Data source' (set to 'AwsDataCatalog'), 'Database' (set to 'default'), and 'Tables and views' with a 'Create' button. The central workspace shows 'Query 1' and 'Query 2'. Query 1 contains the same SQL code as the previous screenshot. Below the queries is a 'SQL' editor with 'Ln 1, Col 1'. At the bottom of the editor are buttons for 'Run again', 'Explain', 'Cancel', 'Save', 'Clear', and 'Create'. The status bar at the bottom indicates 'Completed' with a green checkmark, 'Time in queue: 46 ms', 'Run time: 404 ms', and 'Data scanned: -'. The footer includes standard links for 'Feedback', 'Unified Settings', 'Privacy', 'Terms', and 'Cookie preferences'.

- Before running query, you have to create workgroup to save your output in separate bucket.

Workgroups Creation

- Go to **workgroups** on left side of the **Amazon Athena** page.

The screenshot shows the Amazon Athena Query editor interface. On the left sidebar, the 'Workgroups' option is highlighted with a red box. The main area displays a 'Data' panel with 'Data source' set to 'AwsDataCatalog' and 'Database' set to 'ethereum_dataset'. Below this, there are sections for 'Tables and views' and 'Views'. A 'Query 3' tab is open in the top right. At the bottom, there's a SQL editor with buttons for Run, Explain, Cancel, Save, Clear, and Create, and tabs for 'Query results' and 'Query stats'.

- Click **create workgroup**

The screenshot shows the Amazon Athena Workgroups page. The 'Workgroups' option is highlighted with a red box in the sidebar. The main area shows a table with two entries: 'dodo' and 'primary'. A 'Create workgroup' button is prominently displayed at the top of the table area. The table columns include Name, Description, Query engine v..., Query engine u..., Added, and Status.

Name	Description	Query engine v...	Query engine u...	Added	Status
dodo	-	Athena engine ver...	Automatic	2022-07-27T14:1...	Current
primary	-	Athena engine ver...	Automatic	2019-04-10T11:4...	Enabled

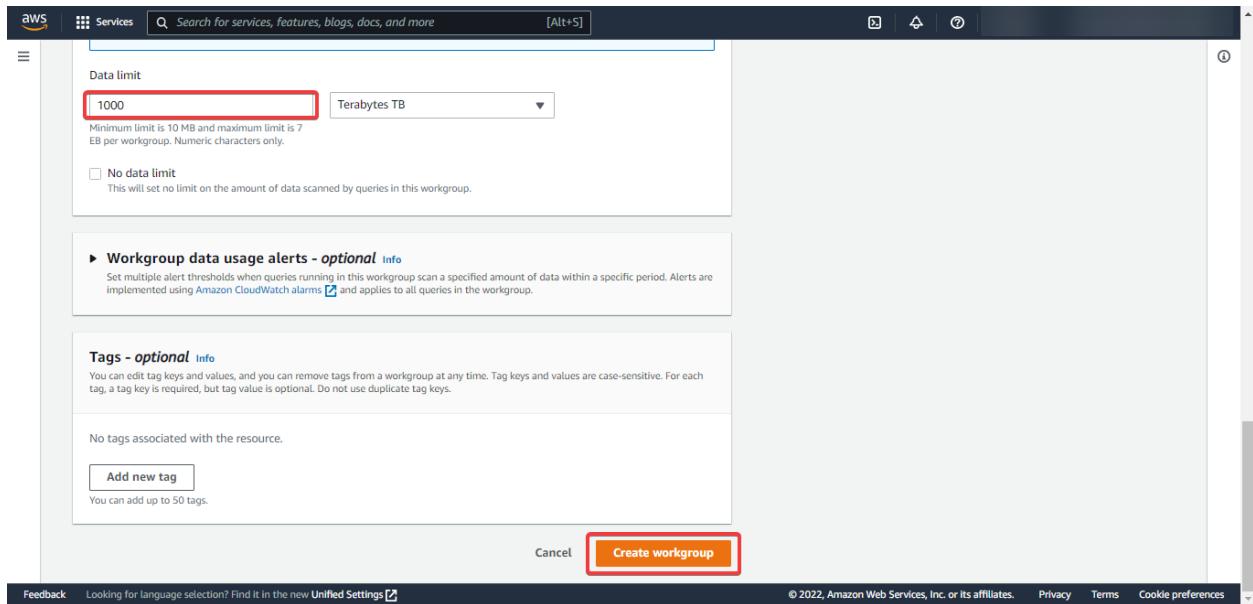
- Give a **Workgroup name JOJO**

The screenshot shows the 'Workgroup details' section of the AWS Athena Workgroup creation interface. It includes fields for 'Workgroup name' (set to 'JOJO'), 'Description - optional', and 'Query engine version' settings (Automatic). A note at the bottom of the 'Query engine version' section states: 'Use up to 1024 characters. 1024 characters remaining.'

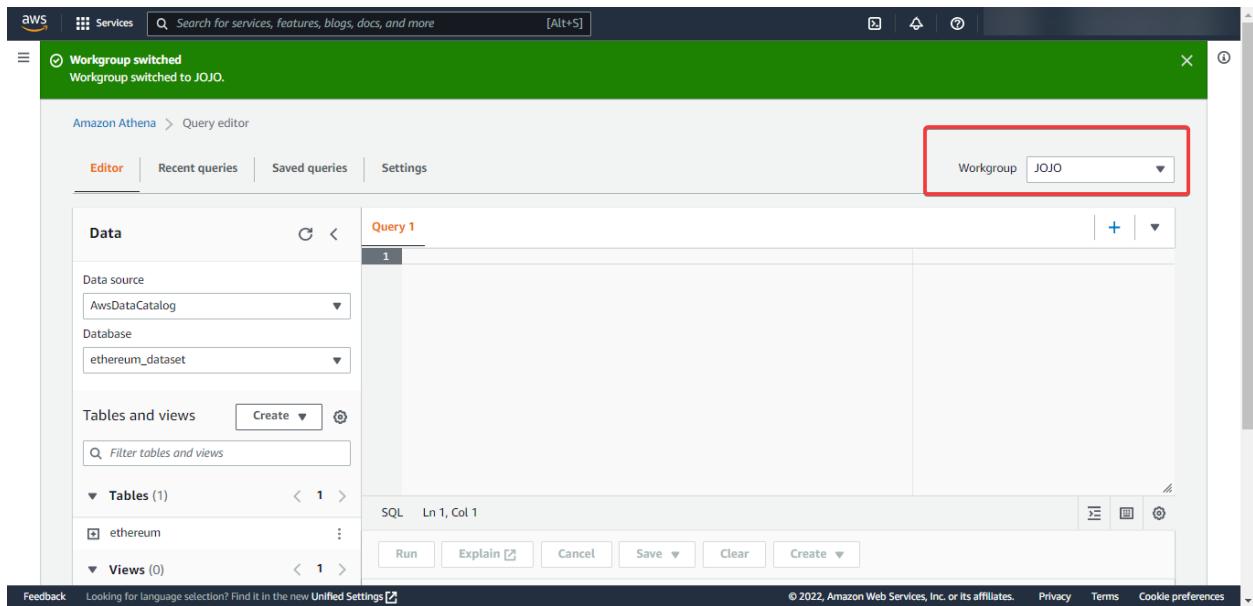
- Create a folder in S3 bucket where you want to save your output.
- Give s3 location to store query result **s3://ethvclassic/output/**

The screenshot shows the 'Query result configuration' and 'Settings' sections of the AWS Athena Workgroup configuration interface. In the 'Query result configuration' section, the 'Location of query result' field is set to 's3://ethvclassic/output/'. The 'Expected bucket owner' field is empty. Under 'Encrypt query results', the 'Enable' checkbox is unchecked. Under 'Assign bucket owner full control over query results', the checkbox is checked. In the 'Settings' section, under 'Metrics', the 'Publish query metrics to AWS CloudWatch' checkbox is checked.

- Limit **Data limit 1000 Terabytes TB**
- Once completed click **Create workgroup.**



- Change your **workgroup** into JOJO on the right side of the Amazon Athena Query editor dashboard.



Command SQL query

- Run the following command on query editor
select * from ethereum limit 10;

The screenshot shows the AWS Glue Data Catalog Editor interface. On the left, the 'Data' sidebar displays a 'Data source' set to 'AwsDataCatalog' and a 'Database' set to 'ethereum_dataset'. Under 'Tables and views', there is one table named 'ethereum' and zero views. In the main area, a query editor window titled 'Query 1' contains the SQL command: 'select * from ethereum limit 10;'. Below the query, the status bar shows 'SQL Ln 1, Col 25'. A red box highlights the 'Run' button. The results section below shows '(Results (0))' and includes 'Copy' and 'Download results' buttons.

- Query results are run successfully.
- Click **Download results**.

The screenshot shows the AWS Glue Data Catalog Editor interface after a query has been run. The 'Tables' section now shows '(1)' tables, with 'ethereum' listed. The results section displays a table titled 'Results (10)' with 10 rows of data. The 'Download results' button is highlighted with a red box. The table data is as follows:

#	address	eth_balance
1	0xe141cff6024f4b2c7abbd64d958b52208b378a8	0
2	0x4759e4a499f296f8a3b9a05d5a317002ec083558	0
3	0xa3de13edeae9b9ff7e269230a719669f16799557	0
4	0x69288681d885d415dbe0f1cd51c76e1a3b5511d2	0
5	0x35115d371a504f799a80198de20ca9abeea1d64f	0
6	0x9e9c8d7d94d4db675321c44bda1ad4138aba31e0	0
7	0xceef2aac10c2051cbdac0844b9cf0cf509cf03f	0
8	0x141cff6024f4b2c7abbd64d958b52208b378a8	0
9	0x4759e4a499f296f8a3b9a05d5a317002ec083558	0
10	0xa3de13edeae9b9ff7e269230a719669f16799557	0

- Similarly, result is also stored automatically in S3 bucket folder.
s3://ethvclassic/output/

AWS Services Search for services, features, blogs, docs, and more [Alt+S]

Amazon S3 output/

Copy S3 URI

Buckets

- Access Points
- Object Lambda Access Points
- Multi-Region Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

Storage Lens

- Dashboards
- AWS Organizations settings

Feature spotlight 3

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

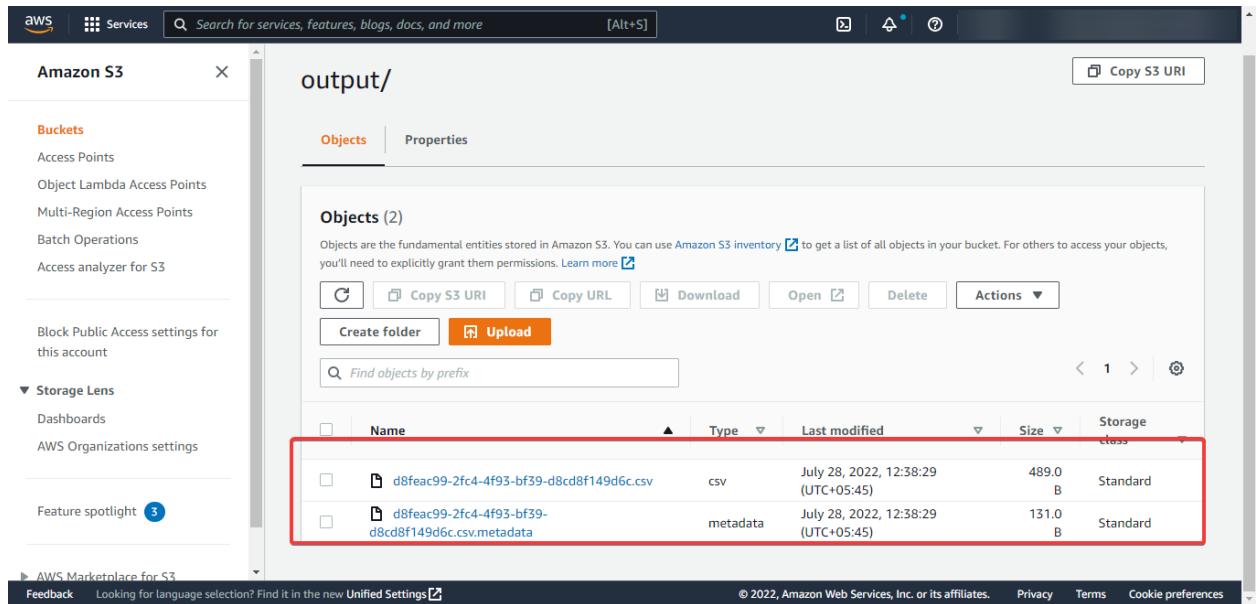
Actions ▾

Create folder Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
d8feac99-2fc4-4f93-bf39-d8cd8f149d6c.csv	csv	July 28, 2022, 12:38:29 (UTC+05:45)	489.0 B	Standard
d8feac99-2fc4-4f93-bf39-d8cd8f149d6c.csv.metadata	metadata	July 28, 2022, 12:38:29 (UTC+05:45)	131.0 B	Standard

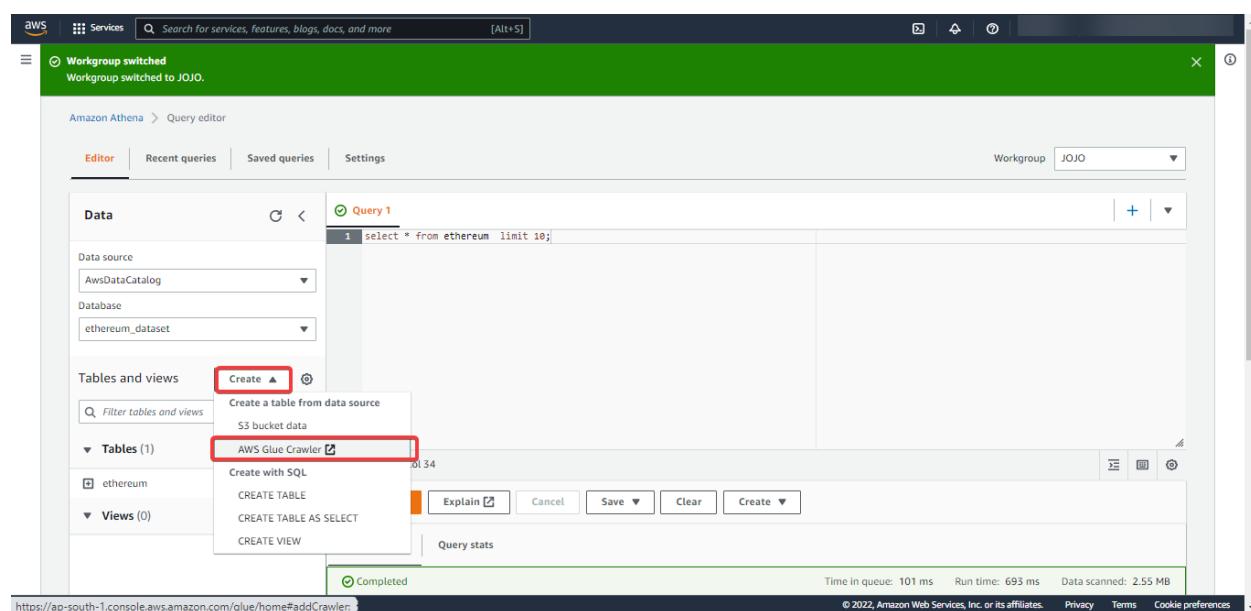
AWS Marketplace for S3 Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



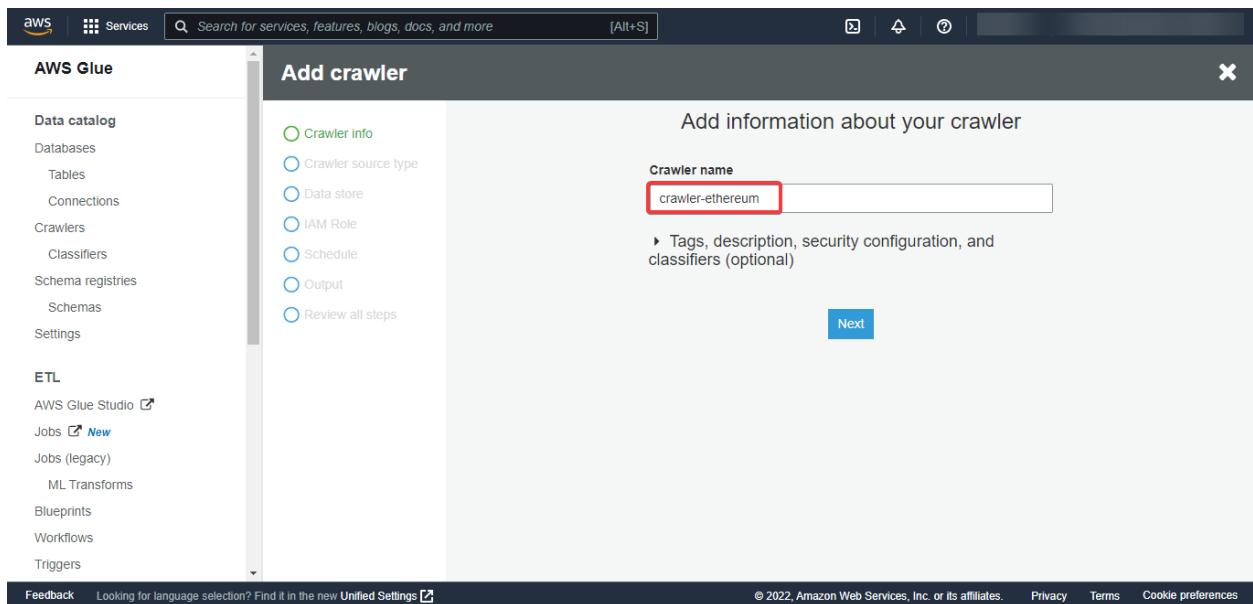
Query data on Amazon Athena from AWS Glue crawler

Create AWS Glue crawler

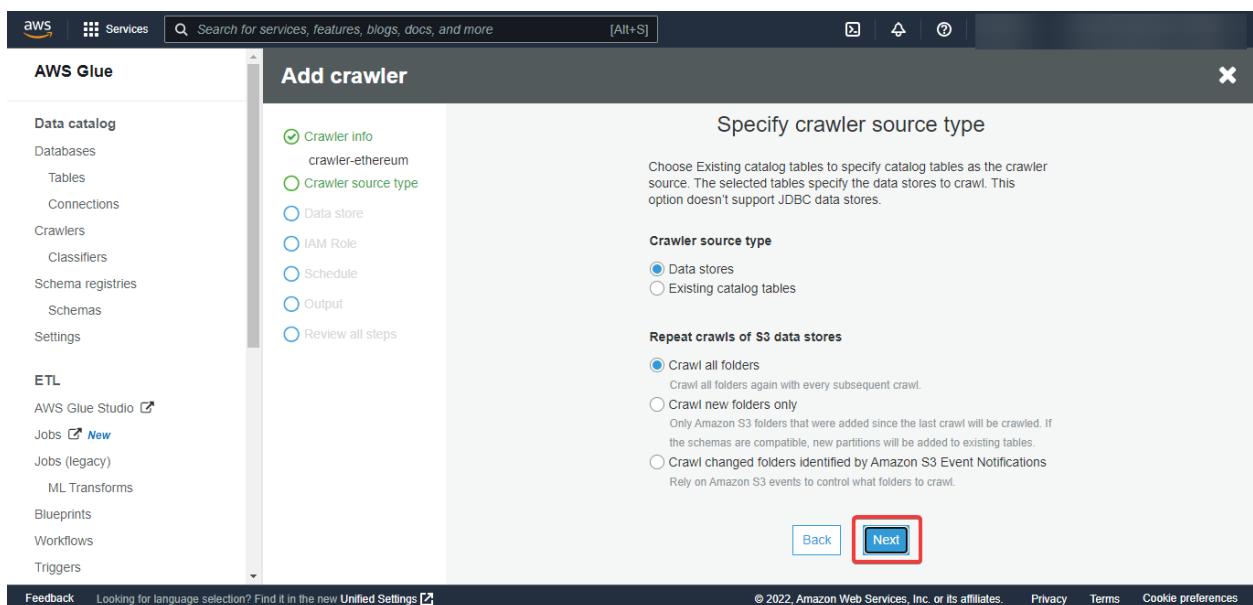
- Go to **Amazon Athena** query editor dashboard
- Click **Create** and select **AWS Glue Crawler**.



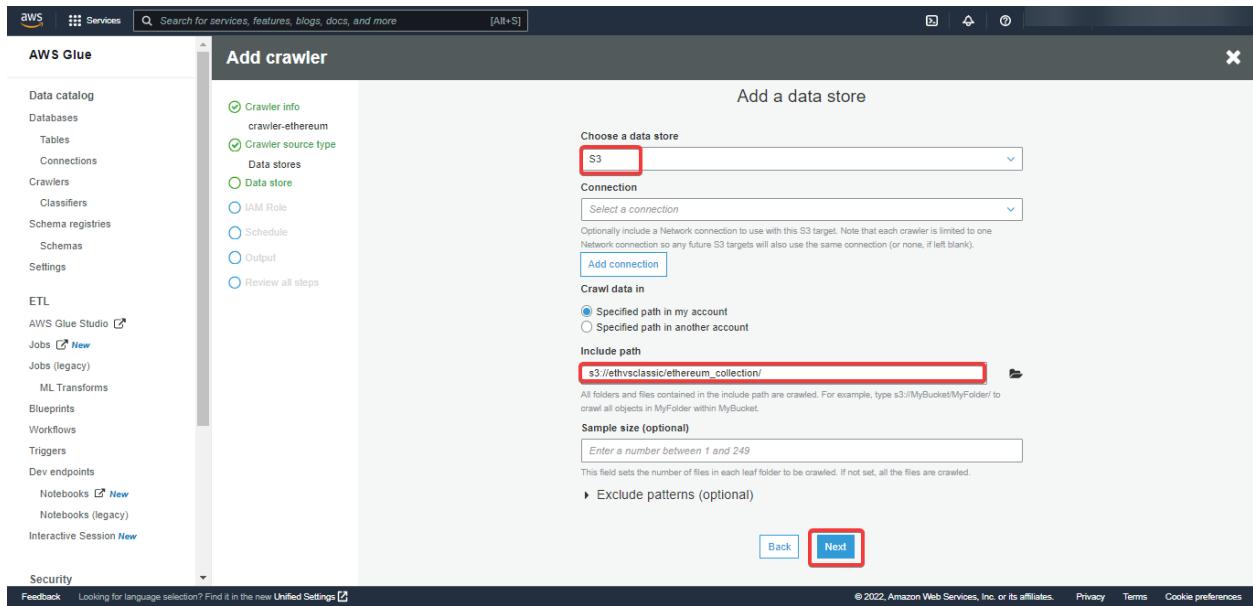
- **Add crawler**
- Create **crawler name** Crawler-ethereum



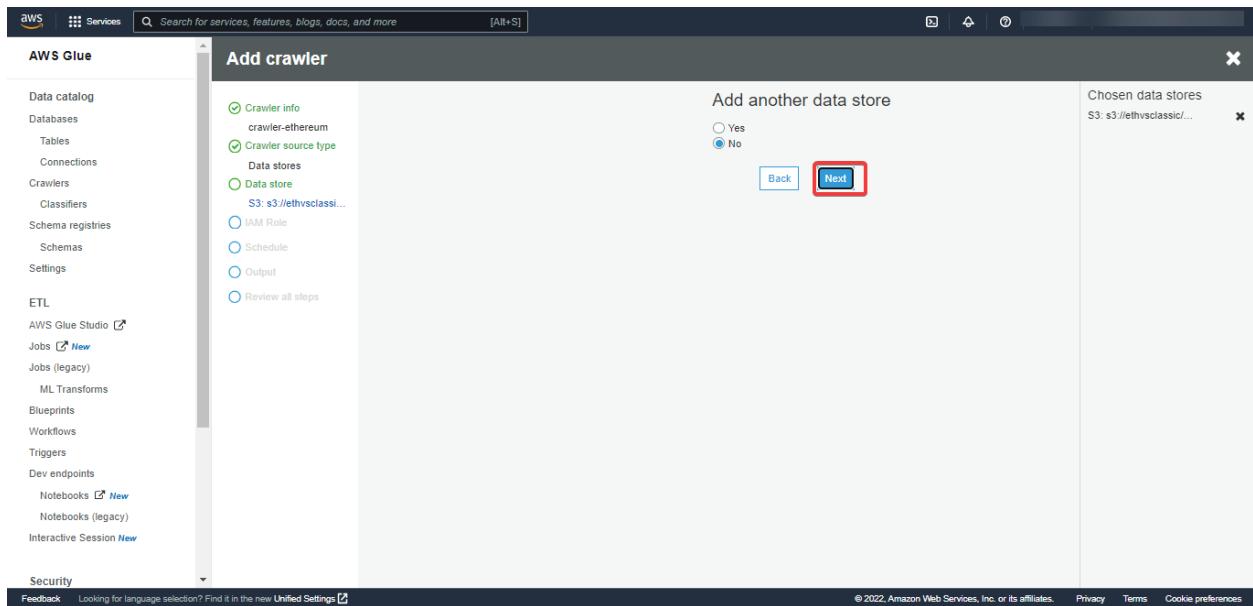
Click Next with default option.



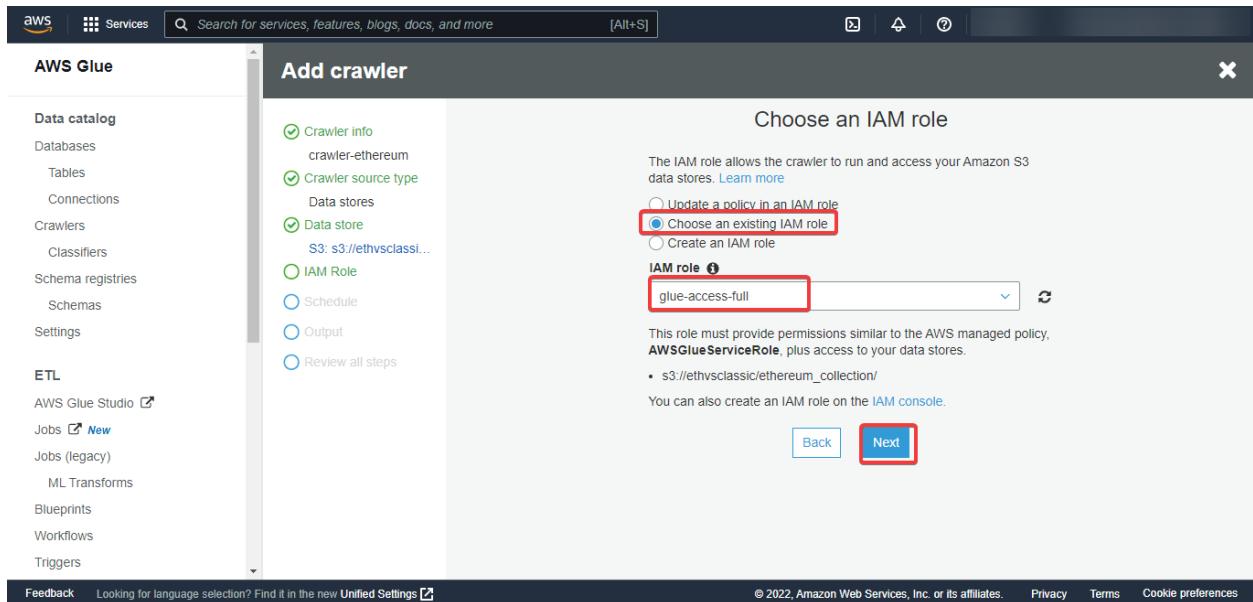
- **Choose a data store as s3.**
- **Include path s3 dataset location s3://ethvsclassic/ethereum_collection/**
- **Click Next.**



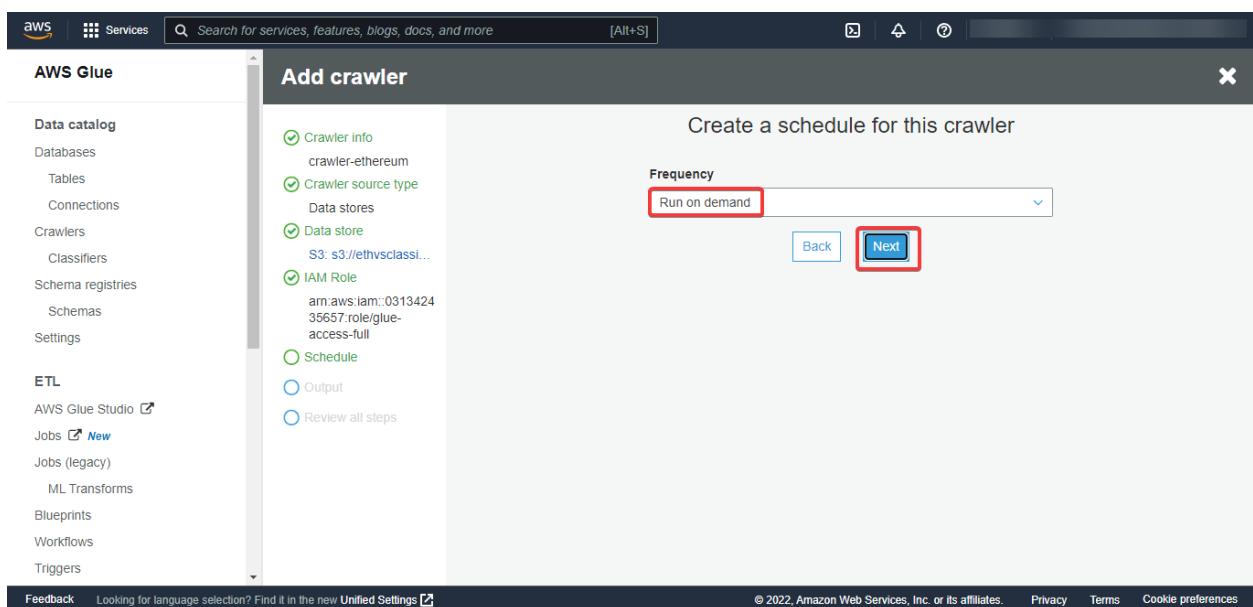
- **Add another data store No**
- **Click Next**



- **Choose an existing IAM role that access glue.** See bottom page about creating IAM role.
- **Click Next.**



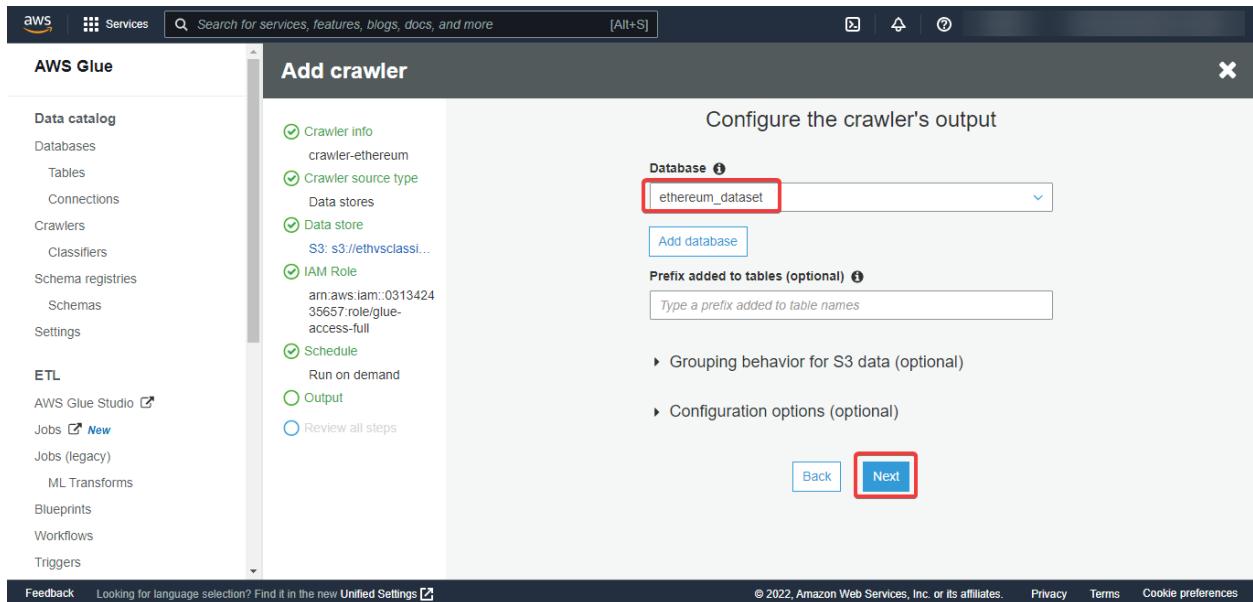
- Set Frequency as **Run-on demand**.
- Click **Next**.



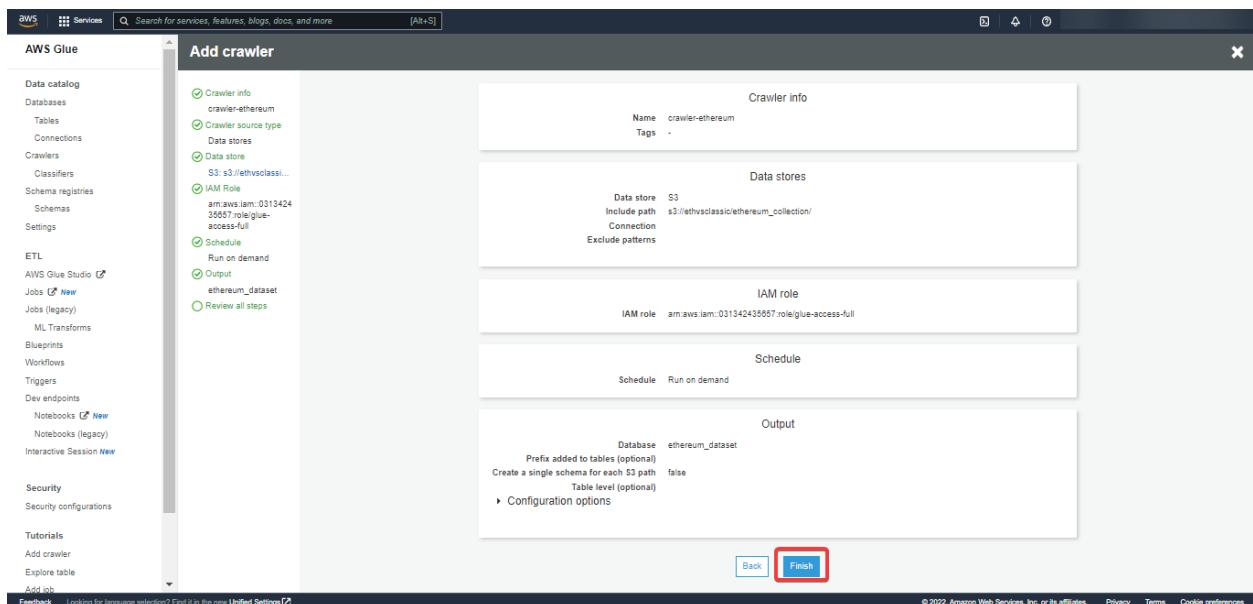
Choose exiting **Database** as ethereum_dataset(s3 bucket is already attached on aws Glue).

Or create a new Database on **Databases name** on AWS Glue **Data catalog**.

Click **Next**.



- Once you review all, click **Finish**.



Run Crawler

- Select your **crawler** crawler-ethereum and click **Run crawler**.

The screenshot shows the AWS Glue service interface. On the left, there's a sidebar with navigation links for Data catalog, Databases, Tables, Connections, Crawlers (which is selected and highlighted in orange), Classifiers, Schema registries, Schemas, and Settings. Below that is the ETL section with links for AWS Glue Studio, Jobs (New), Jobs (legacy), ML Transforms, Blueprints, Workflows, and Triggers. The main content area is titled 'Crawlers' and contains a message: '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.' Below this is a success message: 'Crawler crawler-ethereum was created to run on demand. Run it now?' with a green 'Run it now?' button. At the top of the crawler list table, there are buttons for 'Add crawler' (disabled), 'Run crawler' (highlighted with a red box and a red number '2'), 'Action', and a search bar. The table has columns: Name, Schedule, Status, Logs, Last runtime, Median runtime, Tables updated, and Tables added. There are two rows: 'crawler-ethereum' (selected with a red box and a red number '1') and 'mg-glue'. The 'crawler-ethereum' row shows 'Ready' status, '0 secs' for both last and median runtime, and 0 for both updated and added tables. The 'mg-glue' row shows 'Ready' status, 'Logs' under Status, '46 secs' for both last and median runtime, and 0 for updated and 1 for added tables. The bottom of the page includes standard footer links for Feedback, Unified Settings, Copyright notice (© 2022, Amazon Web Services, Inc. or its affiliates.), Privacy, Terms, and Cookie preferences.

Name	Schedule	Status	Logs	Last runtime	Median runtime	Tables updated	Tables added
<input checked="" type="checkbox"/> crawler-ethereum		Ready		0 secs	0 secs	0	0
<input type="checkbox"/> mg-glue		Ready	Logs	46 secs	46 secs	0	1

- Once a crawler is successful. A Table is added.

The screenshot shows the AWS Glue service interface. On the left, there's a sidebar with navigation links for Data catalog, ETL, and other AWS services. The main area is titled 'Crawlers' and contains a message box stating: 'Crawler "crawler-ethereum" completed and made the following changes: 1 tables created, 0 tables updated. See the tables created in database ethereum_dataset.' Below this is a table listing two crawlers:

Name	Schedule	Status	Logs	Last runtime	Median runtime	Tables updated	Tables added
crawler-ethereum		Ready	Logs	39 secs	39 secs	0	1
mg-glue		Ready	Logs	46 secs	46 secs	0	1

At the bottom, there are footer links for Feedback, Unified Settings, Copyright notice, Privacy, Terms, and Cookie preferences.

This screenshot is similar to the first one but shows a different state for the crawlers. The 'crawler-ethereum' crawler is now listed as 'Stopping'. The table data is as follows:

Name	Schedule	Status	Logs	Last runtime	Median runtime	Tables updated	Tables added
crawler-ethereum		Stopping	Logs	39 secs	39 secs	0	1
mg-glue		Ready	Logs	46 secs	46 secs	0	1

- Go to **databases** below AWS Glue Data catalog.
- Open your database folder ethereum_dataset

S | Services | Search for services, features, blogs, docs, and more [Alt+S] | ☰ | 🔍 | ⓘ

AWS Glue

Data catalog

Databases

Tables
Connections
Crawlers
Classifiers
Schema registries
Schemas
Settings

ETL

AWS Glue Studio
Jobs New
Jobs (legacy)
ML Transforms
Blueprints
Workflows
Triggers

Feedback Looking for language selection? Find it in the new [Unified Settings](#)

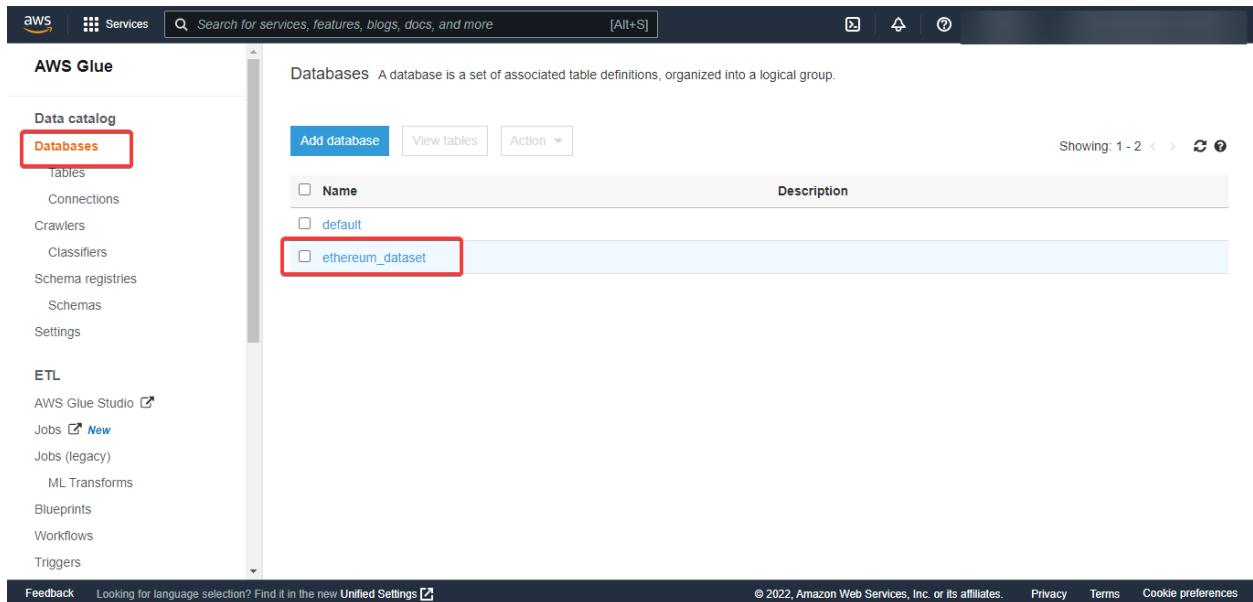
Databases A database is a set of associated table definitions, organized into a logical group.

Add database View tables Action ▾

Showing: 1 - 2 < > ⌂ ⓘ

Name	Description
default	
ethereum_dataset	

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



You will see your database collection.

S | Services | Search for services, features, blogs, docs, and more [Alt+S] | ☰ | 🔍 | ⓘ

AWS Glue

Data catalog

Tables

Tables
Connections
Crawlers
Classifiers
Schema registries
Schemas
Settings

ETL

AWS Glue Studio
Jobs New
Jobs (legacy)
ML Transforms
Blueprints
Workflows
Triggers
Dev endpoints
Notebooks
Notebooks (legacy)
Interactive Session

Feedback Looking for language selection? Find it in the new [Unified Settings](#)

Tables **ethereum_collection** Last updated 28 Jul 2022 01:20 PM Table Version (Current version)

ethereum_collection

Description: ethereum_dataset
Classification: csv
Location: s3://ethvsclassic/ethereum_collection/
Connection: No
Last updated: Thu Jul 28 13:20:24 GMT+545 2022
Input format: org.apache.hadoop.mapred.TextInputFormat
Output format: org.apache.hadoop.hive.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib: org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
Serde parameters: field.delim: ,
skip.header.line.count: 1
sizeKey: 79561789
objectCount: 2
UPDATED_BY_CRAWLER: crawler-ethereum
Table properties: CrawlerSchemaSerializerVersion: 1.0
recordCount: 2273193
averageRecordSize: 35
CrawlerSchemaDeserializerVersion: 1.0
compressionType: none
columnsOrdered: true
areColumnsQuoted: false
delimiter: ,
typeOfData: file

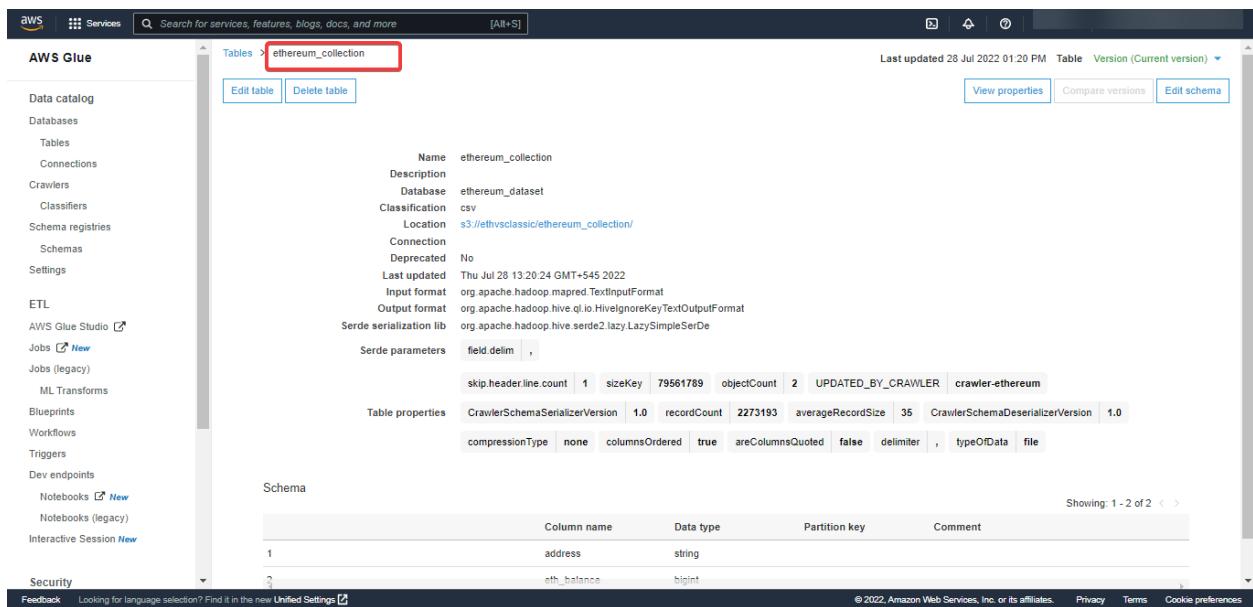
Table properties: CrawlerSchemaSerializerVersion: 1.0
recordCount: 2273193
averageRecordSize: 35
CrawlerSchemaDeserializerVersion: 1.0
compressionType: none
columnsOrdered: true
areColumnsQuoted: false
delimiter: ,
typeOfData: file

Schema

	Column name	Data type	Partition key	Comment
1	address	string		
2	eth_balance	bigint		

Showing: 1 - 2 < >

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences



SQL query on Athena Editor

- Go to AWS Athena query dashboard and run given query

The screenshot shows the AWS Athena Editor interface. On the left, there's a sidebar with 'Data' selected, showing 'Tables (1)' and 'ethereum'. The main area has a 'Query 1' tab open with the following SQL query:

```
1 select * from "ethereum_dataset"."ethereum" limit 10;
```

Below the query, the status bar shows 'SQL Ln 1, Col 43'. At the bottom of the editor, there are buttons for 'Run again', 'Explain', 'Cancel', 'Save', 'Clear', and 'Create'. The results section shows the query completed successfully with a green bar. It displays the following metrics: Time in queue: 89 ms, Run time: 636 ms, Data scanned: 1.70 MB.

- The Result is shown below.

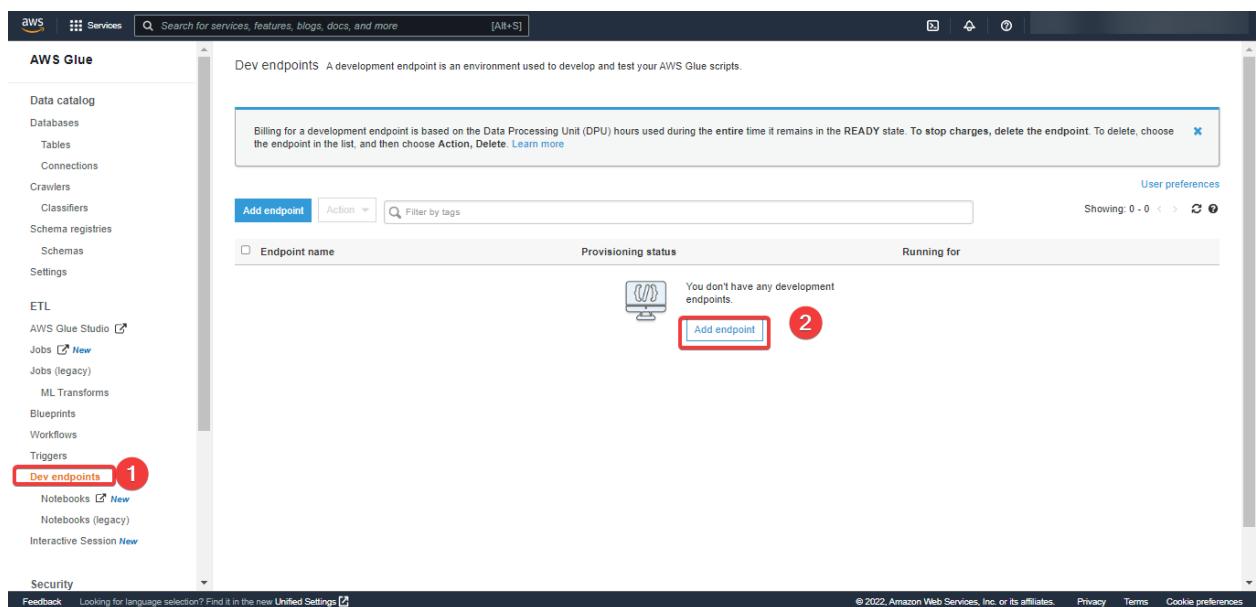
The screenshot shows the results of the executed SQL query. The results are displayed in a table with columns '#', 'address', and 'eth_balance'. There are 10 rows of data, each with a unique address and an eth_balance of 0. The table is highlighted with a red border.

#	address	eth_balance
1	0xf4c4df3d4ca5141c4895af79aa56d8d3ef5c72a9	0
2	0xb474cc8dc8e0b93b1e0696d8a1452cd402eebef5	0
3	0x447d9b46b4e52dc70aa146c5a3e96f23b3a54bb1	0
4	0x04f712d79369011097c3df36ec355da5b940a59c	0
5	0x1e67cace67b3c5d1ad0aeff72f69d83972916d22	0
6	0xdd37a8dff17407a8a4d8b97667e3e4f19d60973e	0
7	0x484382d908db951fba63a047b377e926072c6636	0
8	0x82c90688fa8fcbeb8166a8aacfe7a9c2465a9b1b	0

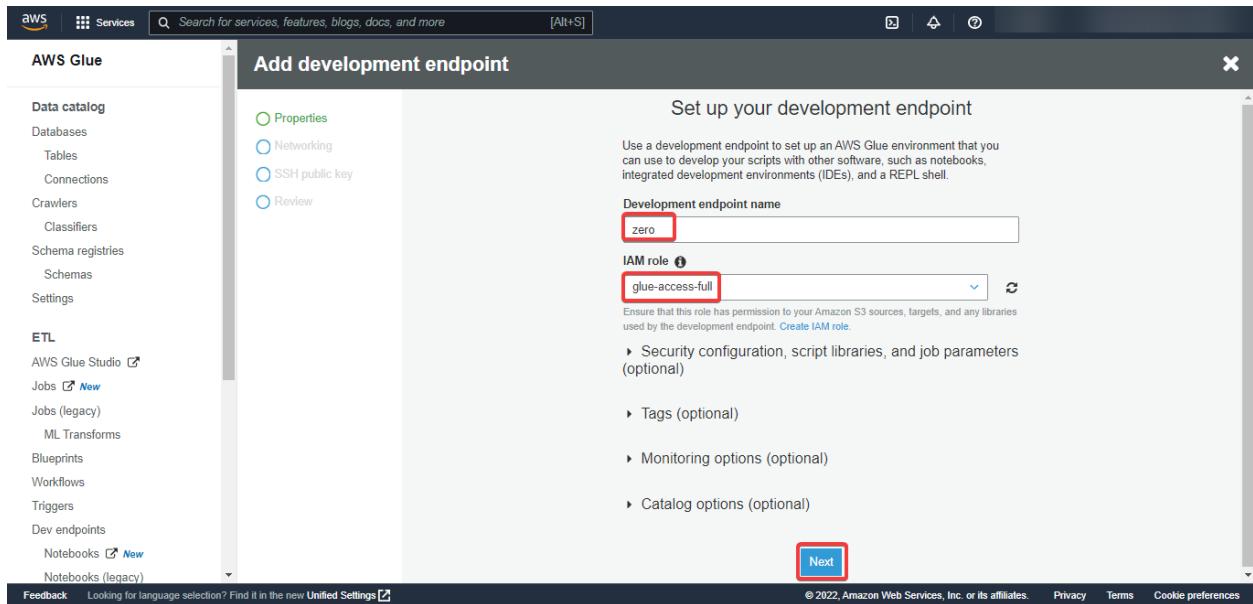
Run Notebook on AWS Glue

Add Dev Endpoints

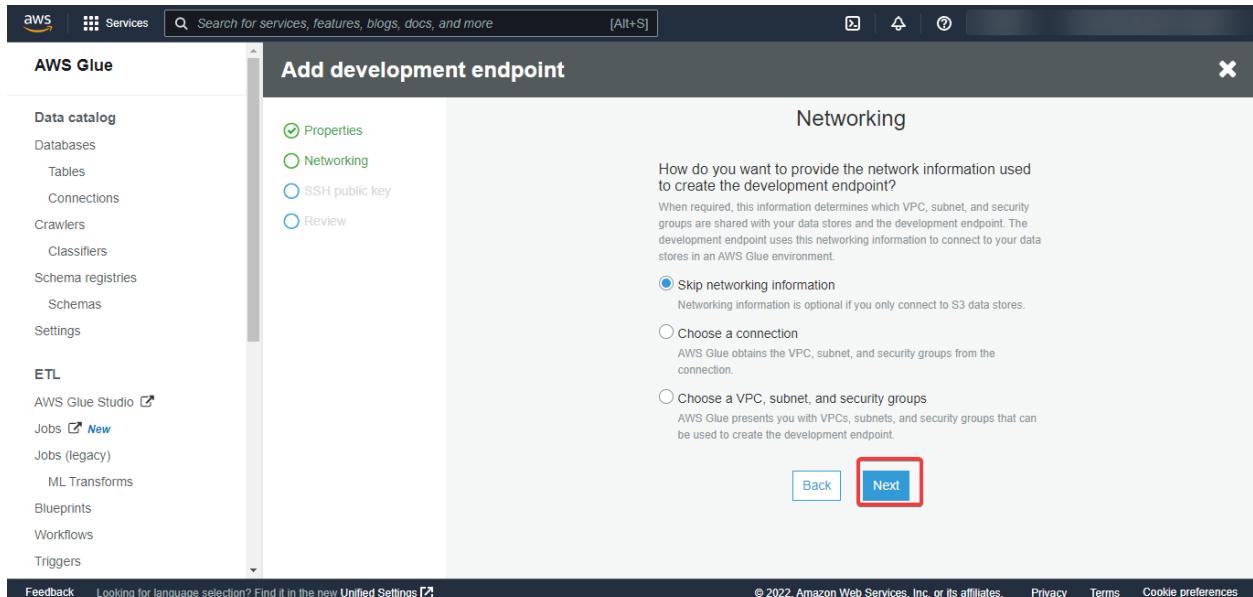
- Go to **AWS Glue**.
- Select **Dev endpoints** on left side of Data catalog
- **Add endpoint**



- Name **Development endpoint name** zero
- Create or choose **IAM role** that access Glue : glue-access-full
- Click **Next.**



- Click Next.



- Click Next.

The screenshot shows the 'Add development endpoint' wizard in the AWS Glue console. The left sidebar lists various AWS services like Data catalog, ETL, and AWS Glue Studio. The main panel is titled 'Add development endpoint' and has a sub-section 'Add an SSH public key (Optional)'. It includes instructions for providing an SSH public key for access from a REPL, IDE, or local notebook client. A 'Public key contents' field with a 'Upload SSH public key' button is present. The navigation bar at the bottom includes 'Back' and 'Next' buttons, with 'Next' being highlighted by a red box.

Once review, Click **Finish**.

The screenshot shows the 'Review' step of the 'Add development endpoint' wizard. It displays three summary sections: 'Dev endpoint properties' (Name: zero, Tags: -), 'Networking' (VPC, Subnet, Security groups), and 'SSH public key' (Public key contents). At the bottom, there are 'Back' and 'Finish' buttons, with 'Finish' being highlighted by a red box.

The screenshot shows the AWS Glue console interface. On the left, there's a navigation sidebar with sections like Data catalog, ETL, and Dev endpoints. The main area displays a table of endpoints. A message at the top states: "Billing for a development endpoint is based on the Data Processing Unit (DPU) hours used during the entire time it remains in the READY state. To stop charges, delete the endpoint. To delete, choose the endpoint in the list, and then choose Action, Delete. Learn more". Below this, another message says: "Development endpoint zero is provisioning. When ready, choose the development endpoint and create a notebook to test your scripts. Either choose Action, Create SageMaker notebook or Action, Create Zeppelin notebook server. Learn more". The table has columns for Endpoint name, Provisioning status, and Running for. One row is highlighted with a red box, showing "zero" in the Endpoint name column and "PROVISIONING" in the Provisioning status column.

Now, your endpoint is ready.

This screenshot is from the same AWS Glue console session after the provisioning process has completed. The main message at the top now says: "Development endpoint zero is ready. You can create a notebook server at this endpoint to test AWS Glue scripts. To create a notebook server, choose the endpoint, then choose Action, Create notebook server. Learn more". The table in the center shows the endpoint "zero" with a red box around the "READY" status in the "Provisioning status" column. The "Running for" column shows "5m".

Create SageMaker notebook

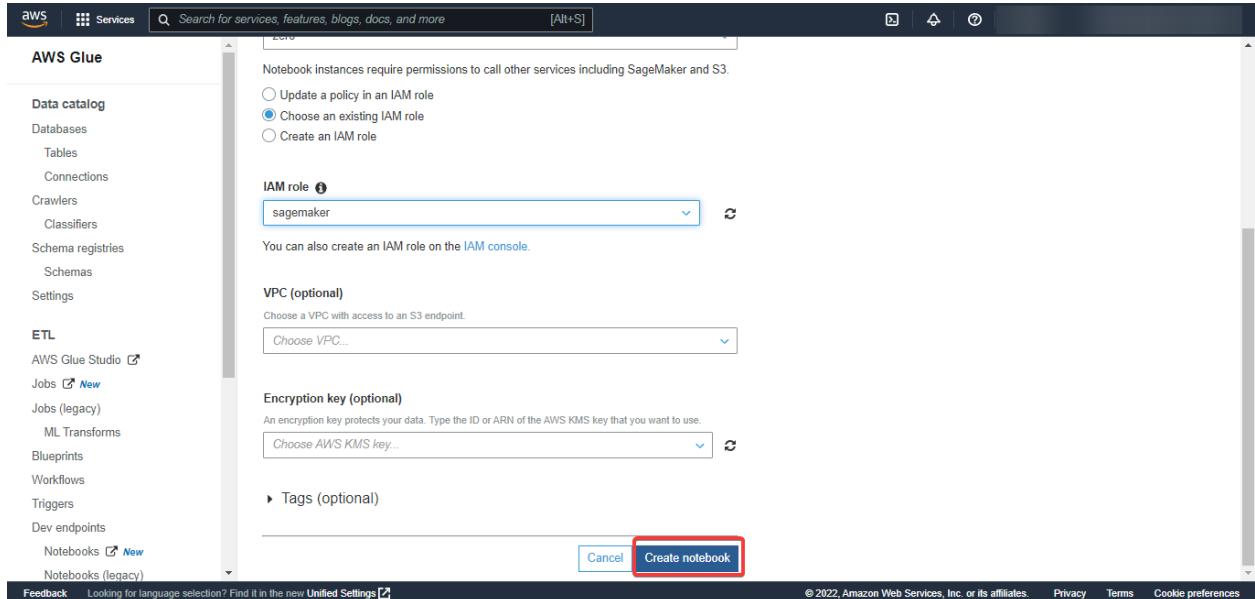
- Select your endpoint zero and click **create SageMaker notebook**.

The screenshot shows the AWS Glue console with the 'Data catalog' section on the left. In the main area, there is a message: 'Development endpoint **zero** is ready. You can create a notebook server at this endpoint to test AWS Glue scripts. To create a notebook server, choose the endpoint, then choose Action, **Create notebook server**. Learn more.' Below this, a dropdown menu for 'Endpoint name' is open, showing 'zero' selected. A sub-menu is displayed with the option 'Create SageMaker notebook' highlighted and surrounded by a red box.

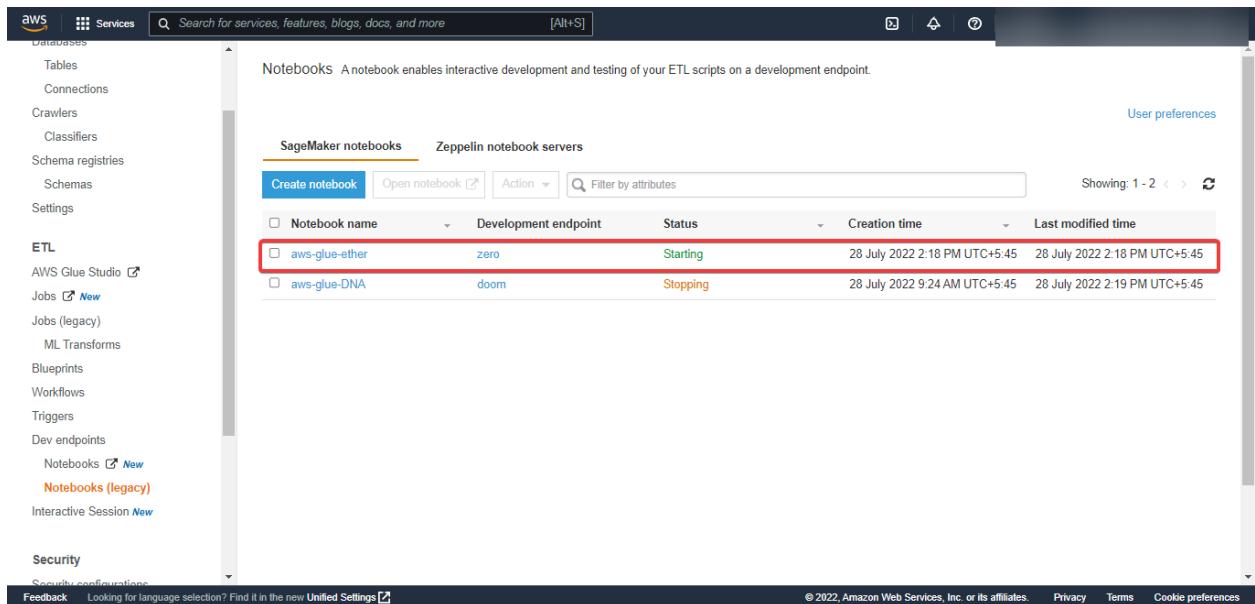
- Give Notebook name **aws-glue-ether**
- **Attach to development endpoint** zero as default we recently created.
- Choose or create IAM sagemaker role for glue. Choose an **existing IAM role**.

The screenshot shows the 'Create notebook' wizard in the AWS Glue console. The 'Notebooks > Create notebook' step is selected. The 'Create and configure a notebook' section includes fields for 'Notebook name' (set to 'aws-glue-ether'), 'Attach to development endpoint' (set to 'zero'), and 'IAM role' (set to 'sagemaker'). The 'Choose an existing IAM role' radio button is selected and highlighted with a red box. Other options like 'Update a policy in an IAM role' and 'Create an IAM role' are also shown.

- Keep other options default
- Click **Create notebook**



- Your aws-glue-ether notebook status is running.



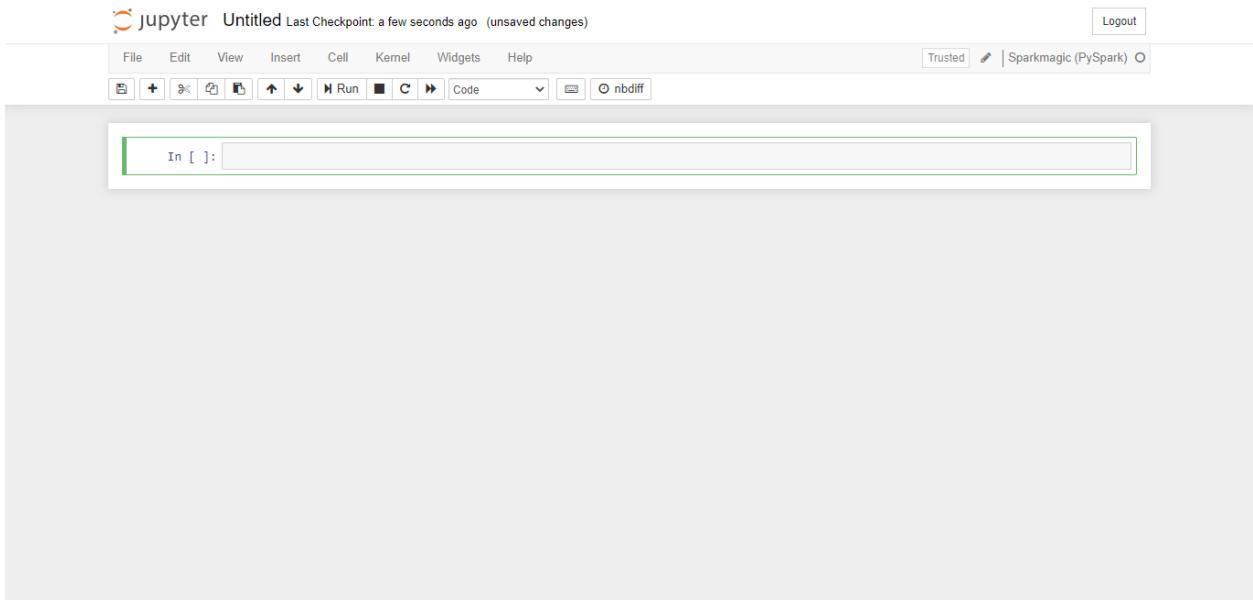
- Select your notebook aws-glue-ether and **open notebook**.

The screenshot shows the AWS SageMaker console interface. On the left, there's a sidebar with various ETL-related options like Tables, Connections, Crawlers, Classifiers, Schema registries, Schemas, Settings, AWS Glue Studio, Jobs, Jobs (legacy), ML Transforms, Blueprints, Workflows, Triggers, Dev endpoints, Notebooks, and Interactive Session. The 'Notebooks' section is currently selected. In the main area, there's a search bar at the top followed by two tabs: 'SageMaker notebooks' (selected) and 'Zeppelin notebook servers'. Below the tabs is a button labeled 'Create notebook' and another labeled 'Open notebook' (which has a red circle '2' over it). There's also a 'Action' dropdown and a 'Filter by attributes' search bar. A table lists the notebooks with columns: Notebook name, Development endpoint, Status, Creation time, and Last modified time. Two entries are shown: 'aws-glue-ether' (Status: Ready) and 'aws-glue-DNA' (Status: Stopped). Both entries have red circles around them; 'aws-glue-ether' has a red circle '1' over it. At the bottom of the page, there are links for Feedback, Unified Settings, Privacy, Terms, and Cookie preferences.

- Jupyter notebook is opened. Click **New** and Select **Pyspark**.

The screenshot shows the Jupyter Notebook interface. At the top, there's a toolbar with 'Open JupyterLab', 'Quit', and 'Logout'. Below the toolbar, there's a navigation bar with 'Files', 'Running', 'Clusters', 'SageMaker Examples', and 'Conda'. Underneath the navigation bar, there's a message 'Select items to perform actions on them.' followed by a file list showing '0' files and a folder named 'Glue Examples'. To the right of the file list, there's a 'New' button with a dropdown menu. The dropdown menu is open, showing a list of kernel options. One option, 'Sparkmagic (PySpark)', is highlighted with a red box. Other options listed include R, Sparkmagic (Spark), Sparkmagic (SparkR), and several Conda-based kernels like conda_amazone_mxnet_p27, conda_amazone_mxnet_p36, etc. At the bottom of the screen, there's a URL bar with the address 'https://aws-glue-ether.notebook.ap-south-1.sagemaker.aws/tree?#'

- Blank jupyter notebook will open with PySpark



Spark Session is successfully installed.

The screenshot shows a Jupyter Notebook cell with the following Python code:

```
In [1]: import sys
from awsglue.transforms import *
from awsglue.utils import getResolvedOptions
from pyspark.context import SparkContext
from awsglue.context import GlueContext
from awsglue.job import Job

glueContext = GlueContext(SparkContext.getOrCreate())
Starting Spark application
```

Below the code, a table displays the current Spark session details:

ID	YARN Application ID	Kind	State	Spark UI	Driver log	User	Current session?
1	application_1658996465043_0002	pyspark	idle	Link	Link	None	✓

Text below the table indicates: "SparkSession available as 'spark'."

At the bottom of the cell is another empty 'In []:' input field.

- Add Database **ethereum_dataset** and table name **ethereum_collection** from AWS Glue

The screenshot shows the AWS Glue Data Catalog interface. On the left, there's a sidebar with 'Data catalog' sections for Databases, Tables, Connections, Crawlers, Classifiers, Schema registries, Schemas, Settings, ETL, AWS Glue Studio, Jobs, ML Transforms, Blueprints, Workflows, and Triggers. The main area shows a table named 'ethereum_collection'. The table details are as follows:

Name	ethereum collection
Description	
Database	ethereum_dataset
Classification	CSV
Location	s3://ethvsclassic/ethereum_collection/
Connection	
Deprecated	No
Last updated	Thu Jul 28 13:20:24 GMT+545 2022
Input format	org.apache.hadoop.mapred.TextInputFormat
Output format	org.apache.hadoop.hive.io.HiveIgnoreKeyTextOutputFormat
Serde serialization lib	org.apache.hadoop.hive.serde2.lazy.LazySimpleSerDe
Serde parameters	field.delim , skip.header.line.count 1 sizeKey 79561789 objectCount 2
Table properties	UPDATED_BY_CRAWLER crawler-ethereum CrawlerSchemaSerializerVersion 1.0 recordCount 2273193 averageRecordSize 35 CrawlerSchemaDeserializerVersion 1.0 compressionType none columnsOrdered true areColumnsQuoted false delimiter , typeOfData file

```
In [2]: ethDF = glueContext.create_dynamic_frame.from_catalog(
    database="ethereum_dataset",
    table_name="ethereum_collection")

In [4]: ethDF.printSchema()

root
 |-- address: string
 |-- eth_balance: choice
 |   |-- long
 |   |-- string

In [5]: glueContext.write_dynamic_frame.from_options(ethDF, connection_type = "s3",
    connection_options = {"path": "s3://ethvsclassic/aws-glue-output/"}, format = "json")
    <awsglue.dynamicframe.DynamicFrame object at 0x7f0b5dc70358>

In [ ]:
```

Once completed **save** your notebook.

- Your **output** is also stored in **S3 bucket**.

S | Services | Search for services, features, blogs, docs, and more | [Alt+S]

Amazon S3 > Buckets > ethysclassic > aws-glue-output/

aws-glue-output/

Objects | Properties

Objects (2)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory [to get a list of all objects in your bucket](#). For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Actions ▾ | **Create folder**

Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
run-165899035026-part-r-00000	-	July 28, 2022, 14:49:06 (UTC+0:45)	61.4 MB	Standard
run-165899035026-part-r-00001	-	July 28, 2022, 14:49:06 (UTC+0:45)	61.4 MB	Standard

AWS Marketplace for S3 | Feedback | Looking for language selection? Find it in the new [Unified Settings](#) | © 2022, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences

run-165899035026-part-r-00000 - Notepad

```
[{"address": "0xe141cff6024f4b2e7babd64d958b522082b378a8", "eth_balance": 0}, {"address": "0x4759e4a499f296f8a3c9a05da317002ec083558", "eth_balance": 0}, {"address": "0xa3de13deaae9bf7e269230a719669f16799557", "eth_balance": 0}, {"address": "0x69288681d885d415dbe0f1cd51e76e1a3b5511d2", "eth_balance": 0}, {"address": "0x3115d371a504f790a80198de20ca9abeea1d64f", "eth_balance": 0}, {"address": "0x9e9c8d7d94d4db675321c4bdada1d4138ab331e0", "eth_balance": 0}, {"address": "0xceef2aac10c2051cbdac0844b9c01e509c0f3f", "eth_balance": 0}, {"address": "0x9098926be23592051ce479e06250ad741d52d16", "eth_balance": 0}, {"address": "0xc473cb3815aeb10587fdaded3430122a4777c8ca", "eth_balance": 0}, {"address": "0x3f821a8b122a8a2d04e358b1a456e2a5d10f8a52", "eth_balance": 0}, {"address": "0xb00553028f0604c3342a74287eef433213dalbb", "eth_balance": 0}, {"address": "0x1c13bdc0a3b7a63b1ac57d7c1863d05cd9a813d", "eth_balance": 0}, {"address": "0x482ecbaed85b743bd4742106b2a9abc43b41fbcc", "eth_balance": 0}, {"address": "0x5e9d201d6f7c866e00f6f46d8f326923157dff", "eth_balance": 0}, {"address": "0x16f76d5b2a1828210ac54bd0ed1a714f62dbb", "eth_balance": 0}, {"address": "0x8dfefdcfdbe7c739948310f886928c72f0d6ee", "eth_balance": 0}, {"address": "0xda138810d30095fc02c2840fbecd4071ba8345", "eth_balance": 0}, {"address": "0x03b31dd562a9284d48d40681561c2b60fd80f8202", "eth_balance": 0}, {"address": "0xba931bd3b5e3d46abe3d29174d3f9fd150f82fc14", "eth_balance": 0}, {"address": "0x557c977f566b35f1dc7db1fe72bf7eef40fde", "eth_balance": 0}, {"address": "0xeaade137dc17582953b3c84094cf21a0dcda7111", "eth_balance": 0}, {"address": "0x38e02839d7e42c0a07cdcff56819ee9fd4ee9cc", "eth_balance": 0}, {"address": "0xb610e9b63e88510de56f7954732d449ff4170e", "eth_balance": 0}, {"address": "0x5dfc8f6780685c0f069a0de861b984fb0d0696f13", "eth_balance": 0}, {"address": "0x330f8a9769a4e9fc6241df0d7725dc921cd6e2", "eth_balance": 0}, {"address": "0x32b2eed0b60ee39d114d9d1e56c1ef5fbceb9e0", "eth_balance": 0}, {"address": "0xa0d24f024eafea8536154220c7979770ba810dd61", "eth_balance": 0}, {"address": "0x529a80c721ff580b397372301c8b533c6f57584a", "eth_balance": 0}, {"address": "0x0f901f2461ed41ec22d97b716a828dd87db50", "eth_balance": 0}, {"address": "0xc9a1a8750de59b5ca0a2d5fb0ff6d7831f67448", "eth_balance": 0}, {"address": "0x8db68de31f30979d9ee6102bccb0bc9aa236f5c", "eth_balance": 0}, {"address": "0x8a16814abbf0d47a6a6b26793d09fd79dd22e74", "eth_balance": 0}, {"address": "0x8065a9b300731ea3ccfc4bd5a2475a9639b8d18c", "eth_balance": 0}, {"address": "0x644da483087074ca3c580190524af8cdf103f", "eth_balance": 0}, {"address": "0x1646b4340c8a87d1fce2be0619bbc5f645621206", "eth_balance": 0}, {"address": "0xe5ea973b0452543b1327a0f148b4d49eb998f86", "eth_balance": 0}]
```

Ln 1, Col 1 | 100% | Unix (LF) | UTF-8

jupyter aws-sagemaker-notebook-glue Last Checkpoint: a few seconds ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Sparkmagic (PySpark) ○

In [1]:

```
import sys
from awsglue.transforms import *
from awsglue.utils import getResolvedOptions
from pyspark.context import SparkContext
from awsglue.context import GlueContext
from awsglue.job import Job

glueContext = GlueContext(SparkContext.getOrCreate())
```

Starting Spark application

ID	YARN Application ID	Kind	State	Spark UI	Driver log	User	Current session?
1	application_1658000446042_0002	pyspark	idle	Link	Link	None	✓

SparkSession available as 'spark'.

In [2]:

```
ethDF = glueContext.create_dynamic_frame.from_catalog(
    database="ethereum_dataset",
    table_name="ethereum_collection")
```

In [4]:

```
ethDF.printSchema()
```

```
root
 |-- address: string
 |-- eth_balance: choice
 |   |-- long
 |   |-- string
```

In [5]:

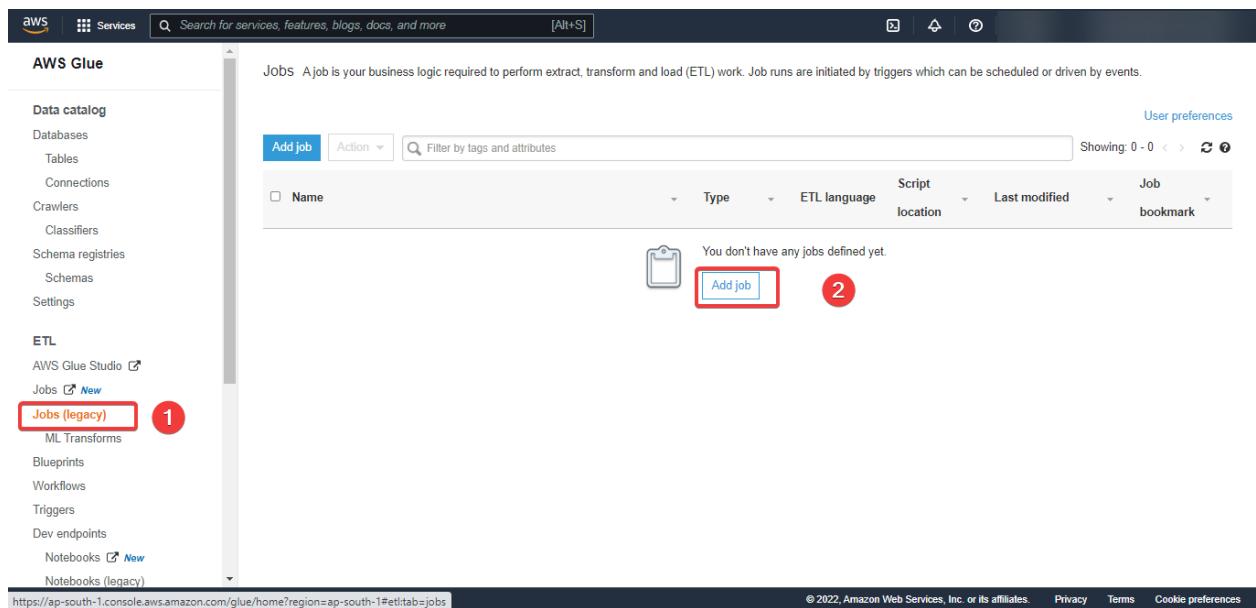
```
glueContext.write_dynamic_frame.from_options(ethDF, connection_type = "s3",
                                             connection_options = {"path": "s3://ethvsclassic/aws-glue-output/"}, format = "json")
```

<awsglue.dynamicframe.DynamicFrame object at 0x7f085dc70358>

In []:

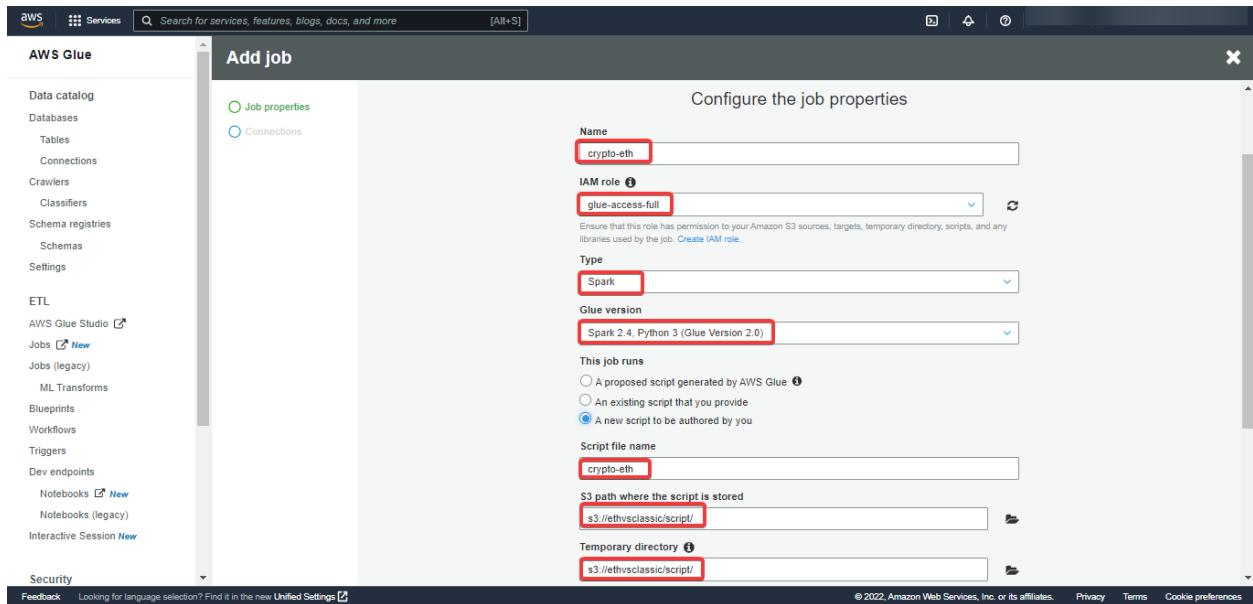
Run PySpark Job in AWS Glue

- Go to **AWS Glue** on AWS console.
- click **jobs(legacy)** on the ETL on the left side of **AWS Glue**
- Click Add job.

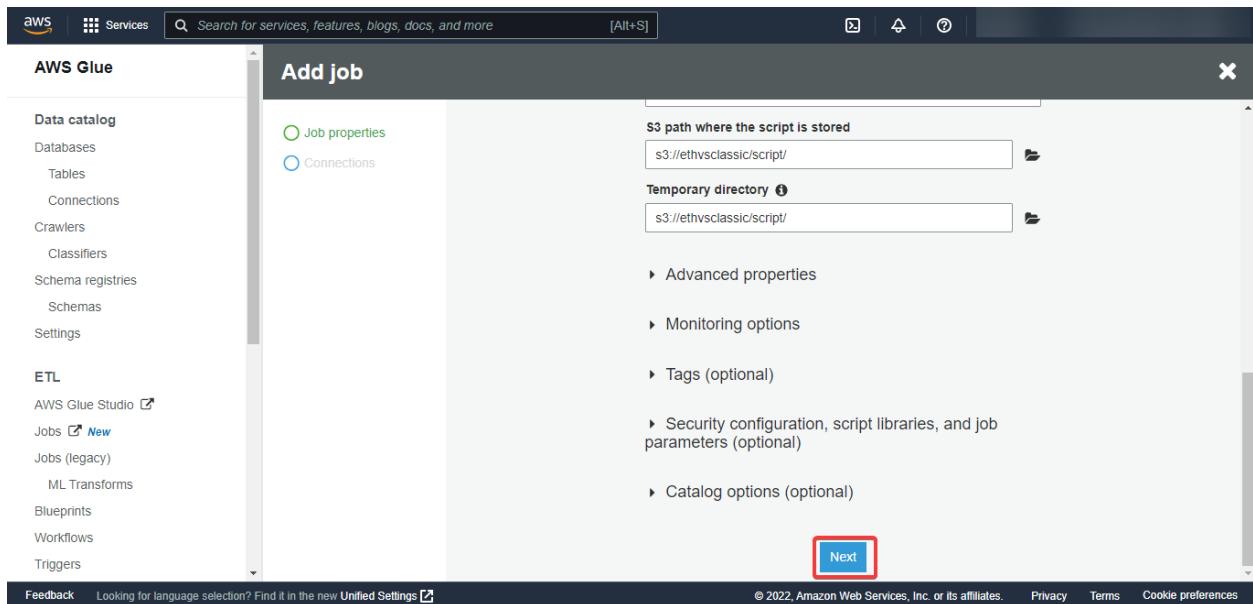


On the next screen,

- Name Job crypto-eth
- Choose **IAM role** glue-access-full
- Type **Spark with Spark 2. Python 3 Version**
- Select a **new script to be authorized by you.**
- Input script file name crypto-eth as default job.
- Select S3 path where you want to store your script.
S3://ethvsclassic/script/



- Click Next



- Click save job and edit script

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Copy paste the following code and click on the **Save** button. There is a bucket location **s3://ethvsclassic/glue-job/** mentioned in the script as the target location for the data. If you created bucket with a different name then use that bucket name.

```

1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 glueContext = GlueContext(SparkContext.getOrCreate())
9
10 etherDF = glueContext.create_dynamic_frame.from_catalog(
11     database="ethereum_dataset",
12     table_name="ethereum_collection")
13
14
15 glueContext.write_dynamic_frame.from_options(etherDF, connection_type = "s3", connection_options = {"path": "s3://ethvsclassic/glue-job/"})
    
```

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

The job is saved. Click on the X button in the right top corner of the job definition to close the job definition.

Job: crypto-eth

```

1 import sys
2 from awsglue.transforms import *
3 from awsglue.utils import getResolvedOptions
4 from pyspark.context import SparkContext
5 from awsglue.context import GlueContext
6 from awsglue.job import Job
7
8 glueContext = GlueContext(SparkContext.getOrCreate())
9
10 etherDF = glueContext.create_dynamic_frame.from_catalog(
11     database="ethereum_dataset",
12     table_name="ethereum_collection")
13
14
15 glueContext.write_dynamic_frame.from_options(etherDF, connection_type = "s3", connection_options = {"path": "s3://eth..."}, format="json")
    
```

Logs Schema

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Select your job and **Run job**

AWS Glue

Jobs

A job is your business logic required to perform extract, transform and load (ETL) work. Job runs are initiated by triggers which can be scheduled or driven by events.

Action [Run job](#) [Stop job run](#) [Choose job triggers](#) [Delete](#) [Edit job](#) [Edit script](#) [Reset job bookmark](#) [Create development endpoint](#)

Type	ETL language	Script location	Last modified	Job bookmark
Spark	python	s3://ethvscl...	28 July 2022 3:11 P...	Disable

View run metrics Rewind job bookmark

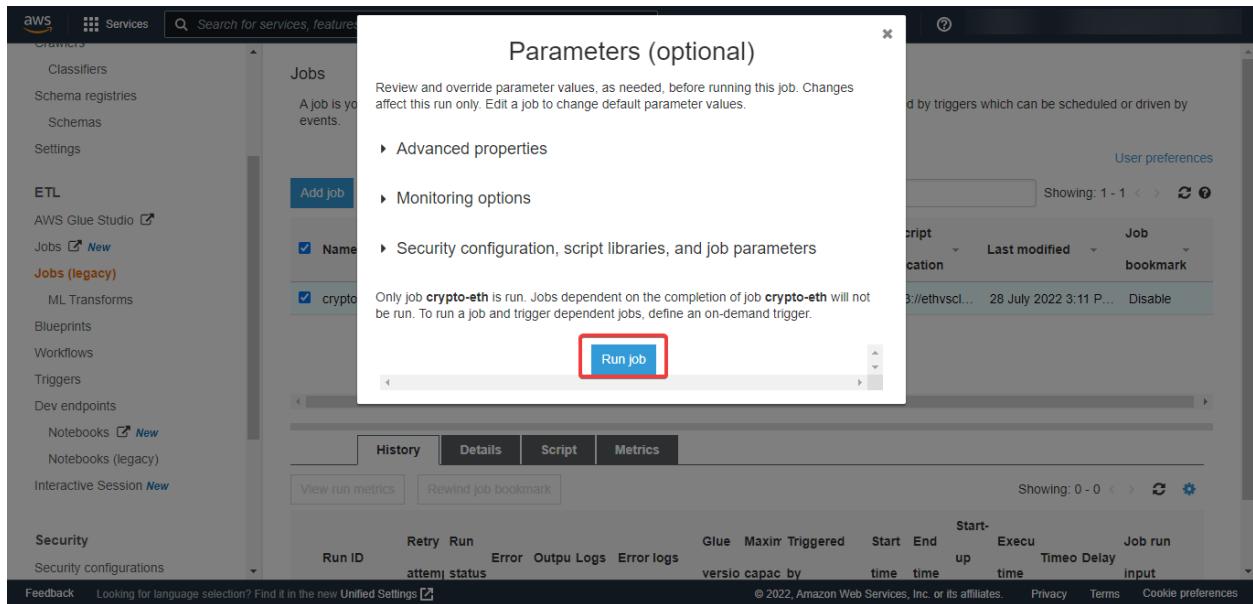
Run ID Retry Run Error Output Logs Error logs Glue Maxima Triggered Start- Execu Job run

attempt status attempt capac by versio time up time time Timeo Delay input

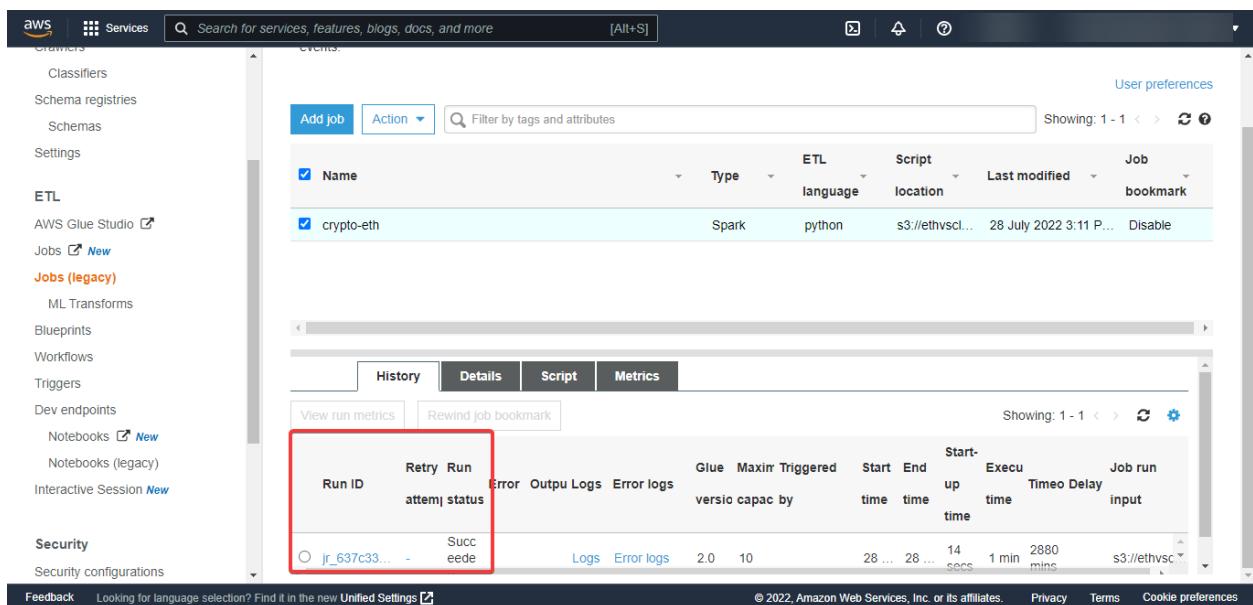
Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Click **Run job** once a dialogue box is opened.



Wait till the job status changes to Succeeded. It might take some time in finishing the job.



Once the job completes, you can see the data created in the S3 location by the job.

Screenshot of the AWS S3 console showing the contents of the 'aws-glue-output' bucket. The 'Objects (6)' section lists six files, all named 'run-[timestamp]-part-r-00000'. Each file is a JSON file with a size of 61.4 MB and a storage class of Standard. The last modified date for all files is July 28, 2022, at 14:49:06 UTC+05:45.

Name	Type	Last modified	Size	Storage class
run-1658999035026-part-r-00000	-	July 28, 2022, 14:49:06 (UTC+05:45)	61.4 MB	Standard
run-1658999035026-part-r-00001	-	July 28, 2022, 14:49:06 (UTC+05:45)	61.4 MB	Standard
run-1659000830852-part-r-00000	-	July 28, 2022, 15:19:10 (UTC+05:45)	61.4 MB	Standard
run-1659000830852-part-r-00001	-	July 28, 2022, 15:19:10 (UTC+05:45)	61.4 MB	Standard
run-1659001287474-part-r-00000	-	July 28, 2022, 15:26:50 (UTC+05:45)	61.4 MB	Standard
run-1659001287474-part-r-00001	-	July 28, 2022, 15:26:50 (UTC+05:45)	61.4 MB	Standard

- Once you download, we can see our json file.

Screenshot of a Notepad window displaying the contents of a JSON file named 'run-1658999035026-part-r-00000'. The file contains a large array of objects, each representing an Ethereum address and its balance. The addresses are long hex strings, and the balances are decimal values.

```

[{"address": "0x7fffeeb67ea660b99052f791be23a0dc4db51e45", "eth_balance": "102856597424177180000"}, {"address": "0x1673a6551f286406a4ddc2b37a6cb9b9e33d07", "eth_balance": "133949413545638074848"}, {"address": "0xa77fabcb65a7360b409051724b75c593114e3bc", "eth_balance": "845097885693493876960"}, {"address": "0x8ed819aecdde02d7a493248bd92f88a13d3ecccb", "eth_balance": "528999999999999900000"}, {"address": "0xeac81d58bd760a2ca543fe1fd5d22abfd4597a", "eth_balance": "609967695493248391776"}, {"address": "0xb2e2fd3c1c884e7df63f78990ed7ca79569857", "eth_balance": "1302038170332428658125"}, {"address": "0xc0c5d4a97ea38c1b3b83d6f0ec713bddbbcb297", "eth_balance": "668262289561868837965"}, {"address": "0xbfa71aba804c2b986d969e175acfda7266cd9c", "eth_balance": "48326422782594365716"}, {"address": "0xc54e016d7f7dcfb1094675d97f9cfaaa800fb543", "eth_balance": "130124927153667926510"}, {"address": "0x2af491ac51687d505284d842e2a05a303b27acd", "eth_balance": "10922696988825137371"}, {"address": "0x1c893a0d22b88478d790db4791b5f84eb2827b5", "eth_balance": "95144108160888209736"}, {"address": "0x7c4a69b3285da18c3bccd76e9b83ac9ba71481", "eth_balance": "50327852027931735112"}, {"address": "0x5e2b4582c76879ba724738709700428609479", "eth_balance": "35334050272219657800"}, {"address": "0xaac41a764f0d4a9149a4aa91555d293fe5", "eth_balance": "150321738364088179253"}, {"address": "0xf25dfa0d6c2291fa3a71571cde5070056586aa8", "eth_balance": "65385628927706617269"}, {"address": "0x0e7399176cae5346aa06df8227e722797da04cc", "eth_balance": "45325575304460785186"}, {"address": "0x34add1b3c6831205d1e650539c84df807bae223", "eth_balance": "40248787540260167344"}, {"address": "0x40736ed5d7ca1fad0f1103927a7f7128d3522", "eth_balance": "57444871658143808816"}, {"address": "0xb069f53ec7011f598e83d5ad98a04eda08c9281a", "eth_balance": "556703896630313087152"}, {"address": "0x9ad0bf6e82c43e015e8a5e273f613f6295b4d59b", "eth_balance": "509052927371687457821"}, {"address": "0xa02a0723d9aec8d744753d52e782fd84b8cae9", "eth_balance": "66246366652062065617"}, {"address": "0x0e673b16ddeab93702671b082d8ac268c448f2c", "eth_balance": "39999580000000004798"}, {"address": "0x87e526f0666a36fdecffaa367b78a2b1d1976f3b", "eth_balance": "4736900229845395134"}, {"address": "0x02ce08c8e603e3d0b33b02dabfc6ae64bf07c76", "eth_balance": "5280128043956047000"}, {"address": "0x7dead318784d688bf3a6018d5220795f6279e3b", "eth_balance": "1906819606699092629528"}, {"address": "0x98b55057e64a451f2835d077e87bc7b03c90f7a", "eth_balance": "150255894183250914399"}, {"address": "0xd427a177fcc3fd1249ac5e0cc03f61081b62", "eth_balance": "572596327981541019321"}, {"address": "0xae6f5ba63967bd4895c914f24d1b14f0cb1b140", "eth_balance": "68804122955370199225"}, {"address": "0xdfef2e99dcab84a1bfff020c03c1e65dd8d9987f", "eth_balance": "157989880000000000000000000000"}, {"address": "0x47bd4832f539a06a47405a5c587b86cb6d3dce", "eth_balance": "14889826013000000000000"}, {"address": "0x8f717ec1552f4c440084fb1a154a81dc003ebdc0", "eth_balance": "1200099497400000000000"}, {"address": "0x49b318f5d2e83104c0d3b32fd11f98405ceo", "eth_balance": "22215956910000000000000"}, {"address": "0x15e348324164ef0890471f6f527451f7a22cf12", "eth_balance": "24099996869274465889000"}, {"address": "0x1d7e528ff5c3c23996fe3c766953e953e8a0e629", "eth_balance": "19259248386889992138052"}, {"address": "0x03377ce556b640103289a6189e1aae63493467", "eth_balance": "17000499085168712539000"}]

```

Using Amazon EMR with AWS Glue Catalog

- Make sure you have already created **IAM role, S3 bucket, glue database, glue cluster**. Follow above procedure

Create IAM Role for Glue

You start with creation of the IAM role which AWS Glue uses for the authorization to call other AWS Services.

1. Login to the AWS Console and select **Mumbai** as the region.
2. Go to the IAM Management console and
3. click on the **Roles** menu in the left and then click on the **Create role** button.

The screenshot shows the AWS IAM Roles page. The left sidebar has 'Identity and Access Management (IAM)' selected (marked with a red circle 1). Under 'Access management', 'Roles' is selected (marked with a red circle 2). At the top right, there is a 'Create role' button (marked with a red circle 3).

Role name	Trusted entities	Last updated
abtesting-lambda-vreq	AWS Service: lambda, and 1 more.	98 min
AccessAnalyzerMonitorServiceRole_HP0M4WFFI7	AWS Service: access-analyzer	-
AmazonAppStreamServiceAccess	AWS Service: appstream	6 hr
AmazonChatBotFullAccess	Identity Provider: cognito-identity.amazonaws.com	▲ 1 hr
AmazonComprehendServiceRole-comprehend	AWS Service: comprehend	100 min
AmazonForecast-ExecutionRole-1601896035407	AWS Service: forecast	101 min
AmazonForecast-ExecutionRole-1601896354684	AWS Service: forecast	101 min
AmazonKendra-sample-s3-role-d9ec45bc-c4ca-4912-8978-0011c89a8067	AWS Service: kendra	56 min

On the next screen, select **Glue** as the service and click on the **Next: Permissions** button.

The screenshot shows the 'Select trusted entity' step of the IAM role creation wizard. The 'AWS service' option is selected and highlighted with a red box. The 'Use case' dropdown also has 'Glue' selected and is highlighted with a red box. The 'Next' button is at the bottom right.

On the next screen, select **PowerUserAccess** as the policy and click on the **Next** button. The exercise is using power user permission but in actual production use it is recommended to use minimum required permission only.

The screenshot shows the 'Add permissions' step of the IAM role creation wizard. The 'PowerUserAccess' policy is selected and highlighted with a red box. The 'Next' button is at the bottom right.

The screenshot shows the AWS IAM 'Create role' wizard at Step 3: Name, review, and create. The left sidebar shows the IAM navigation menu with 'Roles' selected. The main area has three steps: Step 1 (Select trusted entity), Step 2 (Add permissions), and Step 3 (Name, review, and create). Step 3 is active. It contains fields for 'Role name' (set to 'hm-glue-role') and 'Description' (set to 'Allows Glue to call AWS services on your behalf'). Below these are sections for 'Step 1: Select trusted entities' (with a dropdown showing 'Version: "2012-10-17"') and 'Step 2: Add permissions' (which is currently empty). At the bottom right is a 'Create role' button.

- On the next screen, click on the **Next: Review** button.
- On the next screen, type in **hm-glue-role** for the **Role name** and click on the **Create role** button.

The screenshot shows the AWS IAM 'Create role' wizard at Step 2: Add permissions. The left sidebar shows the IAM navigation menu with 'Roles' selected. The main area displays a 'Permissions policy summary' table with one item: 'PowerUserAccess' (Type: AWS managed - job function, Attached as: Permissions policy). Below this is a 'Tags' section with an 'Add tag' button. At the bottom right are 'Cancel', 'Previous', and a red-highlighted 'Create role' button.

- IAM Glue Role is created now.

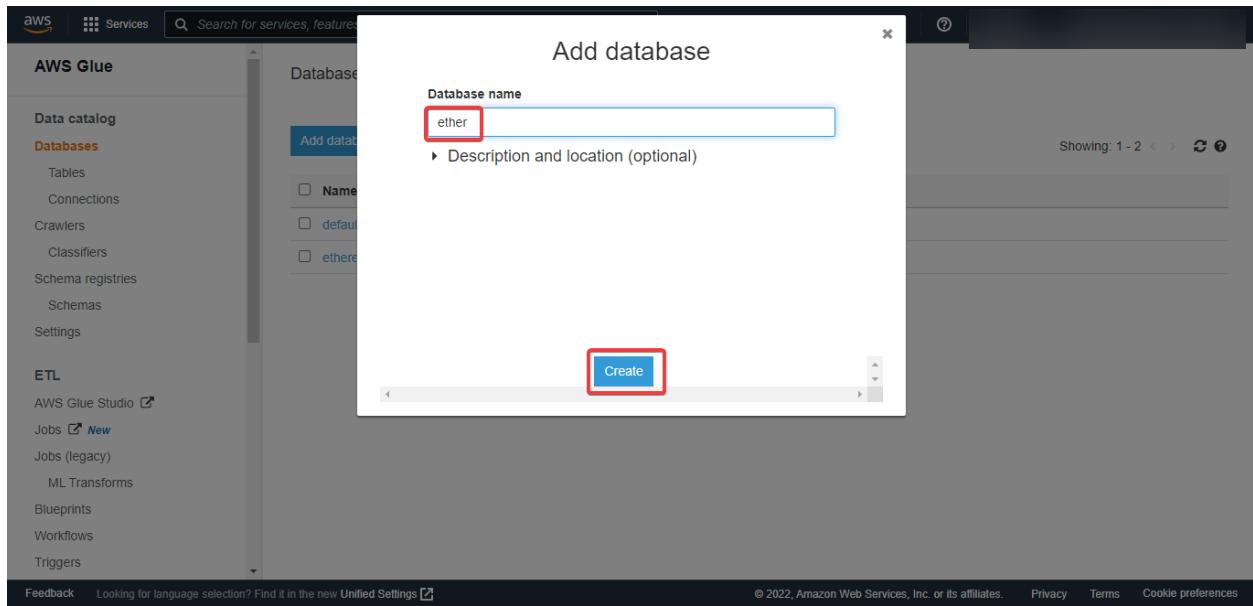
The screenshot shows the AWS IAM Roles page. A green banner at the top indicates that the role 'hm-glue-role' has been created. The main area displays a table of roles, with the 'hm-glue-role' row highlighted by a red box. The table columns include Role name, Trusted entities, and Last activity. Below the table, there are sections for 'Roles Anywhere' and 'Temporary credentials'.

Create Glue database and glue cluster

- Click on Database on Data Catalog and Add database.

The screenshot shows the AWS Glue Data Catalog Databases page. A message box at the top right indicates that the database 'ether' has been successfully deleted. Below this, there is a table with two entries: 'default' and 'ethereum_dataset'. The 'Add database' button is highlighted with a red box. The page also includes a sidebar with ETL and Data catalog sections.

- Name a Database name **ether**
- Click **Create**

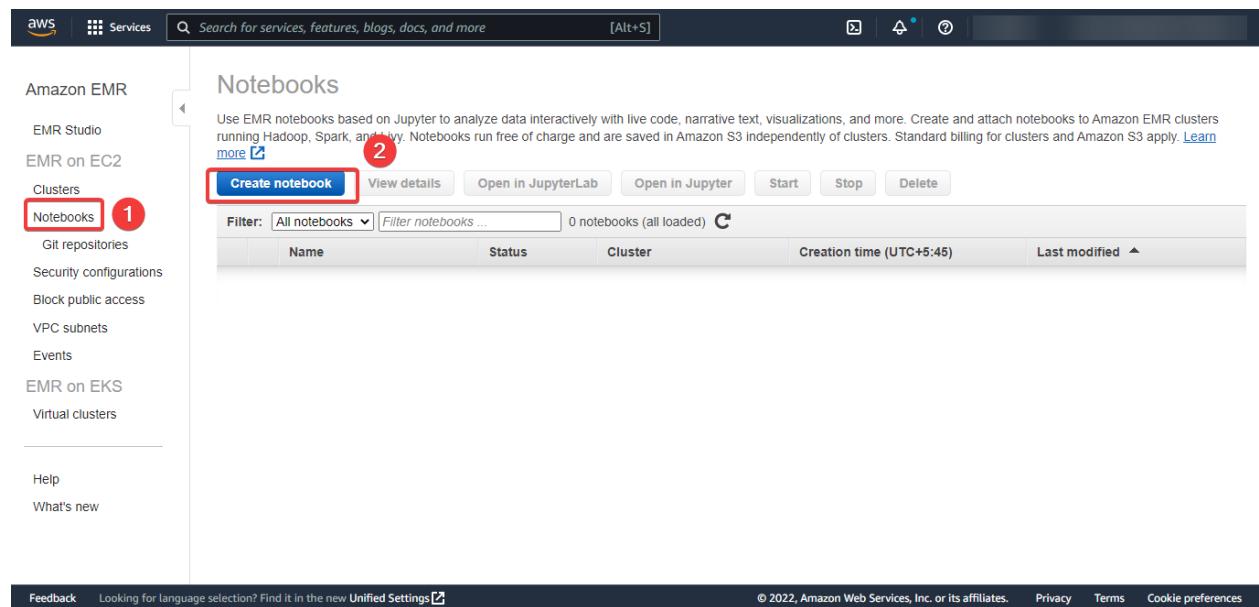


- Then add a crawler follow same process only change IAM role and name as hm-glue-role.
- Make sure you have S3 bucket, glue database, glue cluster as we already created above.

Launch EMR Cluster and Run Jupyter Notebook

You launch EMR cluster which is used to process data using Glue Data Catalog and PySpark code.

1. Go to the EMR Management console and click on the **Create notebook** button.



- Create a **Notebook name** Esther0
- Create a **cluster name** NotebookCluster
- Choose **instance 1 mx5.xlarge size**
- Choose EC2 key pair Proceed without EC2 key

Create notebook

Name and configure your notebook

Name your notebook, choose a cluster or create one, and customize configuration options if desired. [Learn more](#)

Notebook name* Names may only contain alphanumeric characters, hyphens (-), or underscores (_).

Description

256 characters max.

Cluster* Choose an existing cluster Create a cluster [?](#)

Cluster name:

Release: emr-5.36.0

Applications: Hadoop, Spark, Livy, Hive, JupyterEnterpriseGateway

Instance: 1 m5.xlarge

EMR role: [EMR_DefaultRole](#) Use EMR_DefaultRole_V2 [?](#)

EC2 instance profile: [EMR_EC2_DefaultRole](#) [?](#)

EC2 key pair: [?](#)

[Feedback](#) Looking for language selection? Find it in the new [Unified Settings](#) [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

- Select **default security group** and **AWS service role** as **EMR_Notebook_DefaultRole**
- Click **Create notebook**

EC2 instance profile [EMR_EC2_DefaultRole](#) [?](#)

EC2 key pair: [?](#)

Auto-termination Enable auto-termination
Terminate cluster when it is idle after hours minutes

Security groups Use default security groups [?](#) Choose security groups (vpc-697b2001)

AWS service role [?](#)

Notebook location* Choose an S3 location where files for this notebook are saved.
 Use a location that EMR creates [?](#)
 s3://aws-emr-resources-031342435657-ap-south-1/notebooks/
 Choose an existing S3 location in ap-south-1

Git repository Link to a Git repository

Tags [?](#)

* Required [Cancel](#)

[Feedback](#) Looking for language selection? Find it in the new [Unified Settings](#) [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

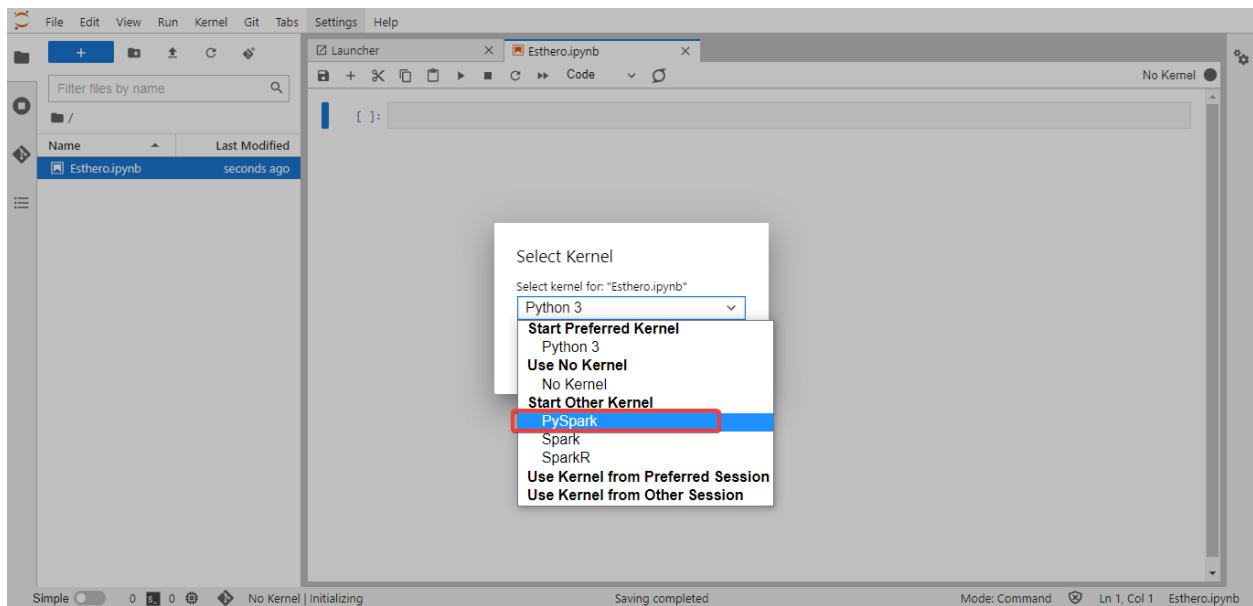
- **Notebook: Esther** is still pending.
- Once successful click Open in **JupyterLab** or **Open in Jupyter**

The screenshot shows the AWS EMR Management console. On the left, there's a sidebar with navigation links like Amazon EMR, EMR Studio, EMR on EC2, Clusters, Notebooks, EMR on EKS, and Help. The main content area displays a notebook named 'Notebook: Esther'. The status is 'Pending' with a note: 'Workspace(notebook) can now be used in local mode. Workspace(notebook) is attaching to cluster j-3JCOE9GMT7Y15.' Below the status are several buttons: 'Open in JupyterLab' (highlighted with a red box), 'Open in Jupyter', 'Stop', and 'Delete'. A 'Notebook' section follows, containing details such as Notebook ID (e-6A80DEP5PNX3VP5D27KIENF1X), Description (--), Last modified (20 seconds ago), Last modified by (...user/hem.gurung), Created on (2022-07-28 16:15 UTC+5:45), Created by (...user/hem.gurung), Service IAM role (EMR_Notebooks_DefaultRole), Notebook tags (creatorUserId = AIDAQOTBHTVEZHSTOABL2), and Notebook location (s3://aws-emr-resources-031342435657-ap-south-1/notebooks/). A 'Cluster' section shows Cluster (NotebookCluster), Cluster Id (j-3JCOE9GMT7Y15), Cluster status (Starting), and Cluster tags (creator = NOTEBOOK_CONSOLE). At the bottom, there are links for Feedback, Unified Settings, and various AWS links.

- You now run PySpark code to process S3 based data in the EMR Cluster using AWS Glue Catalog.
- In the EMR Management console, on the **Esthero** notebook page, click on the **Open in Jupyter** button.

This screenshot is similar to the one above, showing the 'Notebook: Esther' page in the AWS EMR Management console. The 'Open in Jupyter' button is highlighted with a red box. The rest of the interface, including the notebook details and cluster information, is identical to the first screenshot.

It will open Notebook environment in a new browser tab or window. In Jupyter environment, click on **PySpark** option under the **Kernel** menu.



It will open Jupyter notebook IDE. In the cell, copy-paste the following code and run it. The code imports modules for the PySpark.

```
•[1]: import sys
from datetime import datetime

from pyspark.sql import SparkSession
from pyspark.sql.functions import *

Last executed at 2022-07-28 16:29:23 in 36.09s

Starting Spark application
ID          YARN Application ID   Kind  State  Spark UI  Driver log  User  Current session?
0  application_1659004522439_0001  pyspark  idle      Link     Link  None    ✓

SparkSession available as 'spark'.
```

You copy-paste and run the following code to get the spark session.

```
•[2]: spark = SparkSession\
        .builder\
        .appName("SparkETL")\
        .getOrCreate()

Last executed at 2022-07-28 16:30:25 in 1.38s
```

Now you can run your notebook.

IAM ROLE for Glue: Sagemaker

- Go **IAM Role** on aws console
- Select **Roles** on **IAM Dashboard**
- **Create role.**

The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with 'Identity and Access Management (IAM)' selected. Under 'Access management', 'Roles' is also selected. The main area displays a table of roles with columns for 'Role name', 'Trusted entities', and 'Last updated'. A red box highlights the 'Create role' button at the top right of the table header. The table lists several roles, including 'abtesting-lambda-vreq', 'AccessAnalyzerMonitorServiceRole_HP0M4WFFI7', and 'AmazonKendra-sample-s3-role-d9ec45bc-c4ca-4912-8978-0011c89a8067'.

- Select **sagemaker** on **AWS service**.
- Click **Next**.

The screenshot shows the 'Select trusted entity' step in the IAM Role creation wizard. It's Step 1 of 3. The sidebar shows 'Roles' is selected. The main area has three tabs: 'Step 1 Select trusted entity' (selected), 'Step 2 Add permissions', and 'Step 3 Name, review, and create'. Under 'Trusted entity type', a red box highlights the 'AWS service' radio button. Other options include 'AWS account', 'Web identity', 'SAML 2.0 federation', and 'Custom trust policy'. Below these, under 'Use case', 'SageMaker - Execution' is selected, highlighted with a red box. At the bottom right, a red box highlights the 'Next Step' button.

- Click Next.

The screenshot shows the 'Add permissions' step of creating a role. On the left, the navigation menu is visible with 'Roles' selected. The main area shows a table with one policy attached:

Policy name	Type	Attached entities
AmazonSageMaker...	AWS m...	29

Below the table, there's a section for setting a permissions boundary, which is optional. At the bottom right, there are 'Cancel', 'Previous', and 'Next' buttons, with 'Next' being highlighted with a red box.

Create a Role name SAGEMAKER

The screenshot shows the 'Name, review, and create' step of creating a role. The 'Role name' field contains 'SAGEMAKER', which is highlighted with a red box. The 'Description' field contains a placeholder text about allowing SageMaker instances to access S3, ECR, and CloudWatch. At the bottom right, there are 'Edit' and 'Next Step' buttons, with 'Next Step' being highlighted with a red box.

- Click **Create role**.

Identity and Access Management (IAM)

Step 2: Add permissions

Permissions policy summary

Policy name	Type	Attached as
AmazonSageMakerFullAccess	AWS managed	Permissions policy

Tags

Add tags (Optional)

No tags associated with the resource.

Add tag

You can add up to 50 more tags.

Cancel Previous Create role

- Role **SAGEMAKERR** is created.

Identity and Access Management (IAM)

IAM > Roles

Roles (750) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
SAGEMAKERR	AWS Service: sagemaker	-

Roles Anywhere Info

Authenticate your non AWS workloads and securely provide access to AWS services.

Access AWS from your non AWS workloads X.509 Standard Temporary credentials

Manage

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

- Open your recently created sagemaker role **SAGEMAKERR** role.

The screenshot shows the AWS IAM Roles page. The left sidebar is collapsed. The main area displays a table of roles. A search bar at the top right contains the text "SAGEMAKERR". A single row in the table is highlighted with a red border, corresponding to the role name "SAGEMAKERR". The table has columns for "Role name", "Trusted entities", and "Last activity". Below the table, there's a section titled "Roles Anywhere" with three options: "Access AWS from your non AWS workloads", "X.509 Standard", and "Temporary credentials".

- Click Add permission and select Attach policies.

The screenshot shows the "Permissions" tab of the IAM Role details page for "SAGEMAKERR". The "Add permissions" button is highlighted with a red box. The "Attach policies" option under it is also highlighted with a red box. The table below lists one policy: "AmazonSageMakerFullAccess" (AWS managed, provides full access to Amazon SageMaker).

Following above procedure attach other policies such as

AWSGlueConsoleSagemakerNotebookFullAccess,
AmazonS3FullAccess,
AmazonSagemakerFullAccess
AWSGlueConsoleFullAccess

Other permissions policies (Selected 2/1608)			
<input type="text"/> Filter policies by property or policy name and press enter		1 match	Cancel Create policy
<input type="button"/> "AWSGlueConsoleSageMakerNotebookFullAccess" X <input type="button"/> Clear filters			
Policy name	Type	Description	
<input checked="" type="checkbox"/> AWSGlueConsoleSageMakerNotebookFullAccess	AWS managed	Provides full a	<input type="button"/>

<input type="button"/> Cancel <input style="background-color: #0070C0; color: white; font-weight: bold; padding: 2px 10px; border-radius: 5px; border: none; margin-left: 10px;" type="button"/> Attach policies			
<input checked="" type="checkbox"/> AWSGlueConsoleFullAccess	AWS managed	Provides full access to AWS Glue via the A...	<input type="button"/>
<input type="checkbox"/> AwsGlueDataBrewFullAccessPolicy	AWS managed	Provides full access to AWS Glue DataBrew...	<input type="button"/>
<input type="checkbox"/> AWSGlueSchemaRegistryReadOnlyAccess	AWS managed	Provides readonly access to the AWS Glue ...	<input type="button"/>

aws | Services | Search for services, features, blogs, docs, and more [Alt+S] | ? |

Identity and Access Management (IAM)

Dashboard

Access management User groups Roles Policies Identity providers Account settings

Access reports Access analyzer Archive rules Analyzers Settings Credential report Organization activity Service control policies (SCPs)

Permissions Trust relationships Tags Access Advisor Revoke sessions

Permissions policies (4) You can attach up to 10 managed policies.

Simulate Remove Add permissions

Filter policies by property or policy name and press enter

Policy name	Type	Description
<input type="checkbox"/> AmazonS3FullAccess	AWS managed	Provides full access to all buckets
<input type="checkbox"/> AWSGlueConsoleSageMakerNotebookFullAccess	AWS managed	Provides full access to AWS Glue
<input type="checkbox"/> AWSGlueConsoleFullAccess	AWS managed	Provides full access to AWS Glue
<input type="checkbox"/> AmazonSageMakerFullAccess	AWS managed	Provides full access to Amazon Sa

Permissions boundary - (not set)
Set a permissions boundary to control the maximum permissions this role can have.
This is not a common setting but can be used to delegate permission management to

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Create IAM role for Glue: Crawler

- Go to **IAM** dashboard
- Go to **Roles** and **create role**.

The screenshot shows the AWS IAM Roles page. On the left, there's a navigation sidebar with options like Dashboard, Access management, Policies, and Roles (which is highlighted with a red box). The main area displays a table of existing roles, each with a checkbox, Role name, Trusted entities, and Last updated. A search bar and pagination controls are at the top of the table. At the bottom right of the table, there's a prominent blue 'Create role' button.

- Select **Glue** on **AWS service**
- Click **Next**

The screenshot shows the 'Select trusted entity' step in the IAM role creation wizard. The left sidebar shows the 'Roles' section selected. The main area has three tabs: Step 1 (Select trusted entity), Step 2 (Add permissions), and Step 3 (Name, review, and create). Under Step 1, there's a 'Trusted entity type' section with four options: 'AWS service' (selected and highlighted with a red box), 'AWS account', 'Web identity', and 'SAML 2.0 federation'. Below that is a 'Use case' section with 'EC2' and 'Lambda' options. Under 'Common use cases', there's a dropdown menu labeled 'Glue' (highlighted with a red box) and a 'Custom trust policy' option. At the bottom right, there's a 'Cancel' button and a 'Next' button (highlighted with a red box).

- Add **AdministratorAccess** policy.
- Click **Next**

The screenshot shows the AWS IAM 'Create role' wizard. The left sidebar shows the 'Identity and Access Management (IAM)' navigation pane with 'Roles' selected. The main area is titled 'Add permissions'. It shows a list of 'Permissions policies' with one item selected: 'AdministratorAccess'. A red box highlights the 'AdministratorAccess' checkbox. Below the list is a section titled 'Set permissions boundary - optional' with a note about delegating permission management. At the bottom right are 'Cancel', 'Previous', and a blue 'Next' button.

- Create role name `sagemaker-glue`

The screenshot shows the AWS IAM 'Create role' wizard. The left sidebar shows the 'Identity and Access Management (IAM)' navigation pane with 'Roles' selected. The main area is titled 'Name, review, and create'. It shows a 'Role details' section with a 'Role name' field containing 'sagemaker-glue', which is highlighted with a red box. Below it is a 'Description' field with the text 'Allows Glue to call AWS services on your behalf.' At the bottom right are 'Edit' and 'Next Step' buttons.

- Click **create role**.

Identity and Access Management (IAM)

Step 2: Add permissions

Policy name	Type	Attached as
AdministratorAccess	AWS managed - job function	Permissions policy

Add tags (Optional)
Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

Add tag
You can add up to 50 more tags.

Create role

- Sagemaker-IAM-role is created

Identity and Access Management (IAM)

Role sagemaker-gluee created

IAM > Roles

Roles (751)

Role name	Trusted entities	Last activity
sagemaker-gluee	AWS Service: glue	-

Roles Anywhere

Access AWS from your non AWS workloads

X.509 Standard

Temporary credentials

CleanUp

Once you completed your task donot forget to follow this step in order to save from extra fee on using aws services.

- Delete your Glue Job in AWS Glue Console.
- Stop and delete notebook instance AWS Glue Console.
- Delete developer endpoint in AWS Glue Console.
- Delete database in AWS Glue and athena Console.

- Delete s3 bucket in S3 Management Console. If you created bucket with a different name then delete that one.
- Delete IAM Roles your IAM Management Console.