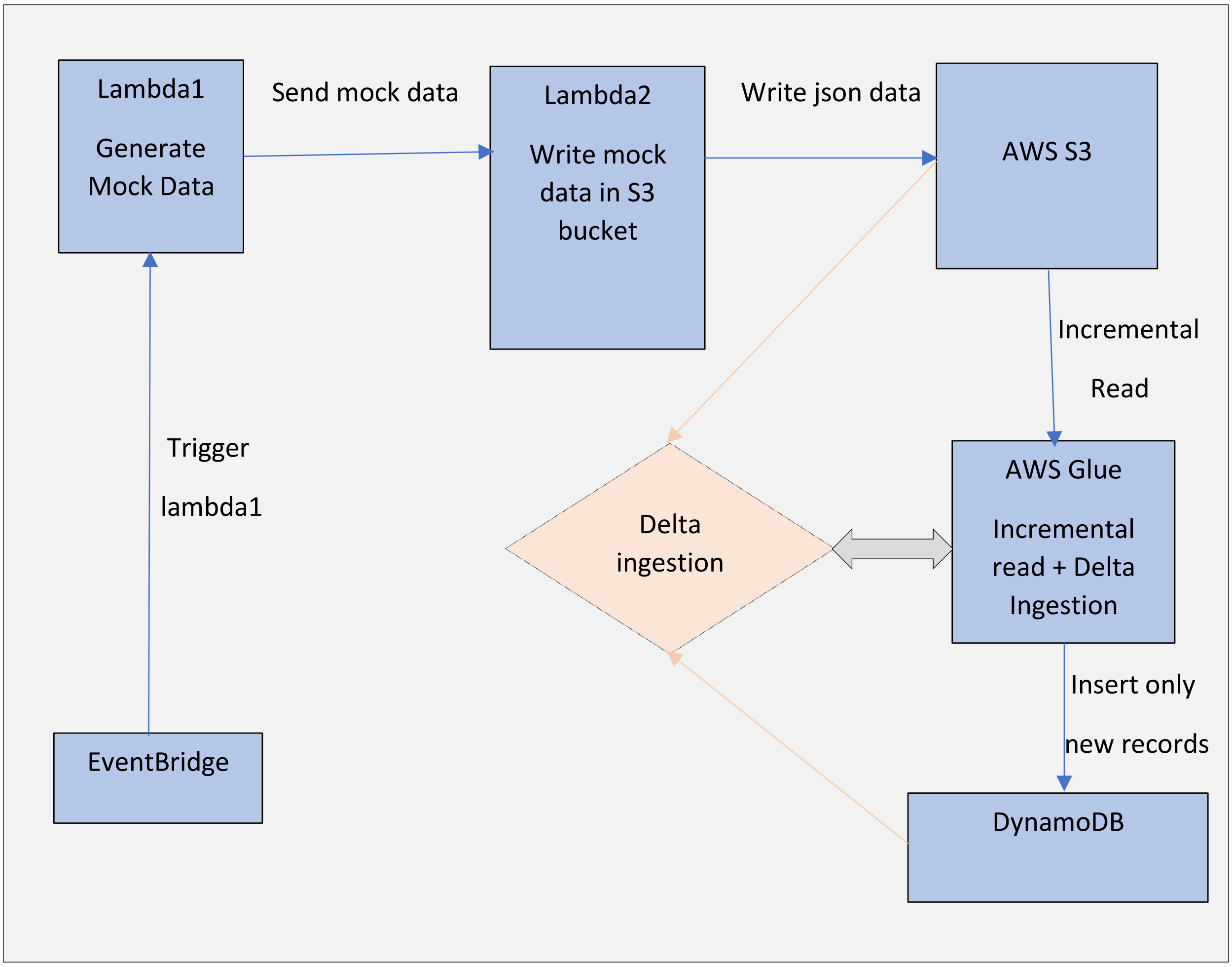
AWS PROJECT 1

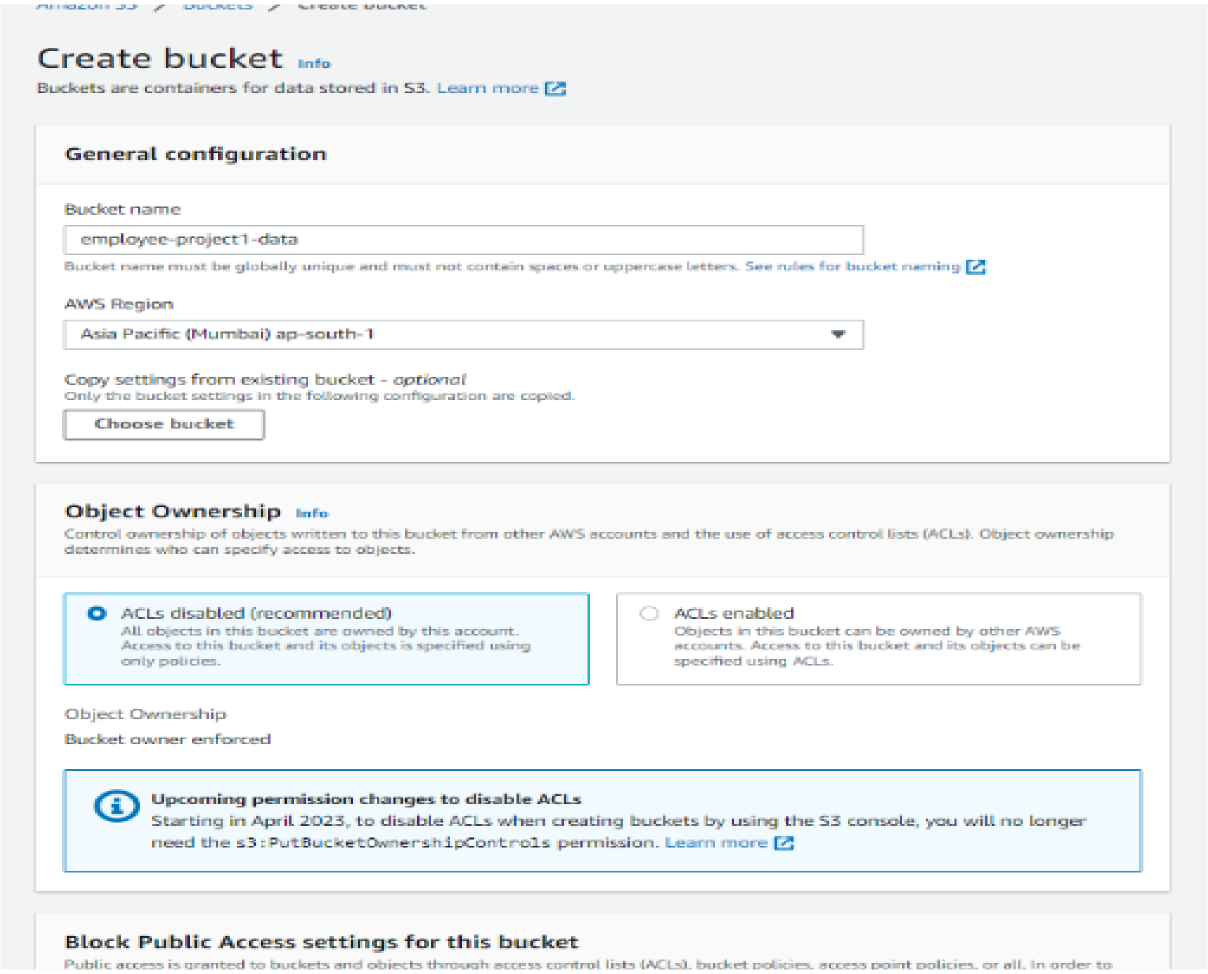
AWS BATCH DATA PIPELINE

Project Architecture:



S3:

Create S3 bucket: employee-project1-data



Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name

employee-project1-data

Bucket name must be globally unique and must not contain spaces or uppercase letters. See rules for bucket naming

AWS Region

Asia Pacific (Mumbai) ap-south-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Object Ownership Info

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.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

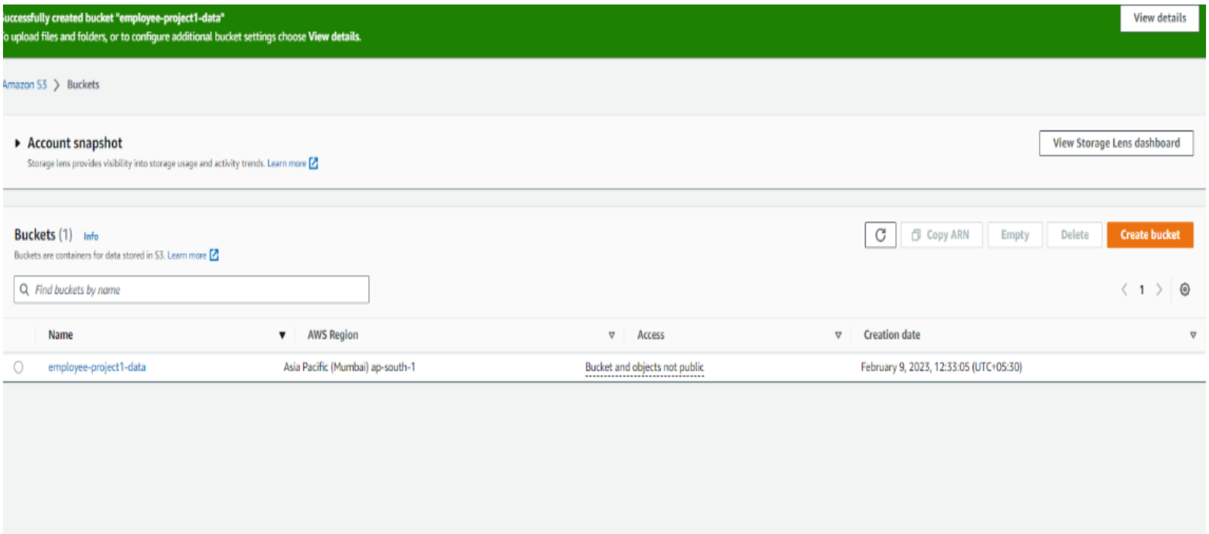
Bucket owner enforced

Upcoming permission changes to disable ACLs

Starting in April 2023, to disable ACLs when creating buckets by using the S3 console, you will no longer need the s3:PutBucketOwnershipControls permission. Learn more

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to



uccessfully created bucket "employee-project1-data"

o upload files and folders, or to configure additional bucket settings choose **View details**.

Amazon S3 > Buckets

Account snapshot

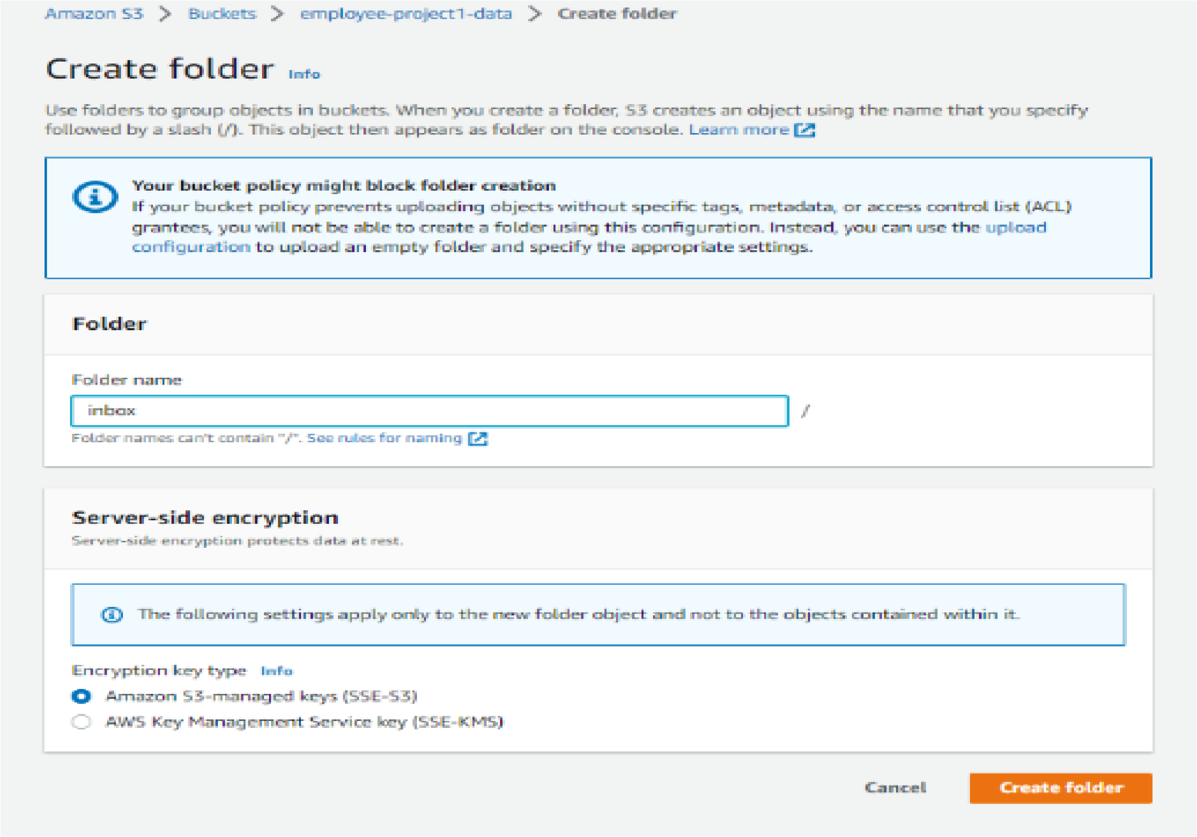
Storage lens provides visibility into storage usage and activity trends. Learn more

Buckets (1) Info

Buckets are containers for data stored in S3. Learn more

	Name	AWS Region	Access	Creation date
○	employee-project1-data	Asia Pacific (Mumbai) ap-south-1	Bucket and objects not public	February 9, 2023, 12:33:05 (UTC+05:30)

Create folder inside bucket: inbox



Amazon S3 > Buckets > employee-project1-data > Create folder

Create folder Info

Use folders to group objects in buckets. When you create a folder, S3 creates an object using the name that you specify followed by a slash (/). This object then appears as folder on the console. Learn more

Your bucket policy might block folder creation
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: inbox /

Folder names can't contain "/". See rules for naming

Server-side encryption

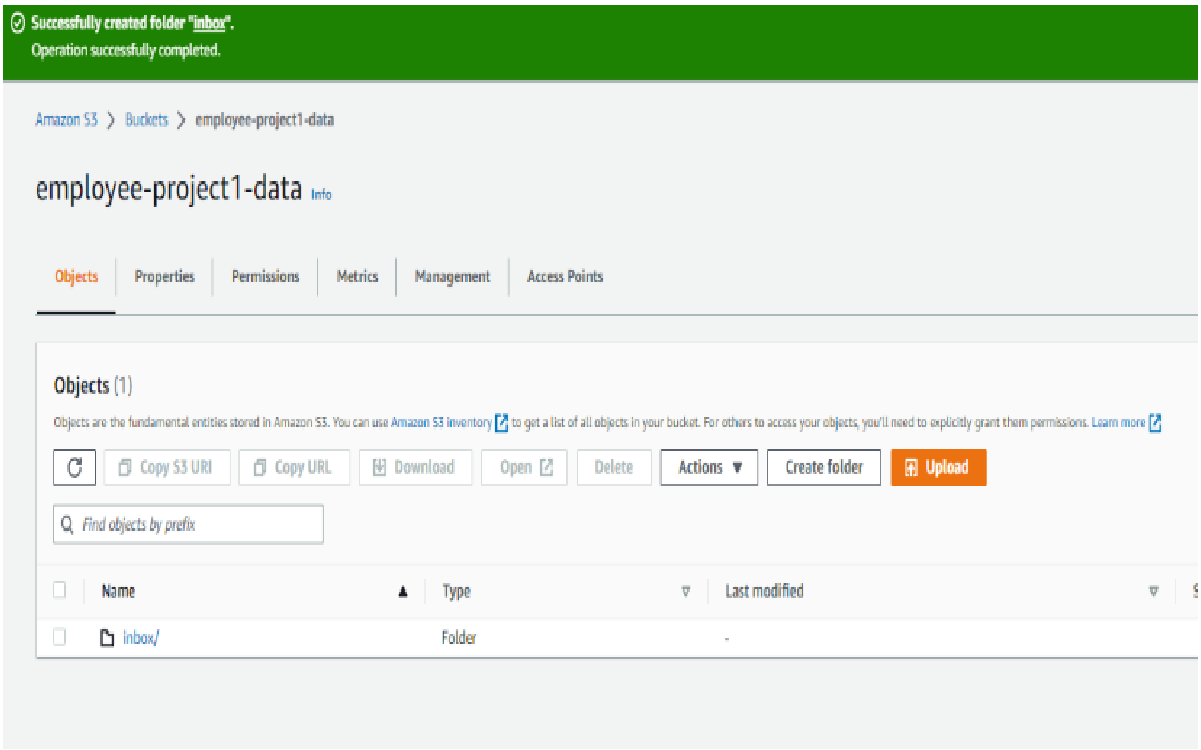
Server-side encryption protects data at rest.

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

Encryption key type Info

● Amazon S3-managed keys (SSE-S3)

○ AWS Key Management Service key (SSE-KMS)



Successfully created folder "inbox".
Operation successfully completed.

employee-project1-data Info

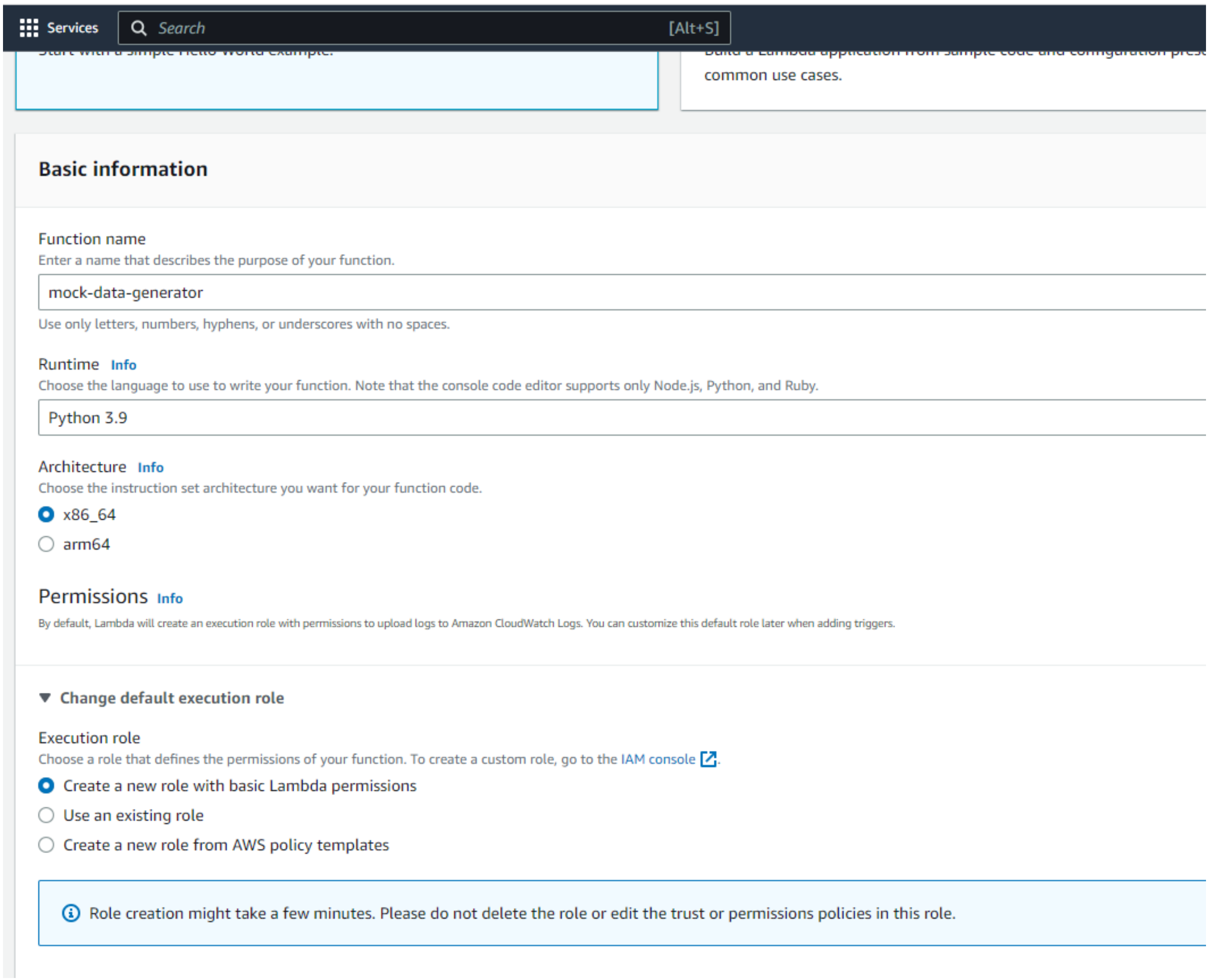
Objects (1)

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

□	Name	Type	Last modified
□	inbox/	Folder	-

Lambda:

Create lambda function 1 to generate mock data: mock-data-generator



common use cases.

Basic information

Function name

Enter a name that describes the purpose of your function.

mock-data-generator

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info

Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.

Python 3.9

Architecture Info

Choose the instruction set architecture you want for your function code.

● x86_64

○ arm64

Permissions Info

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ Change default execution role

Execution role

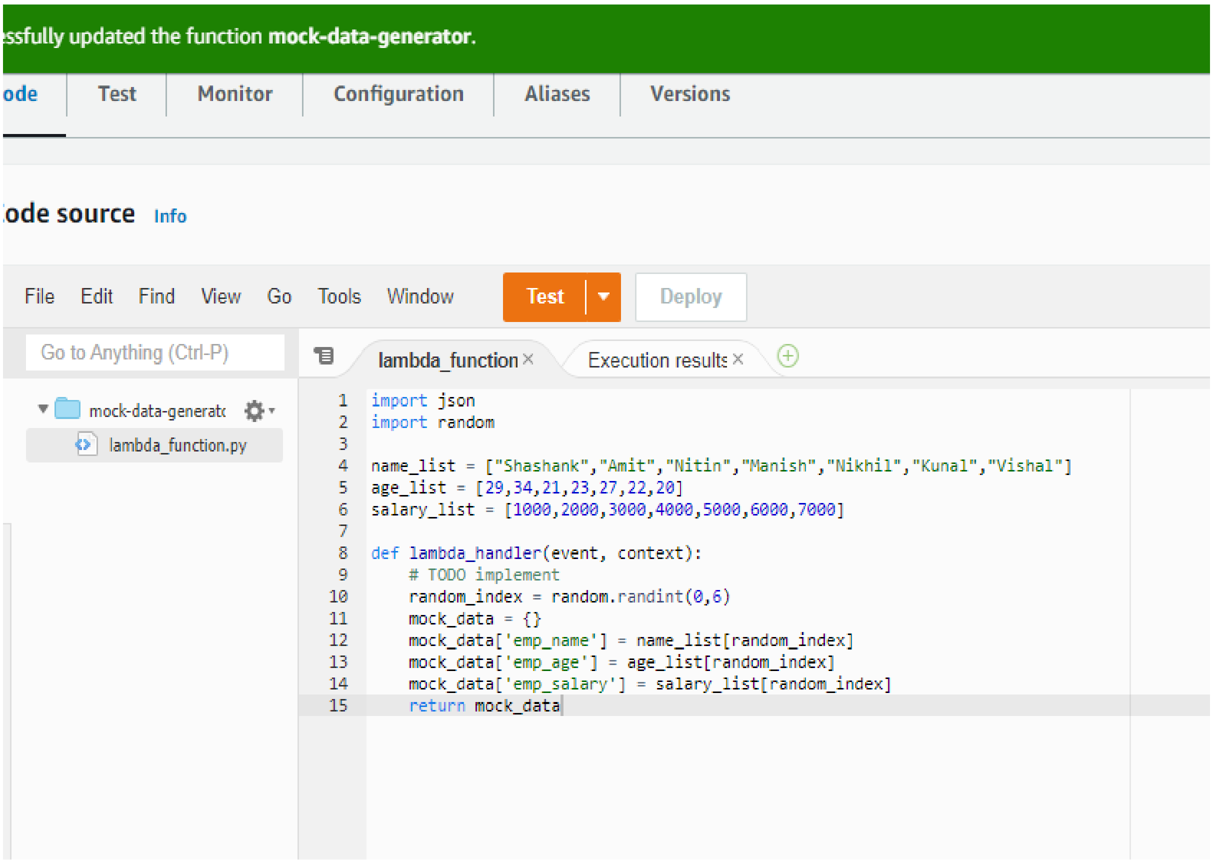
Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.

● Create a new role with basic Lambda permissions

○ Use an existing role

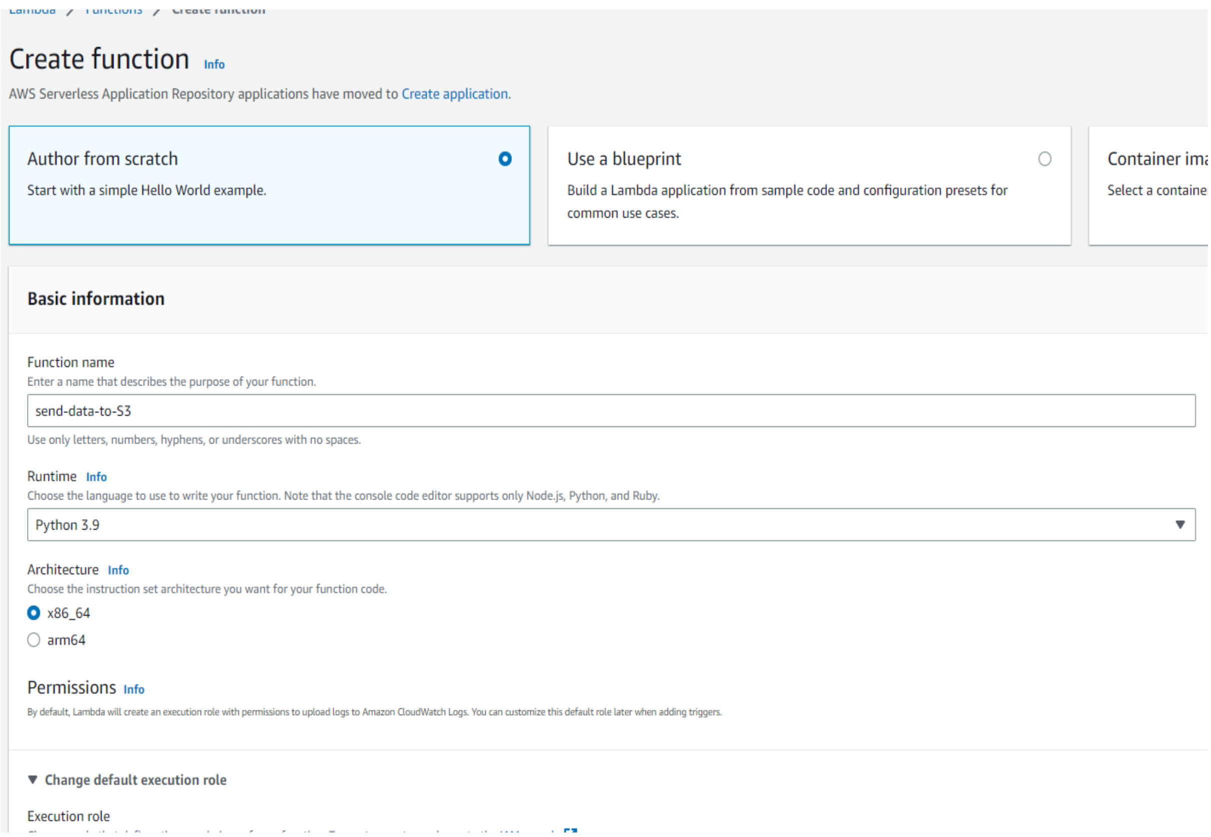
○ Create a new role from AWS policy templates

Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.



```python
import json
import random

name_list = ["Shashank","Amit","Nitin","Manish","Nikhil","Kunal","Vishal"]
age_list = [29,34,21,23,27,22,20]
salary_list = [1000,2000,3000,4000,5000,6000,7000]

def lambda_handler(event, context):
    # TODO implement
    random_index = random.randint(0,6)
    mock_data = {}
    mock_data['emp_name'] = name_list[random_index]
    mock_data['emp_age'] = age_list[random_index]
    mock_data['emp_salary'] = salary_list[random_index]
    return mock_data
```

Create lambda function 2 to write data inside S3 bucket's folder "inbox": send-data-to-S3



AWS Serverless Application Repository applications have moved to Create application.

● Author from scratch
Start with a simple Hello World example.

○ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

Container ima
Select a containe

Basic information

Function name
Enter a name that describes the purpose of your function.
send-data-to-S3
Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.9

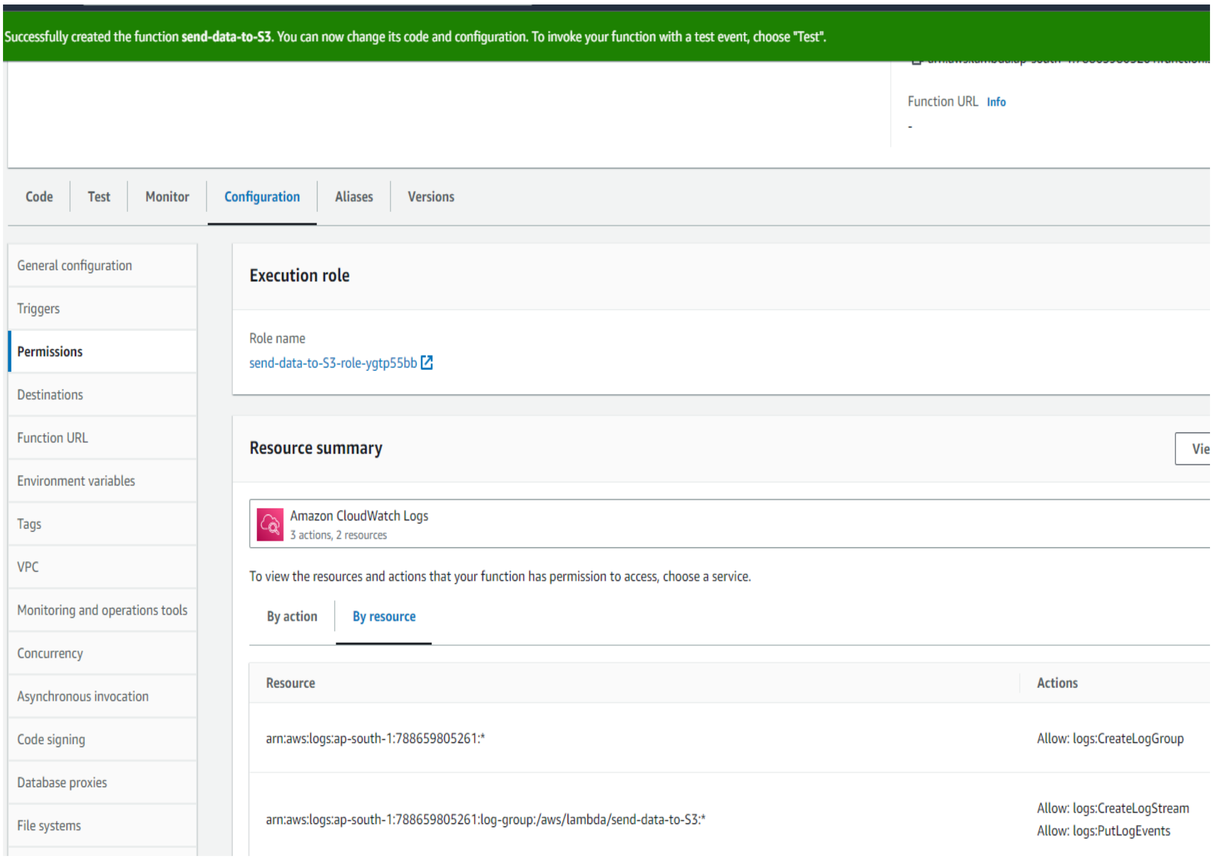
Architecture Info
Choose the instruction set architecture you want for your function code.
● x86_64
○ arm64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▼ Change default execution role

Execution role

Add IAM role to allow lambda function to access S3 bucket:



Successfully created the function **send-data-to-S3**. You can now change its code and configuration. To invoke your function with a test event, choose "Test".

Function URL Info
-

Execution role

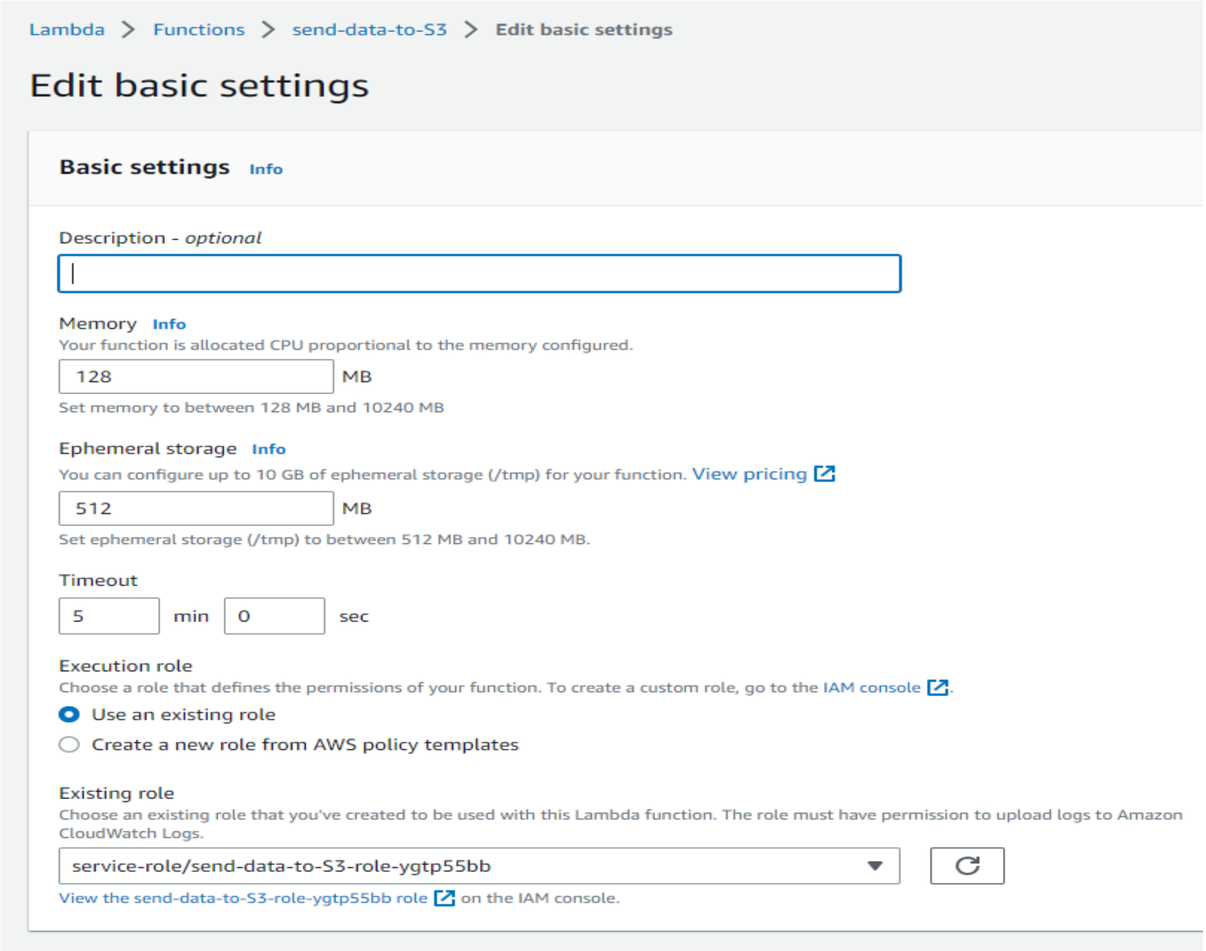
Role name
send-data-to-S3-role-ygtp55bb

Resource summary

Amazon CloudWatch Logs
3 actions, 2 resources

To view the resources and actions that your function has permission to access, choose a service.

Resource	Actions
arn:aws:logs:ap-south-1:788659805261:*	Allow: logs:CreateLogGroup
arn:aws:logs:ap-south-1:788659805261:log-group:/aws/lambda/send-data-to-S3:*	Allow: logs:CreateLogStream Allow: logs:PutLogEvents



Lambda > Functions > send-data-to-S3 > Edit basic settings

Edit basic settings

Basic settings Info

Description - *optional*

Memory Info

Your function is allocated CPU proportional to the memory configured.

128 MB

Set memory to between 128 MB and 10240 MB

Ephemeral storage Info

You can configure up to 10 GB of ephemeral storage (/tmp) for your function. View pricing

512 MB

Set ephemeral storage (/tmp) to between 512 MB and 10240 MB.

Timeout

5 min 0 sec

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the IAM console.

● Use an existing role

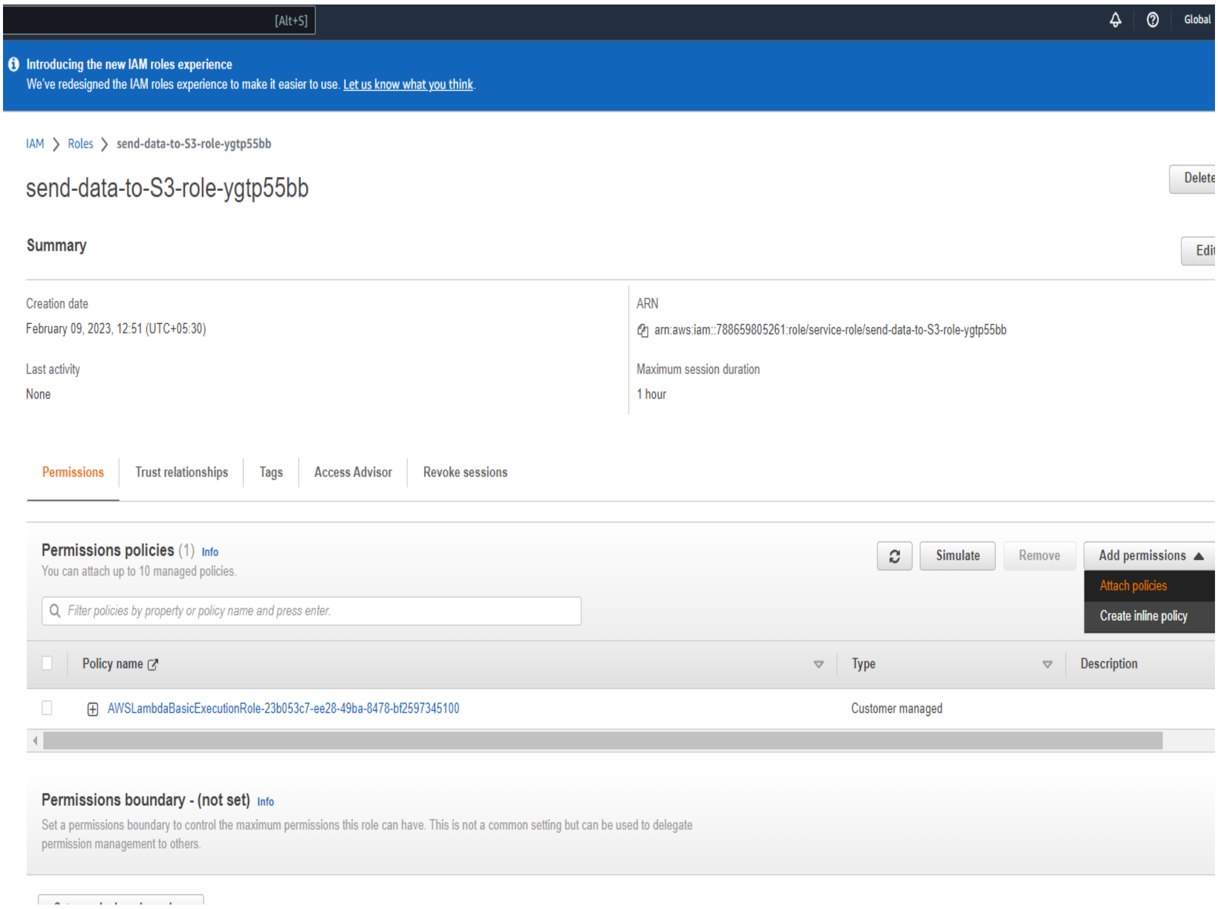
○ Create a new role from AWS policy templates

Existing role

Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

service-role/send-data-to-S3-role-ygtp55bb

View the send-data-to-S3-role-ygtp55bb role on the IAM console.



[Alt+S]

Global

Introducing the new IAM roles experience
We've redesigned the IAM roles experience to make it easier to use. Let us know what you think.

IAM > Roles > send-data-to-S3-role-ygtp55bb

send-data-to-S3-role-ygtp55bb

Delete

Summary

Edit

Creation date
February 09, 2023, 12:51 (UTC+05:30)

Last activity
None

ARN
arn:aws:iam::788659805261:role/service-role/send-data-to-S3-role-ygtp55bb

Maximum session duration
1 hour

Permissions | Trust relationships | Tags | Access Advisor | Revoke sessions

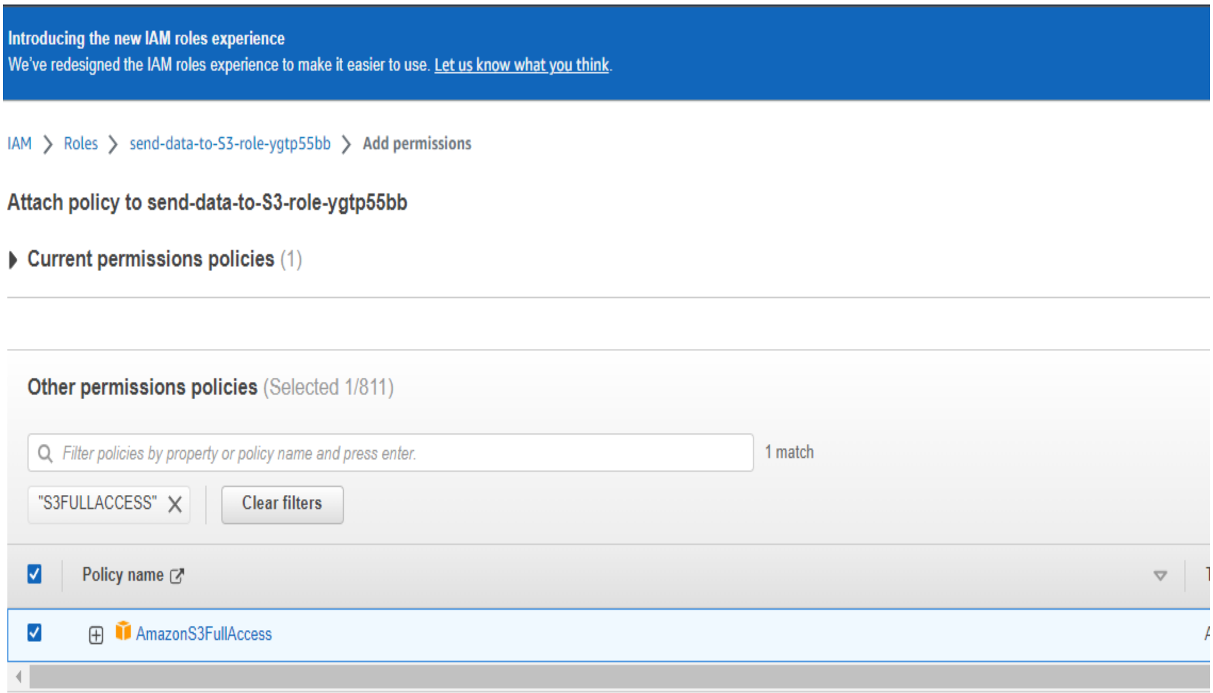
Permissions policies (1) Info
You can attach up to 10 managed policies.

Simulate | Remove | Add permissions ▲
- Attach policies
- Create inline policy

Filter policies by property or policy name and press enter.

□	Policy name	Type	Description
□	AWSLambdaBasicExecutionRole-23b053c7-ee28-49ba-8478-bf2597345100	Customer managed	

Permissions boundary - (not set) Info
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 others.



Introducing the new IAM roles experience
We've redesigned the IAM roles experience to make it easier to use. Let us know what you think.

IAM > Roles > send-data-to-S3-role-ygtp55bb > Add permissions

Attach policy to send-data-to-S3-role-ygtp55bb

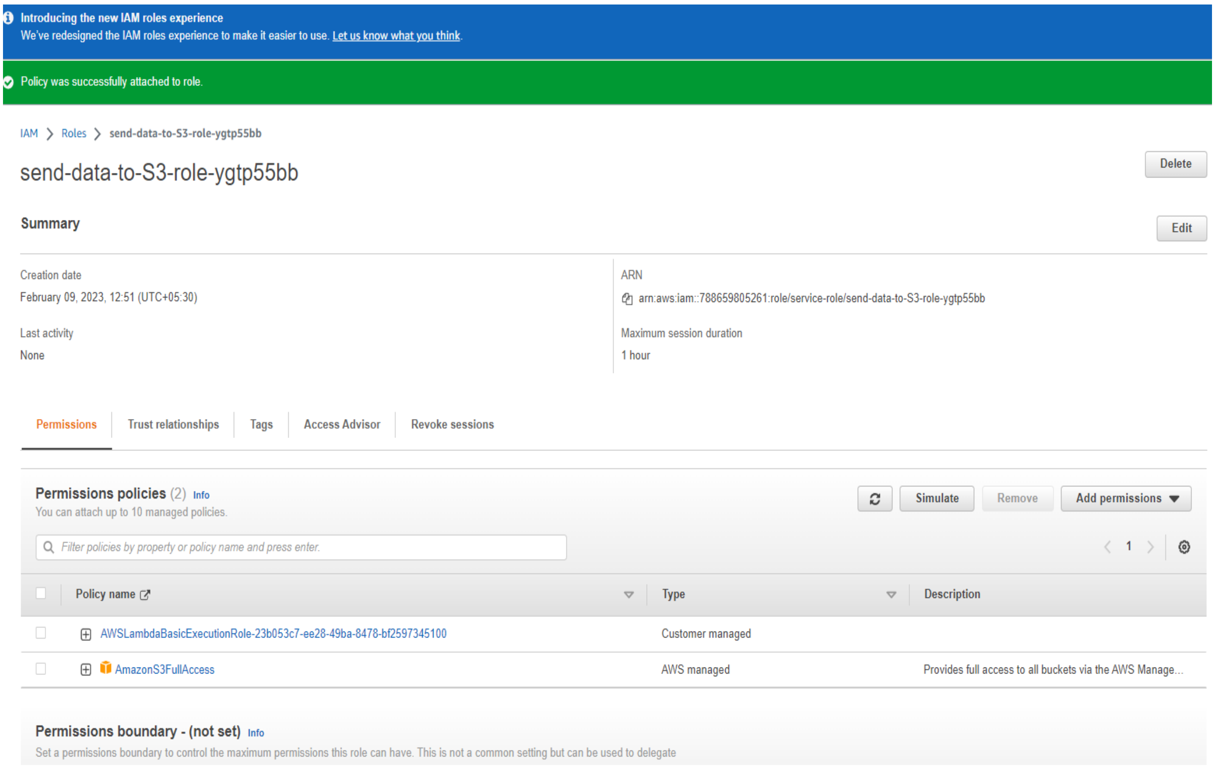
▸ Current permissions policies (1)

Other permissions policies (Selected 1/811)

Filter policies by property or policy name and press enter. 1 match

"S3FULLACCESS" ✕ | Clear filters

☑	Policy name
☑	AmazonS3FullAccess



Introducing the new IAM roles experience
We've redesigned the IAM roles experience to make it easier to use. Let us know what you think.

Policy was successfully attached to role.

IAM > Roles > send-data-to-S3-role-ygtp55bb

send-data-to-S3-role-ygtp55bb

Summary

Creation date
February 09, 2023, 12:51 (UTC+05:30)

ARN
arn:aws:iam::788659805261:role/service-role/send-data-to-S3-role-ygtp55bb

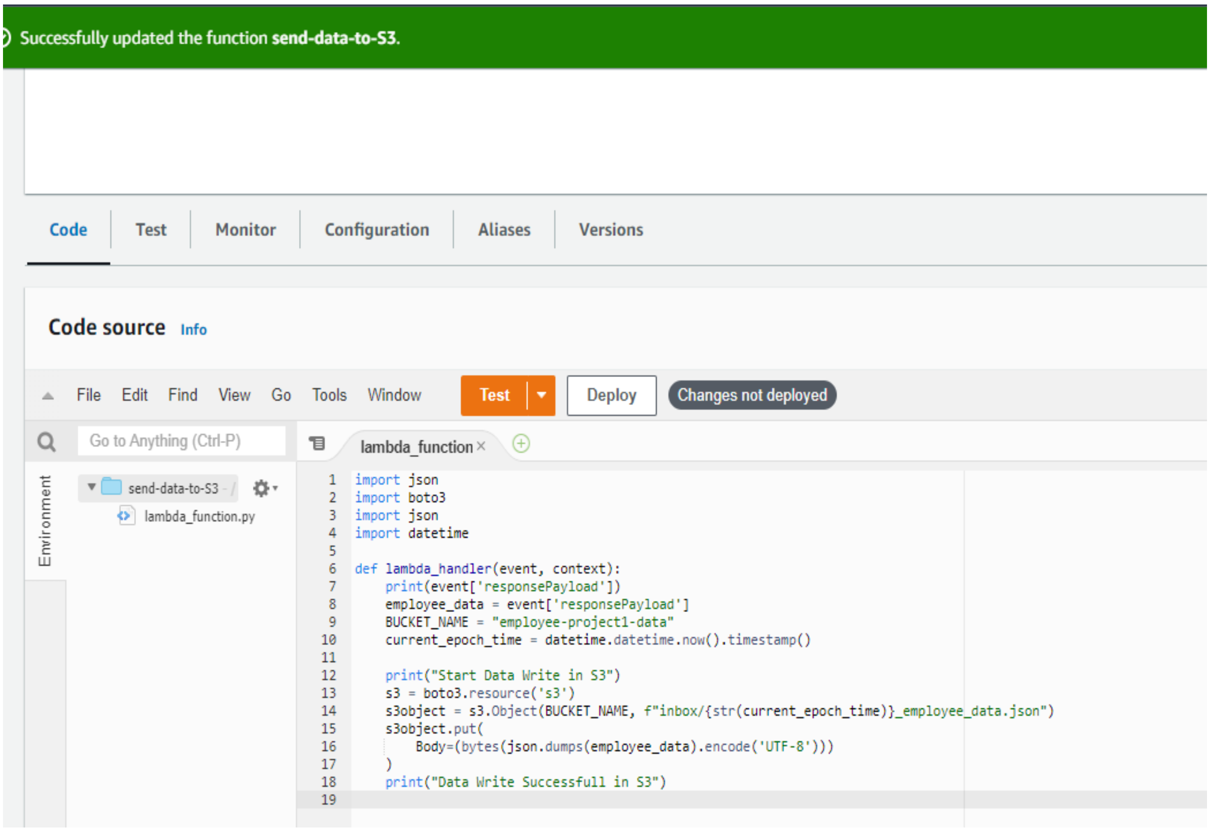
Last activity
None

Maximum session duration
1 hour

Permissions policies (2) Info
You can attach up to 10 managed policies.

	Policy name	Type	Description
□	⊞ AWSLambdaBasicExecutionRole-23b053c7-ee28-49ba-8478-bf2597345100	Customer managed	
□	⊞ AmazonS3FullAccess	AWS managed	Provides full access to all buckets via the AWS Manage...

Permissions boundary - (not set) Info
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



Successfully updated the function **send-data-to-S3**.

Code source Info

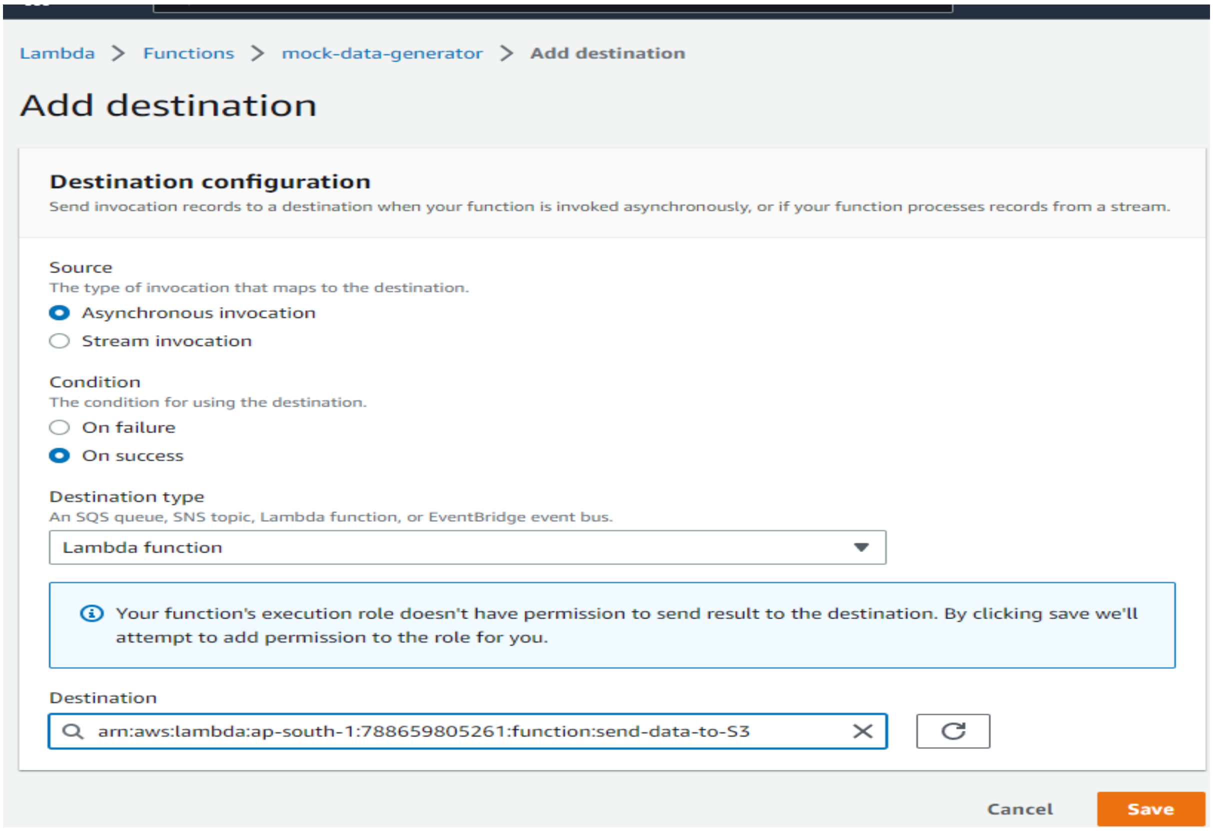
```python
import json
import boto3
import json
import datetime

def lambda_handler(event, context):
    print(event['responsePayload'])
    employee_data = event['responsePayload']
    BUCKET_NAME = "employee-project1-data"
    current_epoch_time = datetime.datetime.now().timestamp()

    print("Start Data Write in S3")
    s3 = boto3.resource('s3')
    s3object = s3.Object(BUCKET_NAME, f"inbox/{str(current_epoch_time)}_employee_data.json")
    s3object.put(
        Body=(bytes(json.dumps(employee_data).encode('UTF-8')))
    )
    print("Data Write Successfull in S3")
```

Epoch time is used here to give unique names to files generated by this lambda function.

An epoch timestamp is a way of representing a specific point in time using a number. This numerical value can be used for various date and time operations, making it easier to compare and manipulate different dates and times.

Add destination to lambda function 1 “mock-data-generator”

“send-data-to-S3” will be the destination for “mock-data-generator”



Lambda > Functions > mock-data-generator > Add destination

Add destination

Destination configuration

Send invocation records to a destination when your function is invoked asynchronously, or if your function processes records from a stream.

Source

The type of invocation that maps to the destination.

● Asynchronous invocation

○ Stream invocation

Condition

The condition for using the destination.

○ On failure

● On success

Destination type

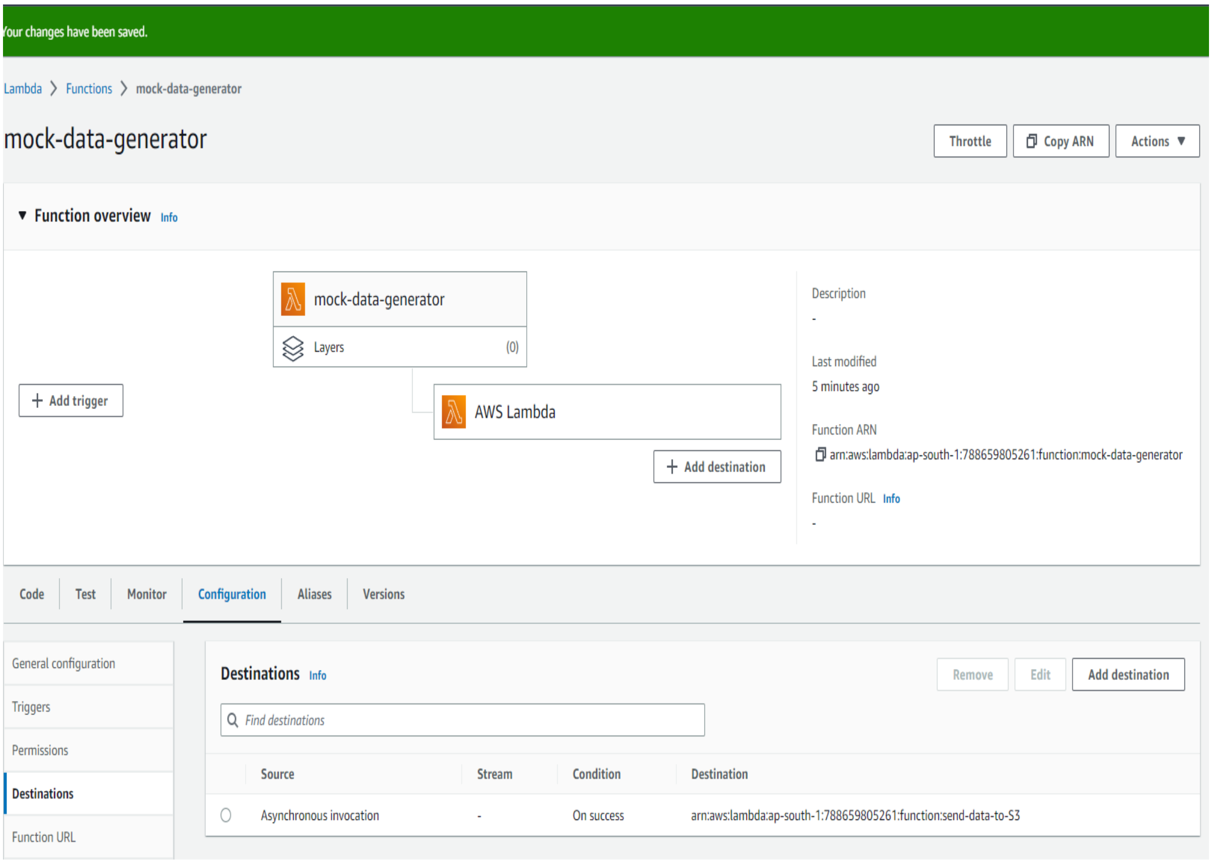
An SQS queue, SNS topic, Lambda function, or EventBridge event bus.

Lambda function

Your function's execution role doesn't have permission to send result to the destination. By clicking save we'll attempt to add permission to the role for you.

Destination

arn:aws:lambda:ap-south-1:788659805261:function:send-data-to-S3



Description
-

Last modified
5 minutes ago

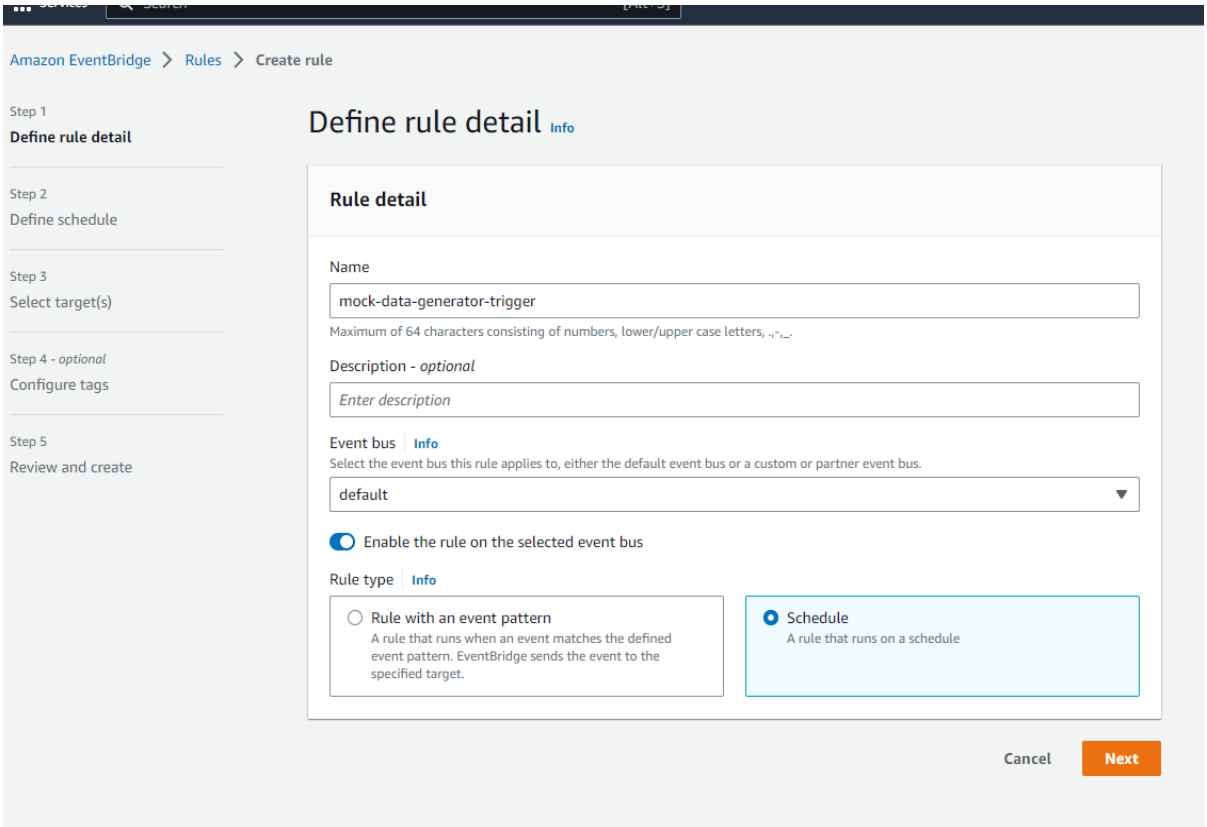
Function ARN
arn:aws:lambda:ap-south-1:788659805261:function:mock-data-generator

Function URL Info
-

	Source	Stream	Condition	Destination
○	Asynchronous invocation	-	On success	arn:aws:lambda:ap-south-1:788659805261:function:send-data-to-S3

Eventbridge:

Create eventbridge rule: schedule mock-data-generator function to trigger.



Amazon EventBridge > Rules > Create rule

Step 1
Define rule detail

Step 2
Define schedule

Step 3
Select target(s)

Step 4 - *optional*
Configure tags

Step 5
Review and create

Define rule detail Info

Rule detail

Name

mock-data-generator-trigger

Maximum of 64 characters consisting of numbers, lower/upper case letters, .,-,_.

Description - *optional*

Enter description

Event bus Info

Select the event bus this rule applies to, either the default event bus or a custom or partner event bus.

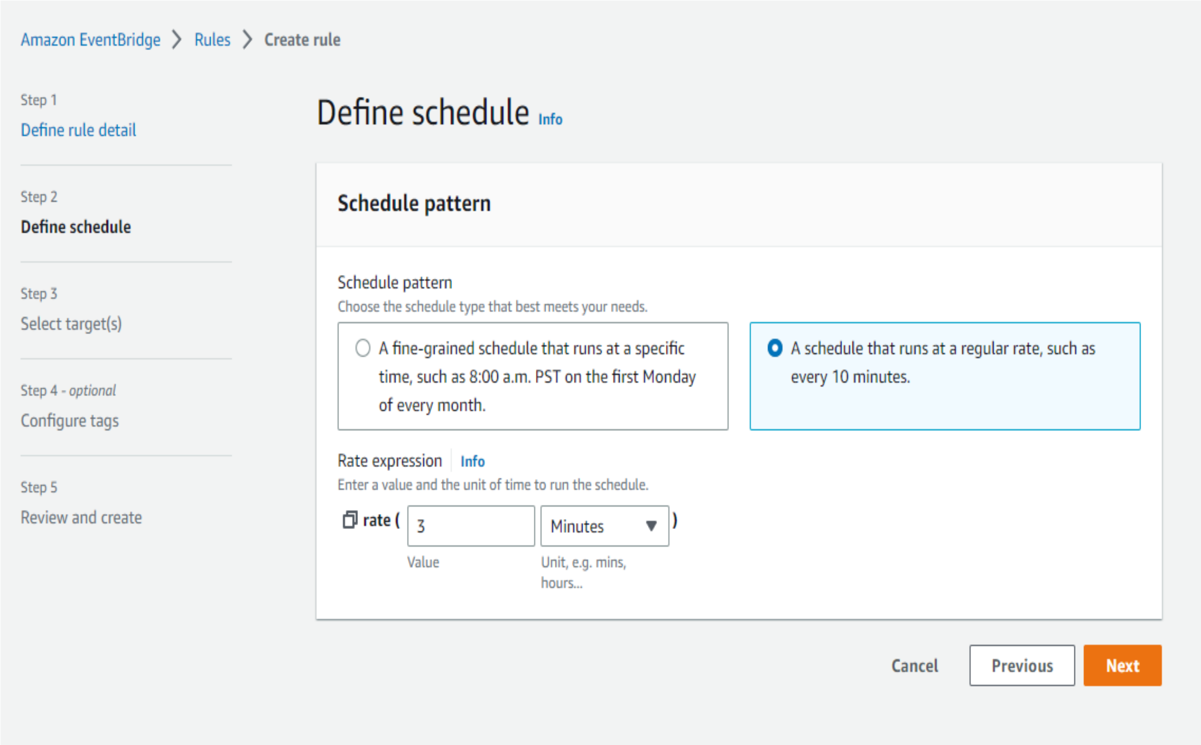
default ▼

Enable the rule on the selected event bus

Rule type Info

○ Rule with an event pattern
A rule that runs when an event matches the defined event pattern. EventBridge sends the event to the specified target.

● Schedule
A rule that runs on a schedule



Amazon EventBridge > Rules > Create rule

Step 1
Define rule detail

Step 2
Define schedule

Step 3
Select target(s)

Step 4 - *optional*
Configure tags

Step 5
Review and create

Define schedule Info

Schedule pattern

Schedule pattern

Choose the schedule type that best meets your needs.

○ A fine-grained schedule that runs at a specific time, such as 8:00 a.m. PST on the first Monday of every month.

● A schedule that runs at a regular rate, such as every 10 minutes.

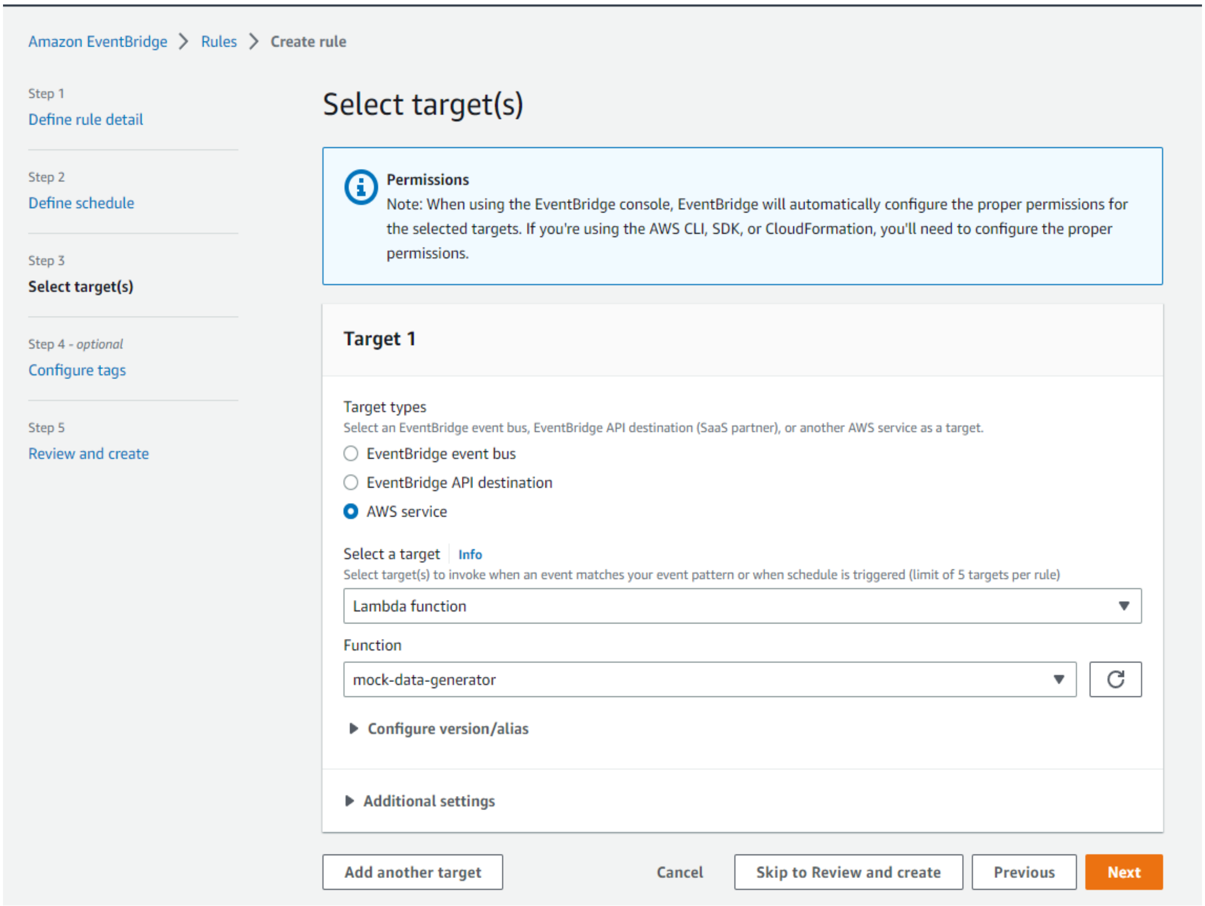
Rate expression Info

Enter a value and the unit of time to run the schedule.

rate (3 Minutes ▼)

Value

Unit, e.g. mins, hours...



Amazon EventBridge > Rules > Create rule

Step 1
Define rule detail

Step 2
Define schedule

Step 3
Select target(s)

Step 4 - *optional*
Configure tags

Step 5
Review and create

Select target(s)

Permissions
Note: When using the EventBridge console, EventBridge will automatically configure the proper permissions for the selected targets. If you're using the AWS CLI, SDK, or CloudFormation, you'll need to configure the proper permissions.

Target 1

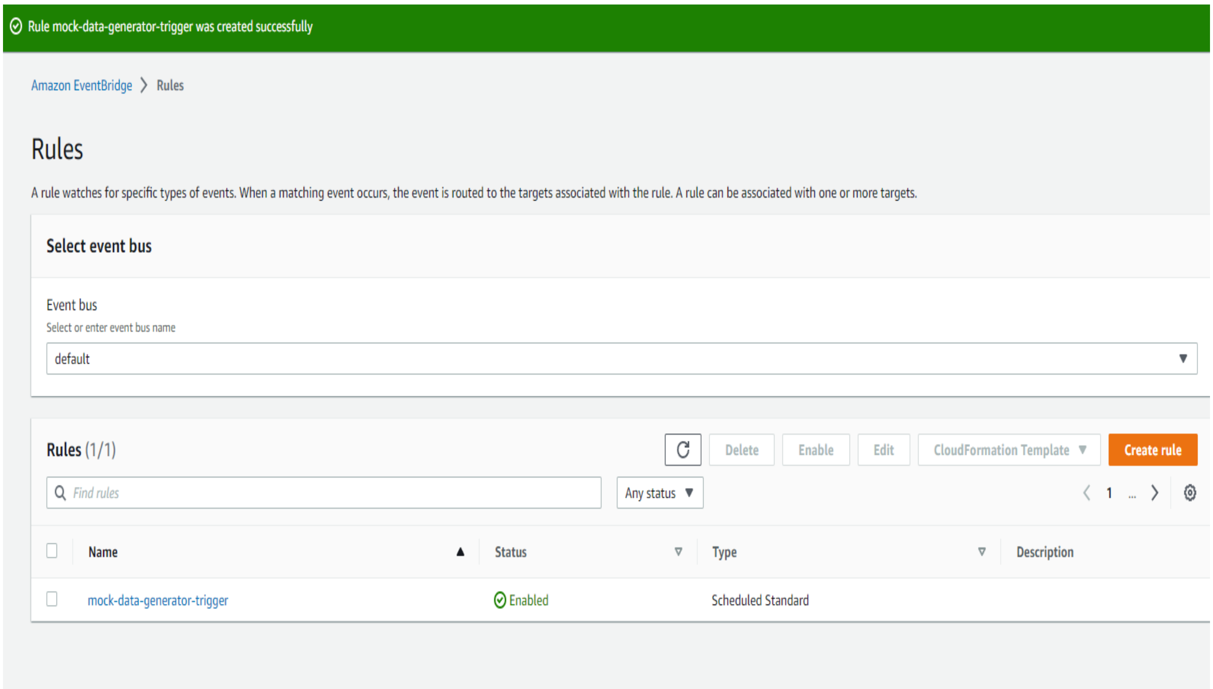
Target types
Select an EventBridge event bus, EventBridge API destination (SaaS partner), or another AWS service as a target.
○ EventBridge event bus
○ EventBridge API destination
● AWS service

Select a target Info
Select target(s) to invoke when an event matches your event pattern or when schedule is triggered (limit of 5 targets per rule)
Lambda function ▼

Function
mock-data-generator ▼

▶ **Configure version/alias**

▶ **Additional settings**



Rule mock-data-generator-trigger was created successfully

Amazon EventBridge > Rules

Rules

A rule watches for specific types of events. When a matching event occurs, the event is routed to the targets associated with the rule. A rule can be associated with one or more targets.

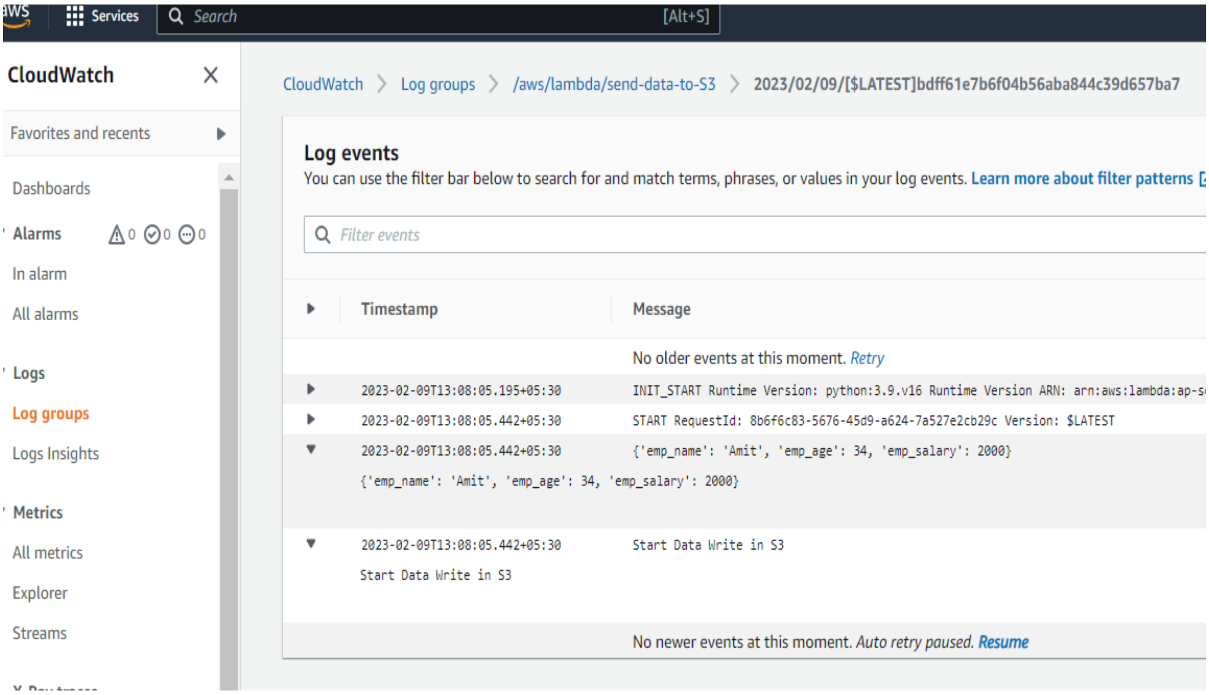
Select event bus

Event bus
Select or enter event bus name
default ▼

Rules (1/1)

□	Name ▲	Status ▽	Type ▽	Description
□	mock-data-generator-trigger	✓ Enabled	Scheduled Standard	

Enable the rule and check logs:



CloudWatch > Log groups > /aws/lambda/send-data-to-S3 > 2023/02/09/[$LATEST]bdff61e7b6f04b56aba844c39d657ba7

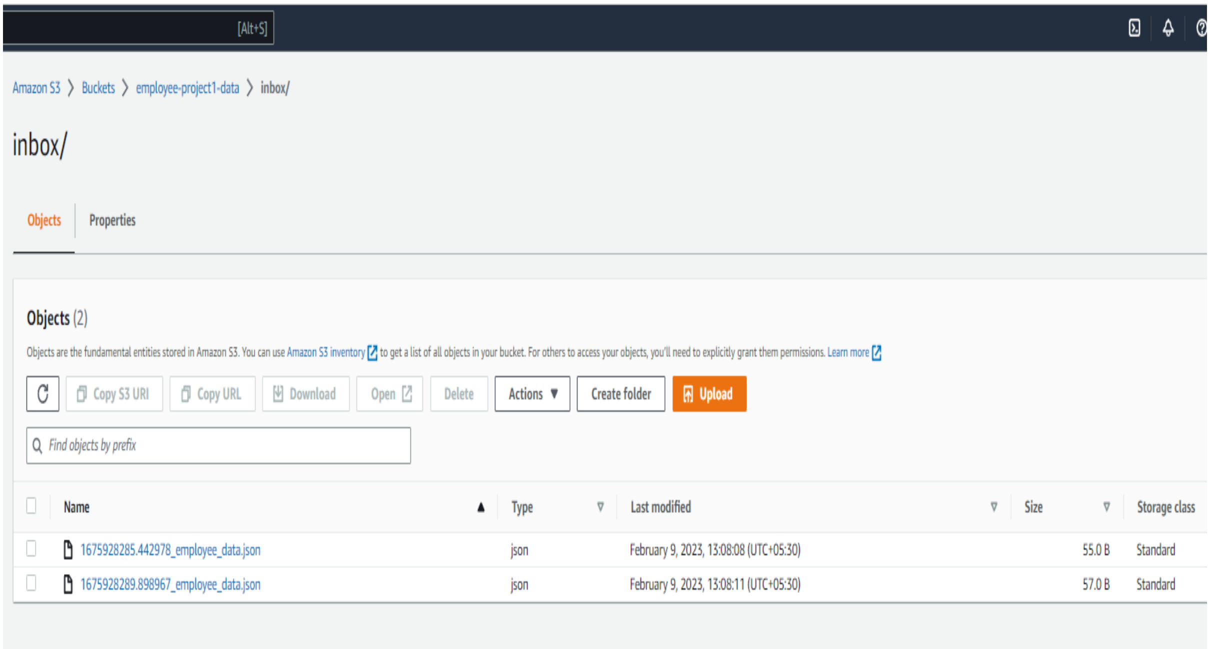
Log events

You can use the filter bar below to search for and match terms, phrases, or values in your log events. Learn more about filter patterns

Filter events

Timestamp	Message
	No older events at this moment. *Retry*
2023-02-09T13:08:05.195+05:30	INIT_START Runtime Version: python:3.9.v16 Runtime Version ARN: arn:aws:lambda:ap-s
2023-02-09T13:08:05.442+05:30	START RequestId: 8b6f6c83-5676-45d9-a624-7a527e2cb29c Version: $LATEST
2023-02-09T13:08:05.442+05:30	{'emp_name': 'Amit', 'emp_age': 34, 'emp_salary': 2000}
{'emp_name': 'Amit', 'emp_age': 34, 'emp_salary': 2000}	
2023-02-09T13:08:05.442+05:30	Start Data Write in S3
Start Data Write in S3	
	No newer events at this moment. *Auto retry paused.* Resume

Check data inside S3:



Amazon S3 > Buckets > employee-project1-data > inbox/

inbox/

Objects (2)

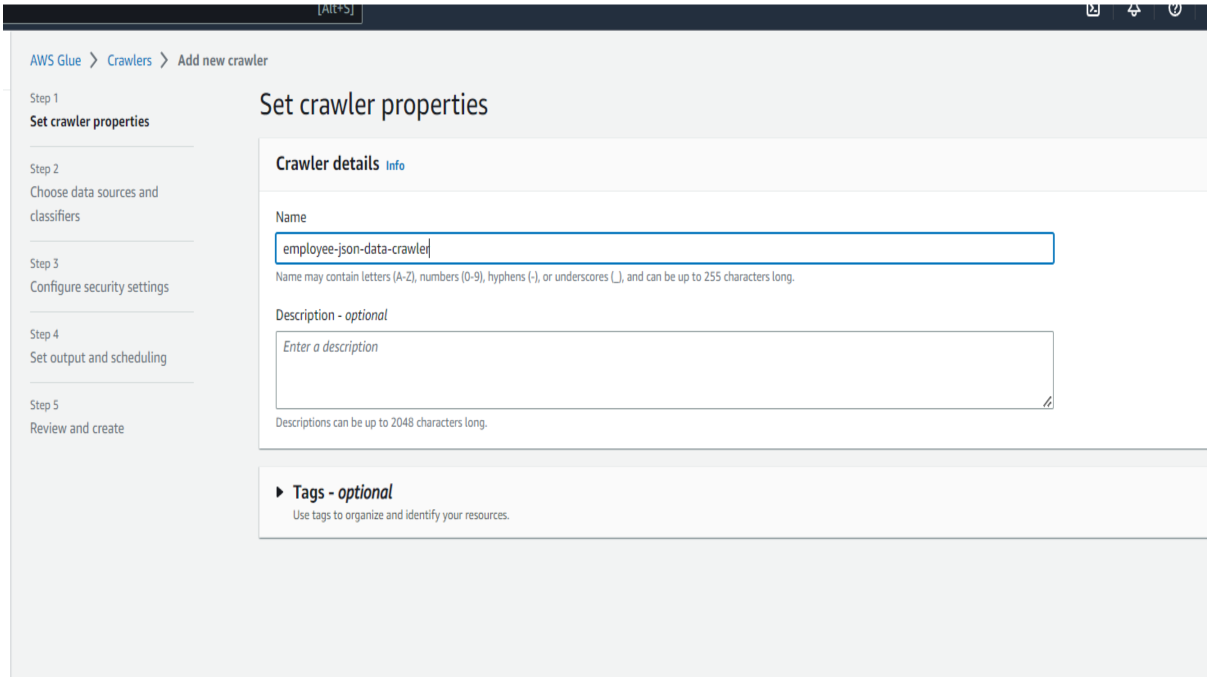
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class
1675928285.442978_employee_data.json	json	February 9, 2023, 13:08:08 (UTC+05:30)	55.0 B	Standard
1675928289.898967_employee_data.json	json	February 9, 2023, 13:08:11 (UTC+05:30)	57.0 B	Standard

-------Mock data is generated and written successfully inside S3-------

Glue:

Create crawler: employee-json-data-crawler



AWS Glue > Crawlers > Add new crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Set crawler properties

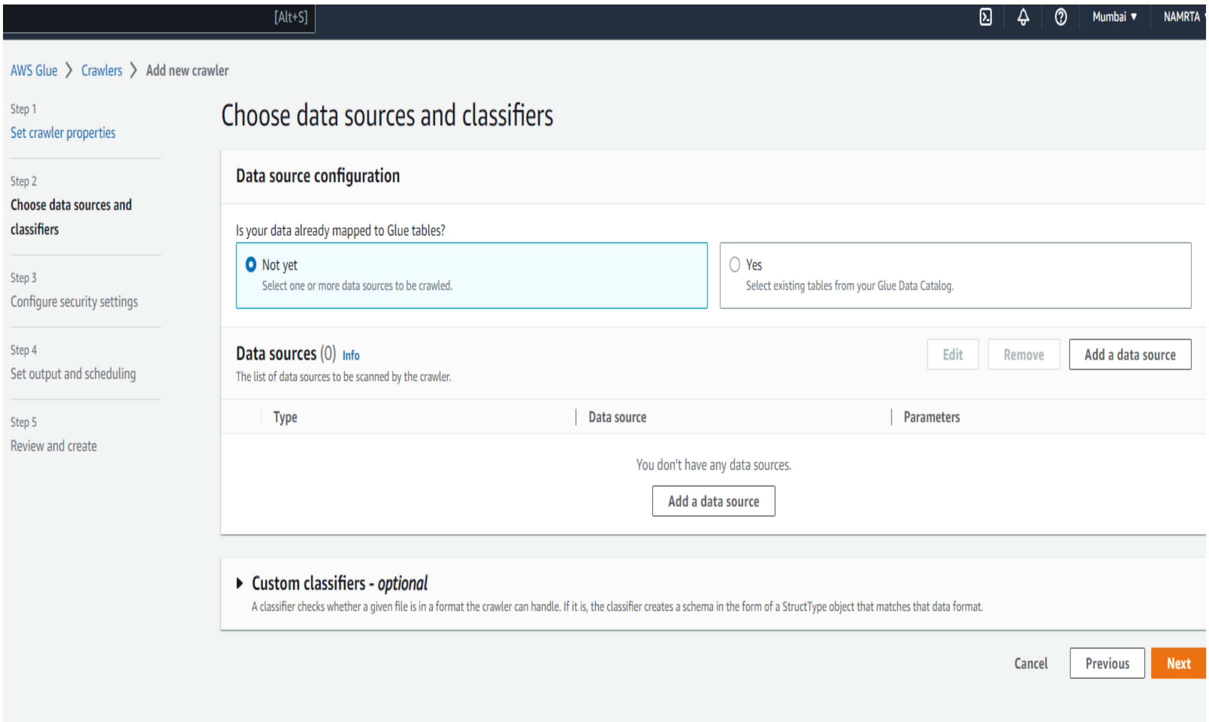
Crawler details Info

Name
employee-json-data-crawler
Name may contain letters (A-Z), numbers (0-9), hyphens (-), or underscores (_), and can be up to 255 characters long.

Description - *optional*
Enter a description
Descriptions can be up to 2048 characters long.

▸ Tags - *optional*
Use tags to organize and identify your resources.

Add data source to crawl:



AWS Glue > Crawlers > Add new crawler

Step 1
Set crawler properties

Step 2
Choose data sources and classifiers

Step 3
Configure security settings

Step 4
Set output and scheduling

Step 5
Review and create

Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

● Not yet
Select one or more data sources to be crawled.

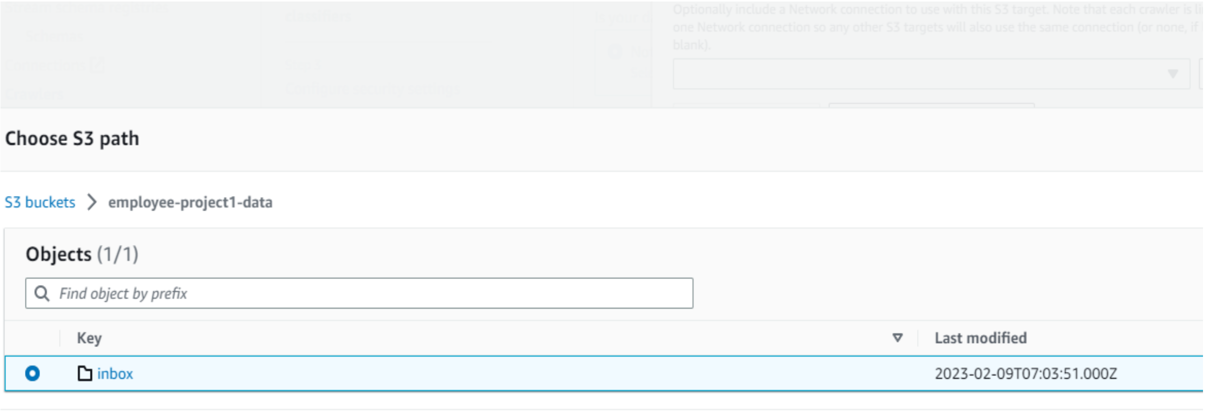
○ Yes
Select existing tables from your Glue Data Catalog.

Data sources (0) Info
The list of data sources to be scanned by the crawler.

Type	Data source	Parameters

You don't have any data sources.

▸ Custom classifiers - *optional*
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.



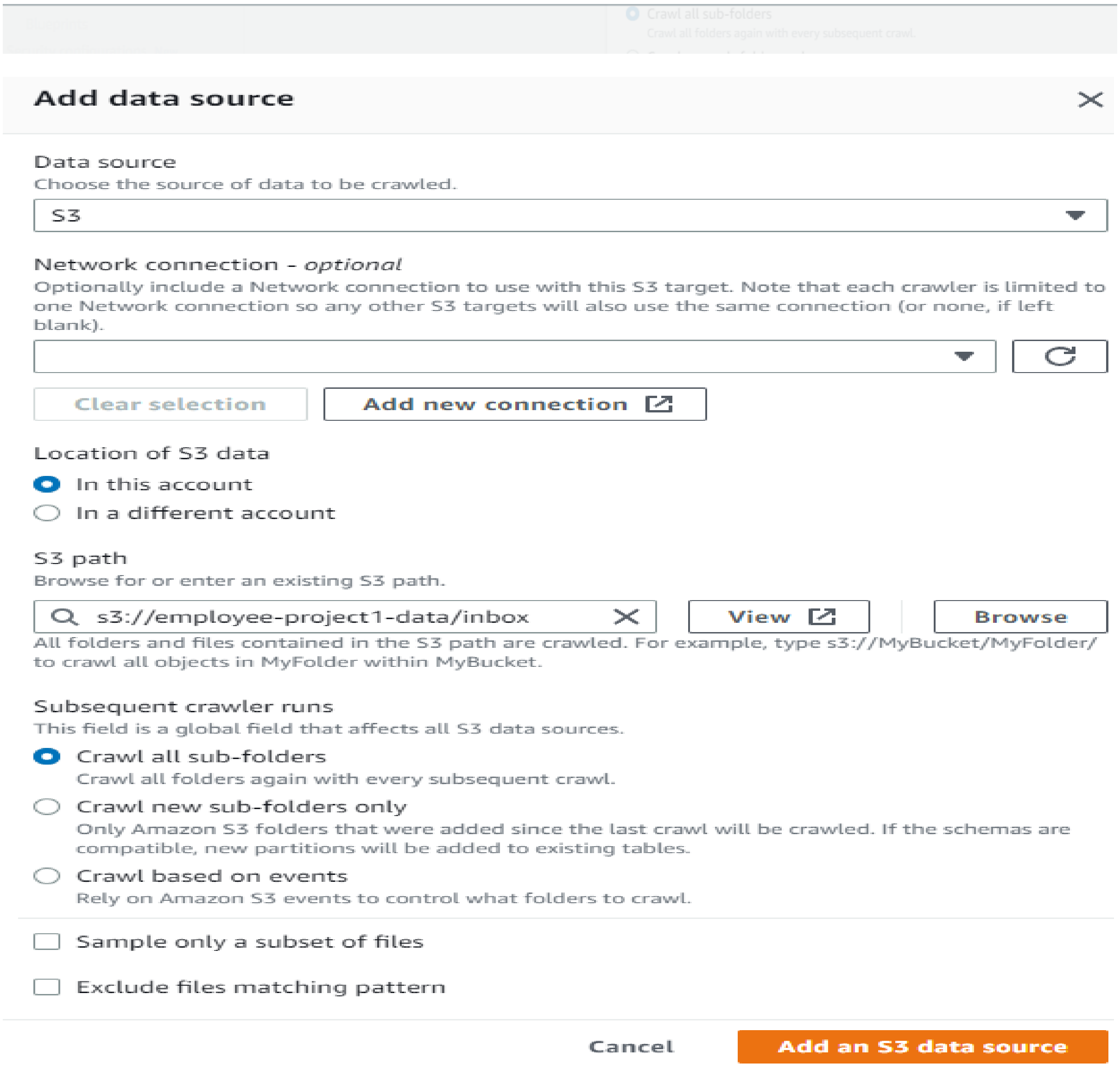
Choose S3 path

S3 buckets > employee-project1-data

Objects (1/1)

Find object by prefix

	Key	Last modified
●	inbox	2023-02-09T07:03:51.000Z



Add data source

Data source
Choose the source of data to be crawled.

S3

Network connection - *optional*
Optionally include a Network connection to use with this S3 target. Note that each crawler is limited to one Network connection so any other S3 targets will also use the same connection (or none, if left blank).

Location of S3 data

● In this account
○ In a different account

S3 path
Browse for or enter an existing S3 path.

s3://employee-project1-data/inbox

All folders and files contained in the S3 path are crawled. For example, type s3://MyBucket/MyFolder/ to crawl all objects in MyFolder within MyBucket.

Subsequent crawler runs
This field is a global field that affects all S3 data sources.

● Crawl all sub-folders
Crawl all folders again with every subsequent crawl.

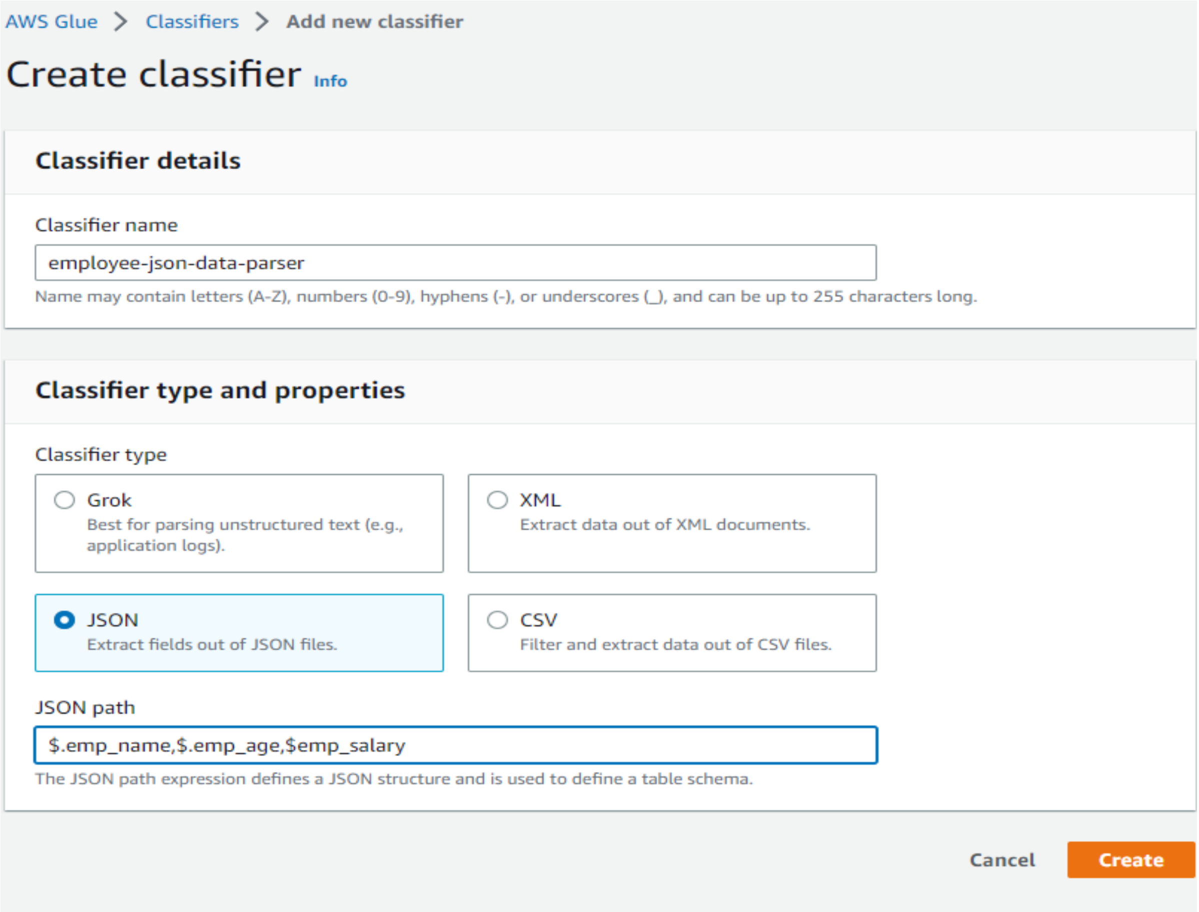
○ Crawl new sub-folders only
Only Amazon S3 folders that were added since the last crawl will be crawled. If the schemas are compatible, new partitions will be added to existing tables.

○ Crawl based on events
Rely on Amazon S3 events to control what folders to crawl.

□ Sample only a subset of files

□ Exclude files matching pattern

Add classifier: Crawler will crawl the files present in selected data source using this classifier.



AWS Glue > Classifiers > Add new classifier

Create classifier Info

Classifier details

Classifier name

employee-json-data-parser

Name may contain letters (A-Z), numbers (0-9), hyphens (-), or underscores (_), and can be up to 255 characters long.

Classifier type and properties

Classifier type

○ Grok
Best for parsing unstructured text (e.g., application logs).

○ XML
Extract data out of XML documents.

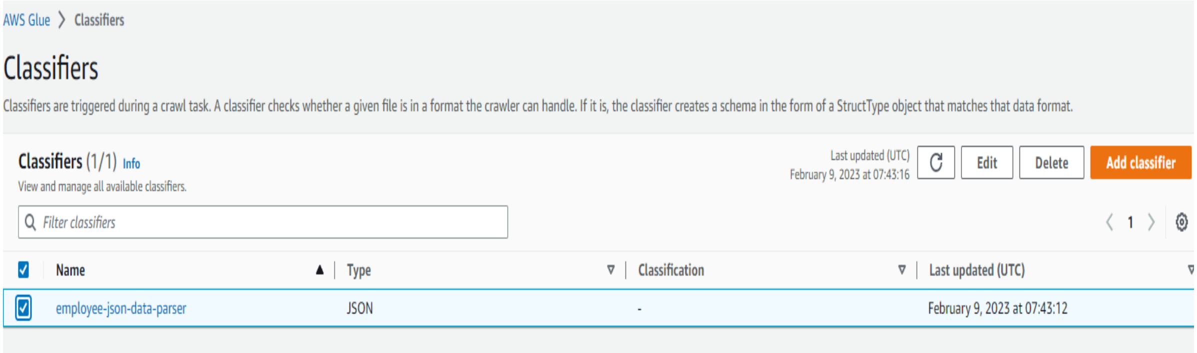
● JSON
Extract fields out of JSON files.

○ CSV
Filter and extract data out of CSV files.

JSON path

$.emp_name,$.emp_age,$emp_salary

The JSON path expression defines a JSON structure and is used to define a table schema.



AWS Glue > Classifiers

Classifiers

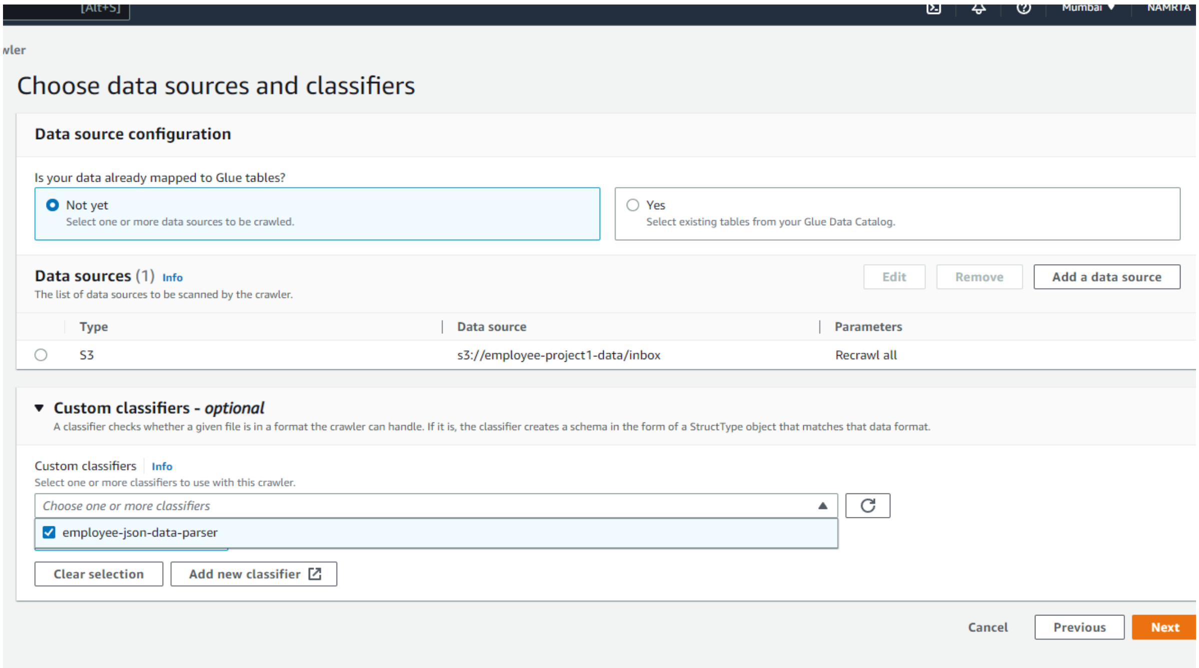
Classifiers are triggered during a crawl task. A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.

Classifiers (1/1) Info

View and manage all available classifiers.

Last updated (UTC)
February 9, 2023 at 07:43:16

☑	Name	Type	Classification	Last updated (UTC)
☑	employee-json-data-parser	JSON	-	February 9, 2023 at 07:43:12



Choose data sources and classifiers

Data source configuration

Is your data already mapped to Glue tables?

● Not yet
Select one or more data sources to be crawled.

○ Yes
Select existing tables from your Glue Data Catalog.

Data sources (1) Info
The list of data sources to be scanned by the crawler.

	Type	Data source	Parameters
○	S3	s3://employee-project1-data/inbox	Recrawl all

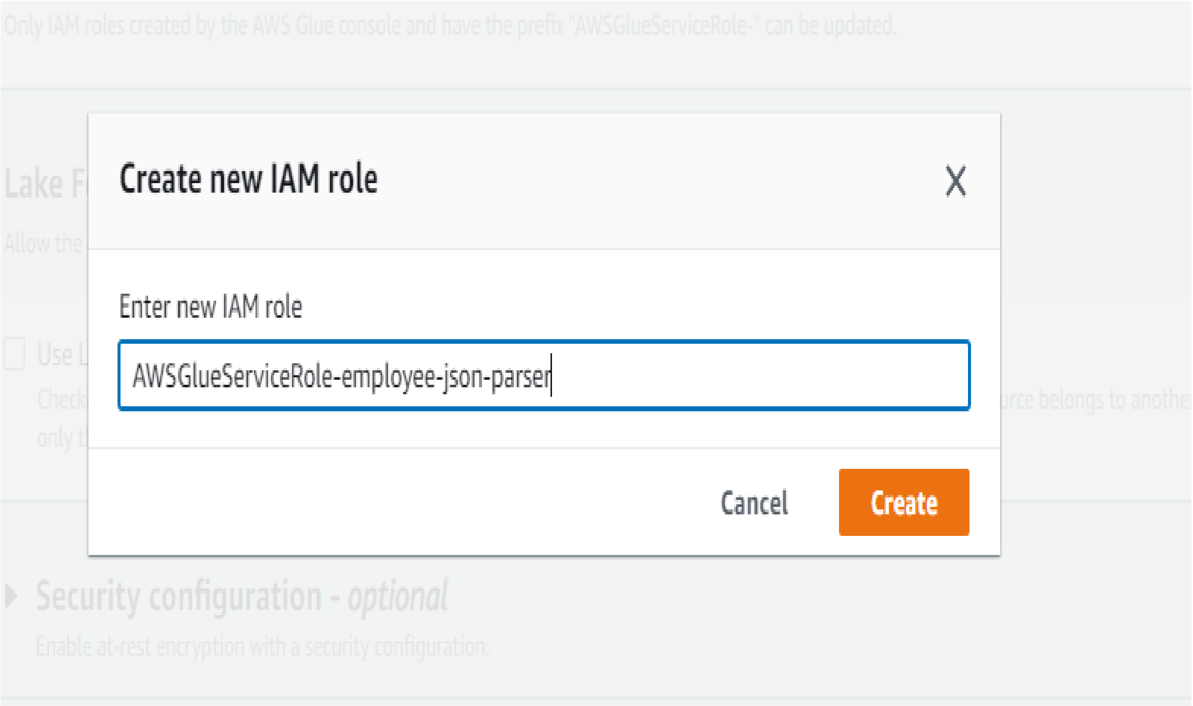
▾ Custom classifiers - *optional*
A classifier checks whether a given file is in a format the crawler can handle. If it is, the classifier creates a schema in the form of a StructType object that matches that data format.

Custom classifiers Info
Select one or more classifiers to use with this crawler.

Choose one or more classifiers

☑ employee-json-data-parser

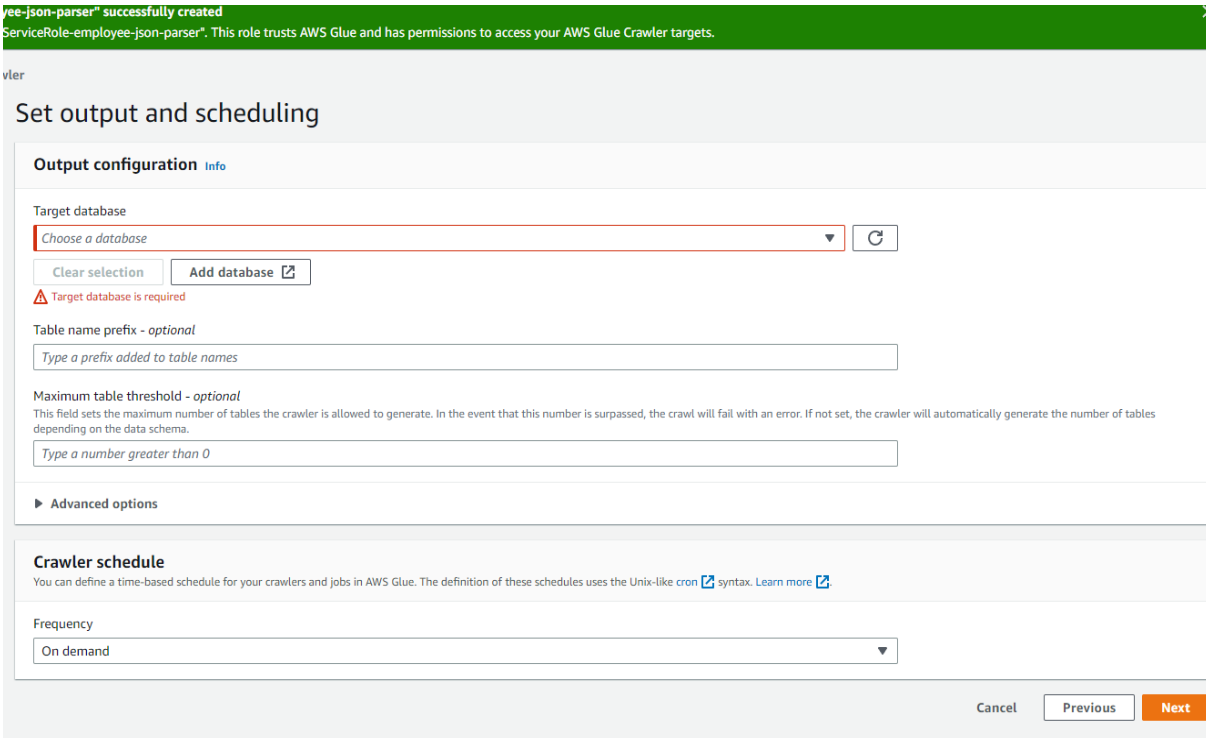
Clear selection
Add new classifier
Cancel
Previous
Next


Only IAM roles created by the AWS Glue console and have the prefix "AWSGlueServiceRole-" can be updated.
Lake F
Allow the
Use L
Check
only t
urce belongs to anothe
Security configuration - optional
Enable at-rest encryption with a security configuration.

Create new IAM role

Enter new IAM role

AWSGlueServiceRole-employee-json-parser



yee-json-parser" successfully created

ServiceRole-employee-json-parser". This role trusts AWS Glue and has permissions to access your AWS Glue Crawler targets.

vler

Set output and scheduling

Output configuration Info

Target database

Choose a database

⚠ Target database is required

Table name prefix - *optional*

Type a prefix added to table names

Maximum table threshold - *optional*

This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

Type a number greater than 0

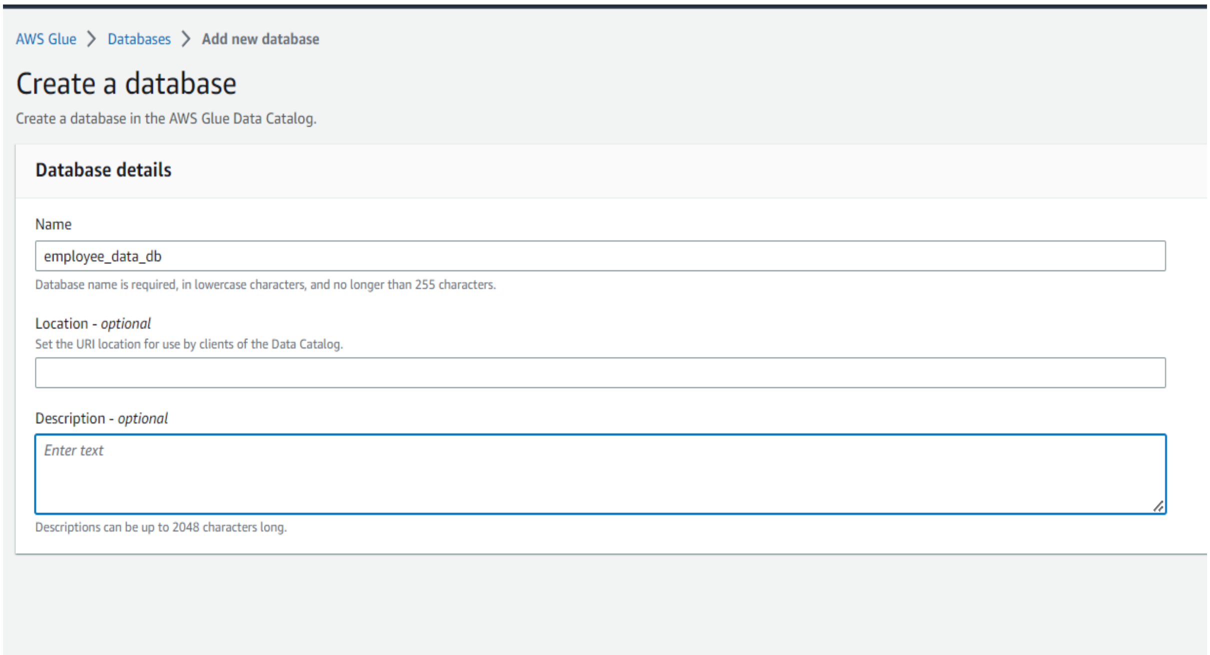
Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.

Frequency

On demand

Create database: crawler will create table inside this database to store metadata.



Create a database

Create a database in the AWS Glue Data Catalog.

Database details

Name

employee_data_db

Database name is required, in lowercase characters, and no longer than 255 characters.

Location - *optional*

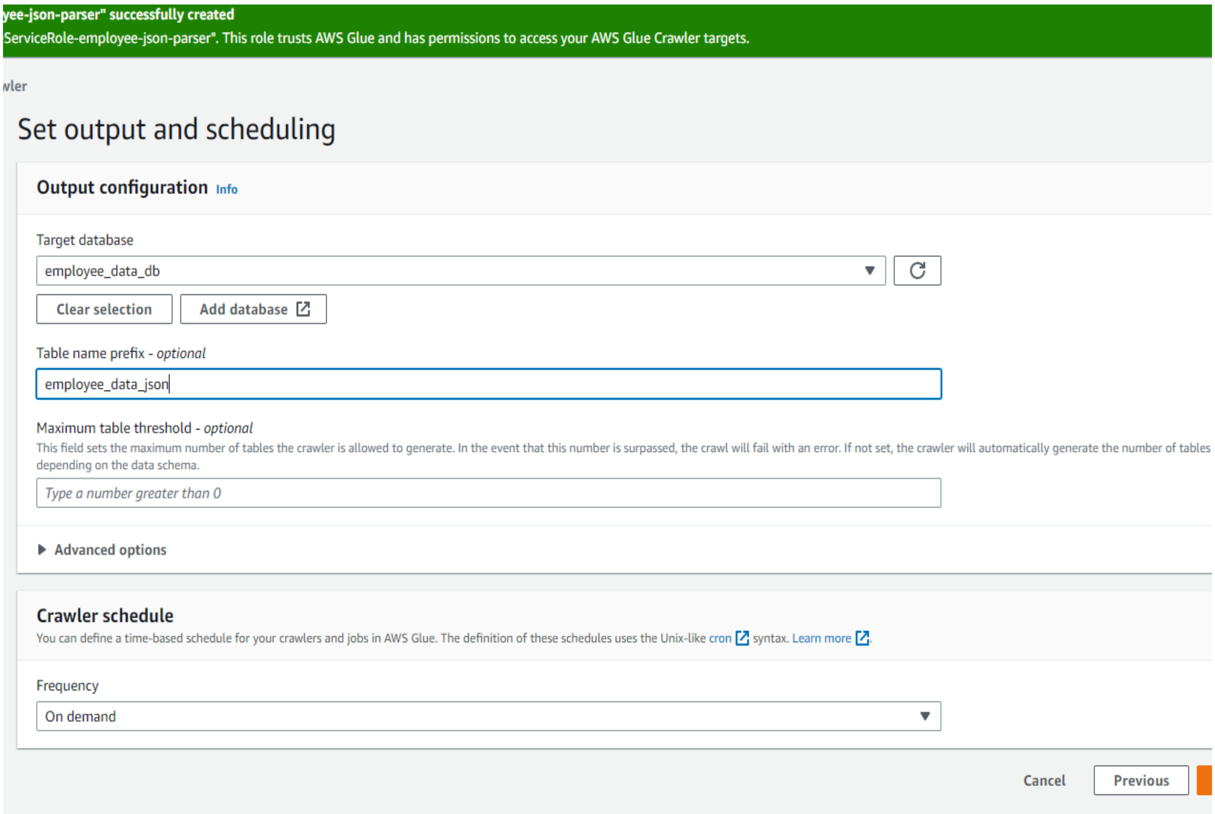
Set the URI location for use by clients of the Data Catalog.

Description - *optional*

Enter text

Descriptions can be up to 2048 characters long.

Give prefix to the table name that crawler will create for storing metadata -> this prefix will help in identifying the required table.



yee-json-parser" successfully created

ServiceRole-employee-json-parser". This role trusts AWS Glue and has permissions to access your AWS Glue Crawler targets.

wler

Set output and scheduling

Output configuration Info

Target database

employee_data_db

Table name prefix - *optional*

employee_data_json

Maximum table threshold - *optional*

This field sets the maximum number of tables the crawler is allowed to generate. In the event that this number is surpassed, the crawl will fail with an error. If not set, the crawler will automatically generate the number of tables depending on the data schema.

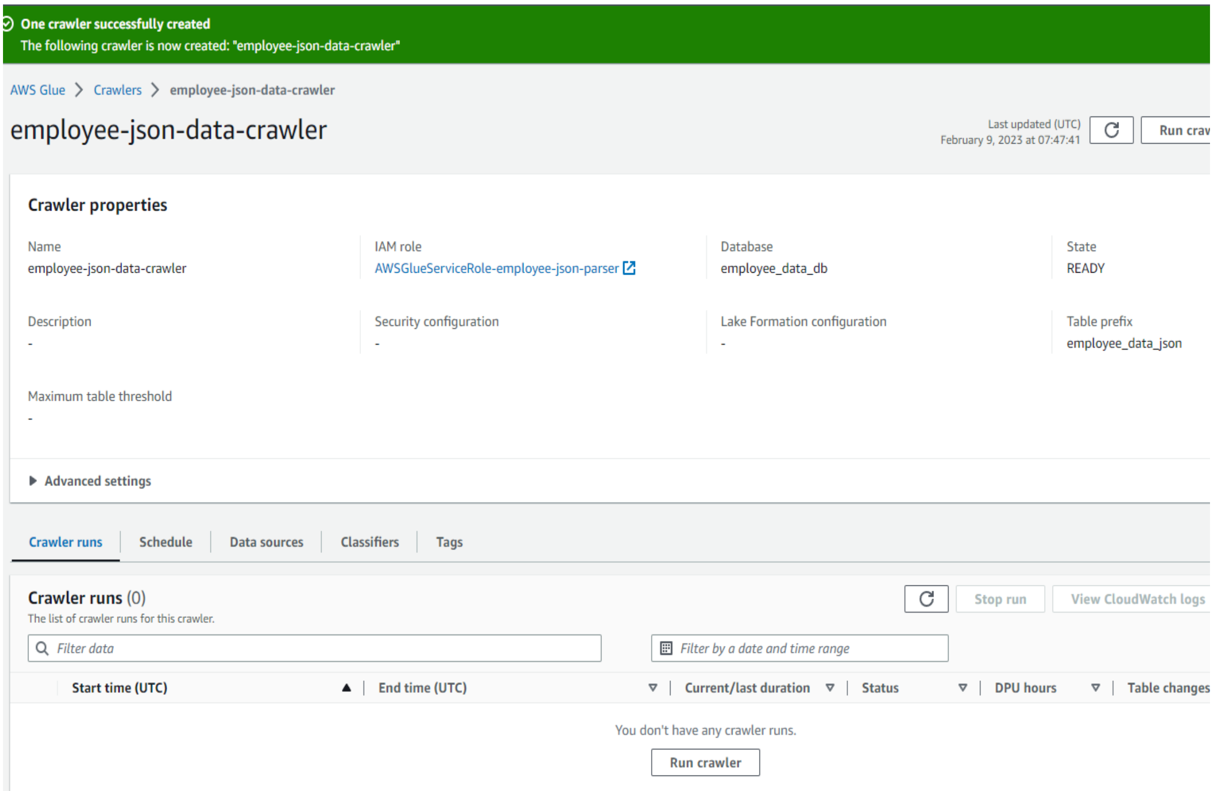
Type a number greater than 0

Crawler schedule

You can define a time-based schedule for your crawlers and jobs in AWS Glue. The definition of these schedules uses the Unix-like cron syntax. Learn more.

Frequency

On demand



One crawler successfully created

The following crawler is now created: "employee-json-data-crawler"

AWS Glue > Crawlers > employee-json-data-crawler

employee-json-data-crawler

Last updated (UTC)
February 9, 2023 at 07:47:41

Crawler properties

Name employee-json-data-crawler	IAM role AWSGlueServiceRole-employee-json-parser	Database employee_data_db	State READY
Description -	Security configuration -	Lake Formation configuration -	Table prefix employee_data_json
Maximum table threshold -			

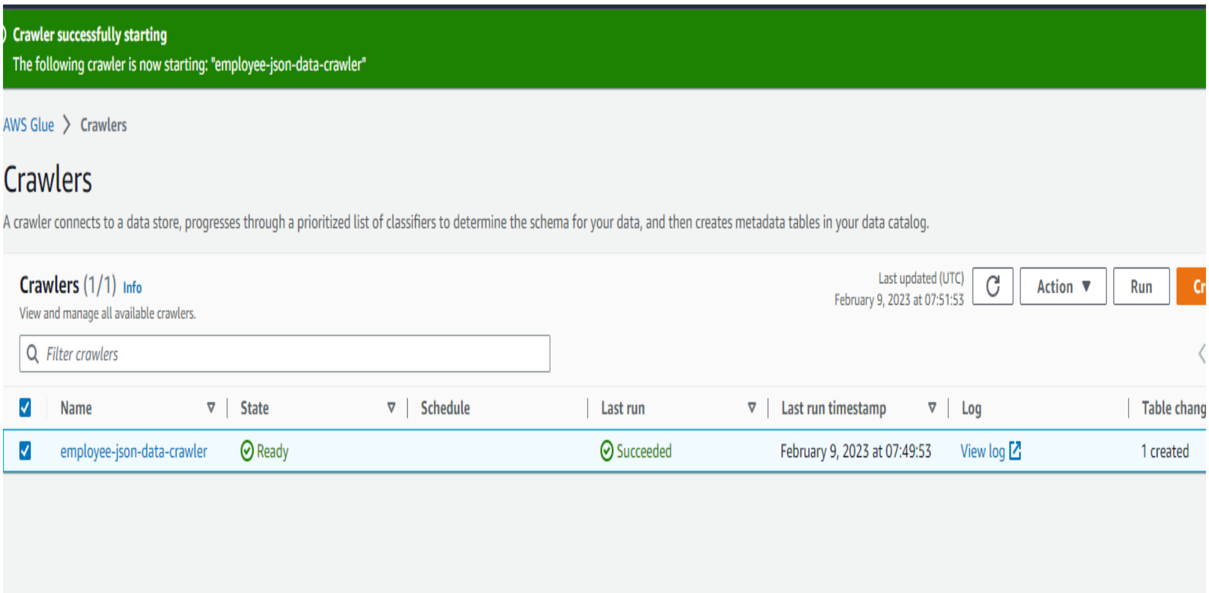
Crawler runs (0)

The list of crawler runs for this crawler.

Start time (UTC)	End time (UTC)	Current/last duration	Status	DPU hours	Table changes

You don't have any crawler runs.

Run crawler:



Crawler successfully starting

The following crawler is now starting: "employee-json-data-crawler"

AWS Glue > Crawlers

Crawlers

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.

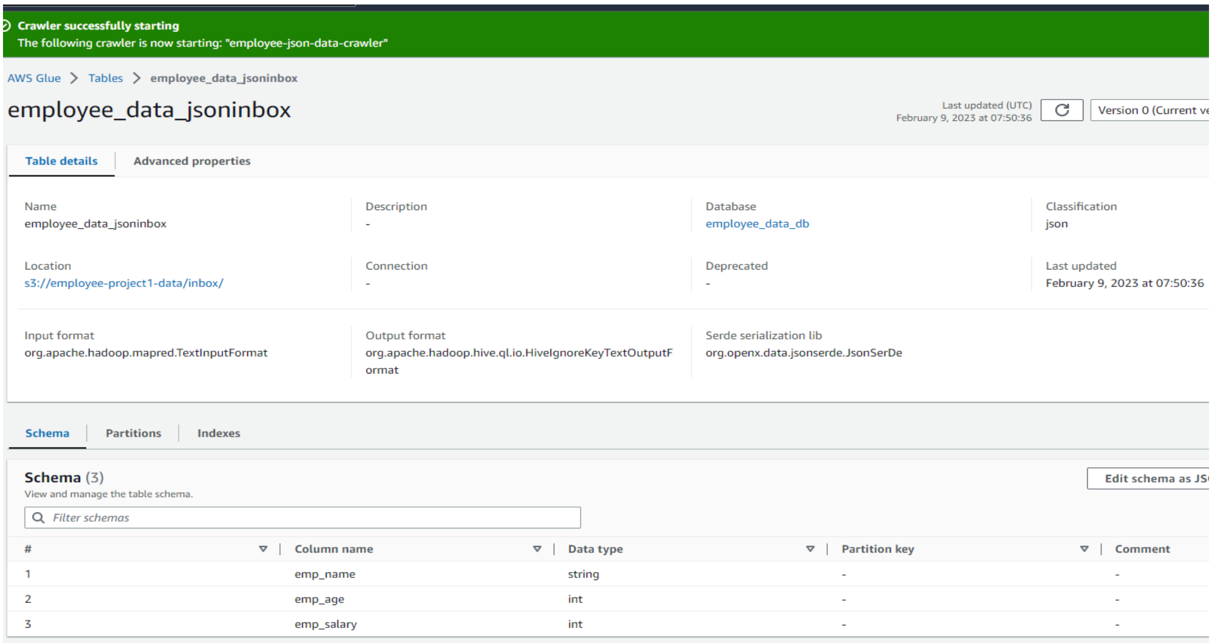
Crawlers (1/1) Info

View and manage all available crawlers.

Last updated (UTC)
February 9, 2023 at 07:51:53

☑	Name	State	Schedule	Last run	Last run timestamp	Log	Table chang
☑	employee-json-data-crawler	✓ Ready		✓ Succeeded	February 9, 2023 at 07:49:53	View log	1 created

This is the table created by crawler:



Crawler successfully starting

The following crawler is now starting: "employee-json-data-crawler"

AWS Glue > Tables > employee_data_jsoninbox

employee_data_jsoninbox

Last updated (UTC)
February 9, 2023 at 07:50:36

Name
employee_data_jsoninbox

Description
-

Database
employee_data_db

Classification
json

Location
s3://employee-project1-data/inbox/

Connection
-

Deprecated
-

Last updated
February 9, 2023 at 07:50:36

Input format
org.apache.hadoop.mapred.TextInputFormat

Output format
org.apache.hadoop.hive.ql.io.HiveIgnoreKeyTextOutputFormat

Serde serialization lib
org.openx.data.jsonserde.JsonSerDe

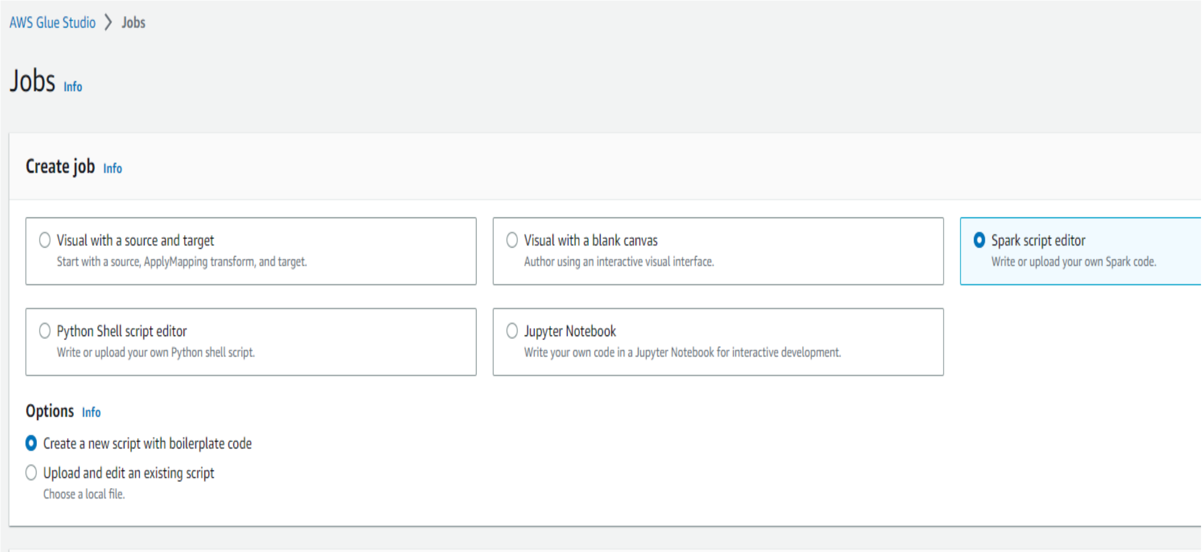
Schema (3)

View and manage the table schema.

#	Column name	Data type	Partition key	Comment
1	emp_name	string	-	-
2	emp_age	int	-	-
3	emp_salary	int	-	-

Glue job:

Select spark script



AWS Glue Studio > Jobs

Jobs Info

Create job Info

○ Visual with a source and target
Start with a source, ApplyMapping transform, and target.

○ Visual with a blank canvas
Author using an interactive visual interface.

● Spark script editor
Write or upload your own Spark code.

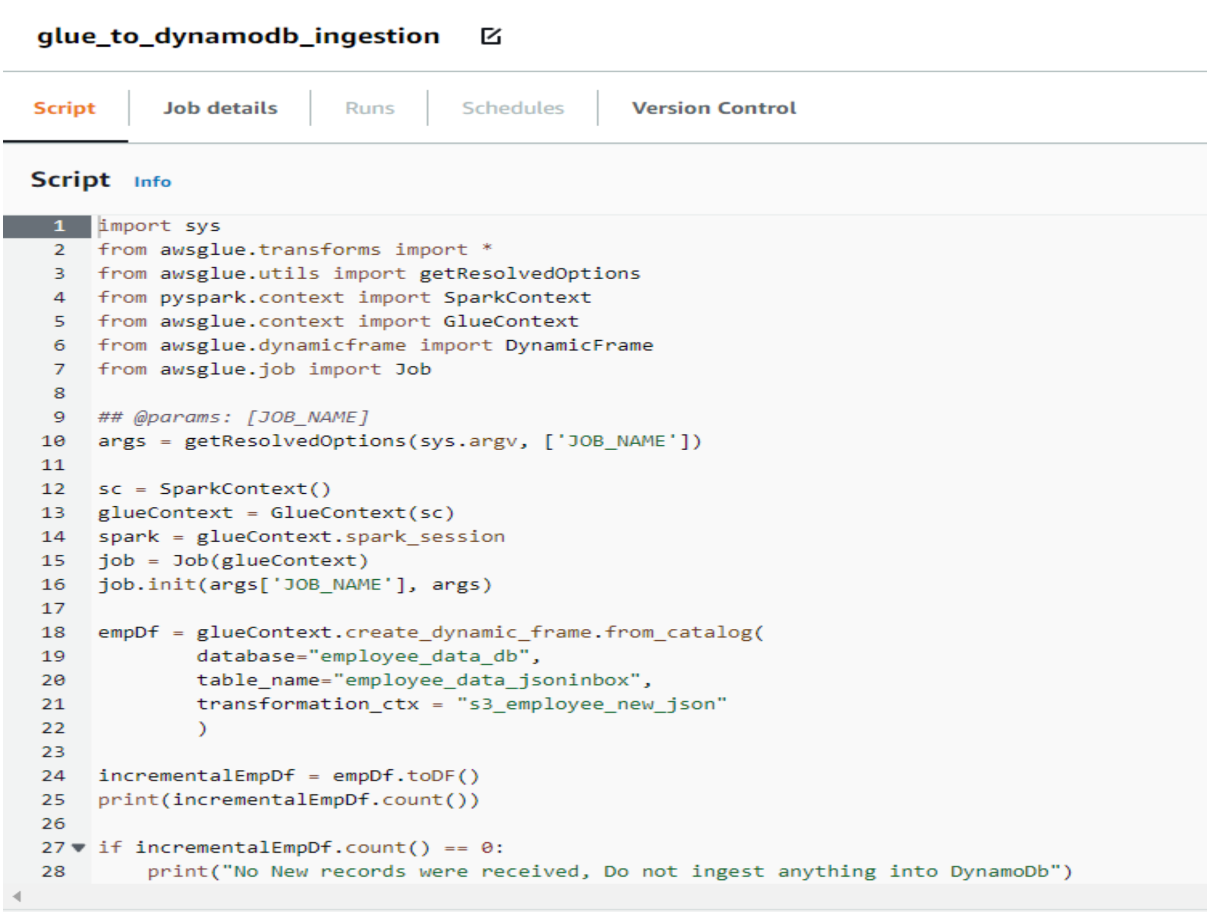
○ Python Shell script editor
Write or upload your own Python shell script.

○ Jupyter Notebook
Write your own code in a Jupyter Notebook for interactive development.

Options Info

● Create a new script with boilerplate code

○ Upload and edit an existing script
Choose a local file.

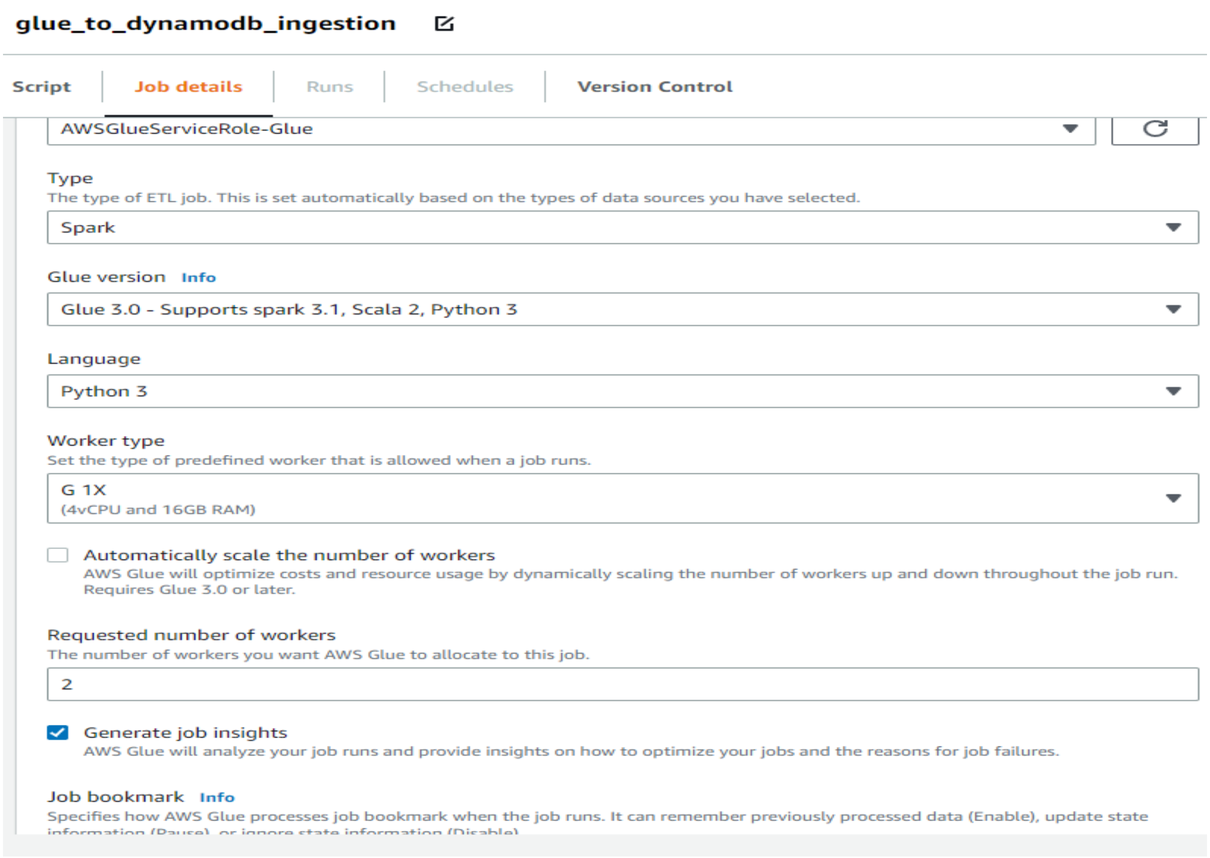


glue_to_dynamodb_ingestion

Script Info

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

## @params: [JOB_NAME]
args = getResolvedOptions(sys.argv, ['JOB_NAME'])

sc = SparkContext()
glueContext = GlueContext(sc)
spark = glueContext.spark_session
job = Job(glueContext)
job.init(args['JOB_NAME'], args)

empDf = glueContext.create_dynamic_frame.from_catalog(
        database="employee_data_db",
        table_name="employee_data_jsoninbox",
        transformation_ctx = "s3_employee_new_json"
        )

incrementalEmpDf = empDf.toDF()
print(incrementalEmpDf.count())

if incrementalEmpDf.count() == 0:
    print("No New records were received, Do not ingest anything into DynamoDb")
```

Enable bookmark: for incremental read



glue_to_dynamodb_ingestion

AWSGlueServiceRole-Glue

Type
The type of ETL job. This is set automatically based on the types of data sources you have selected.
Spark

Glue version Info
Glue 3.0 - Supports spark 3.1, Scala 2, Python 3

Language
Python 3

Worker type
Set the type of predefined worker that is allowed when a job runs.
G 1X
(4vCPU and 16GB RAM)

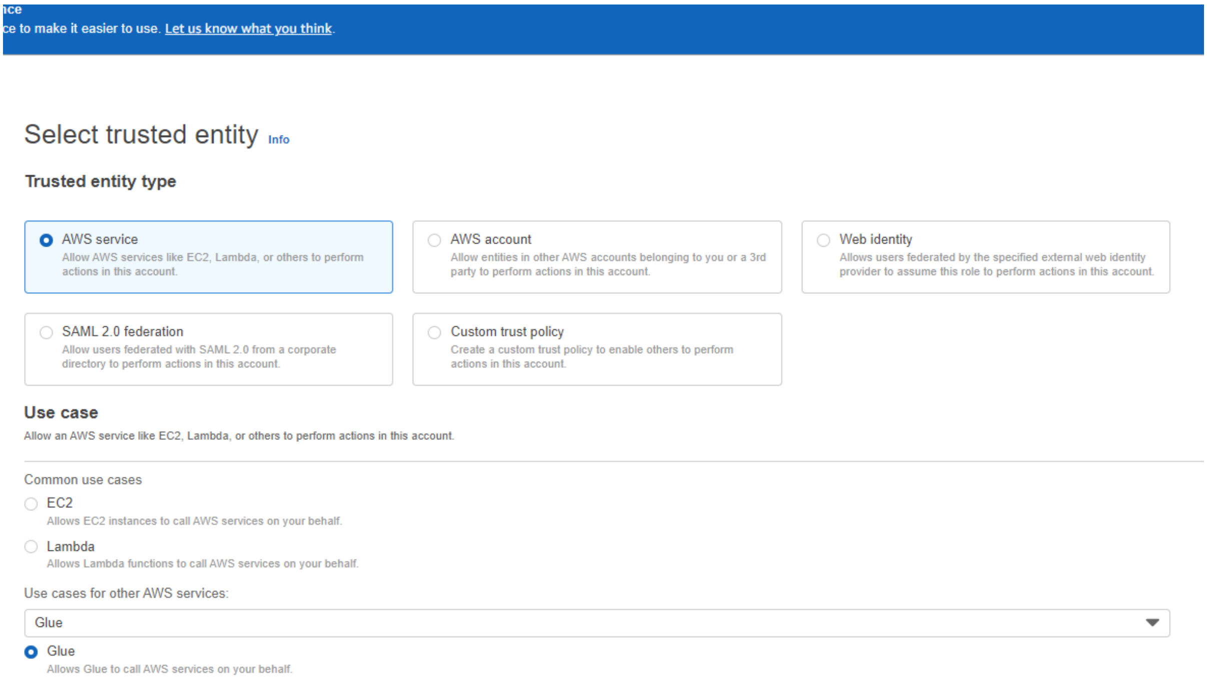
☐ Automatically scale the number of workers
AWS Glue will optimize costs and resource usage by dynamically scaling the number of workers up and down throughout the job run. Requires Glue 3.0 or later.

Requested number of workers
The number of workers you want AWS Glue to allocate to this job.
2

☑ Generate job insights
AWS Glue will analyze your job runs and provide insights on how to optimize your jobs and the reasons for job failures.

Job bookmark Info
Specifies how AWS Glue processes job bookmark when the job runs. It can remember previously processed data (Enable), update state information (Pause), or ignore state information (Disable).

IAM role for glue job:



ice
ce to make it easier to use. Let us know what you think.

Select trusted entity Info

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Use case
Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

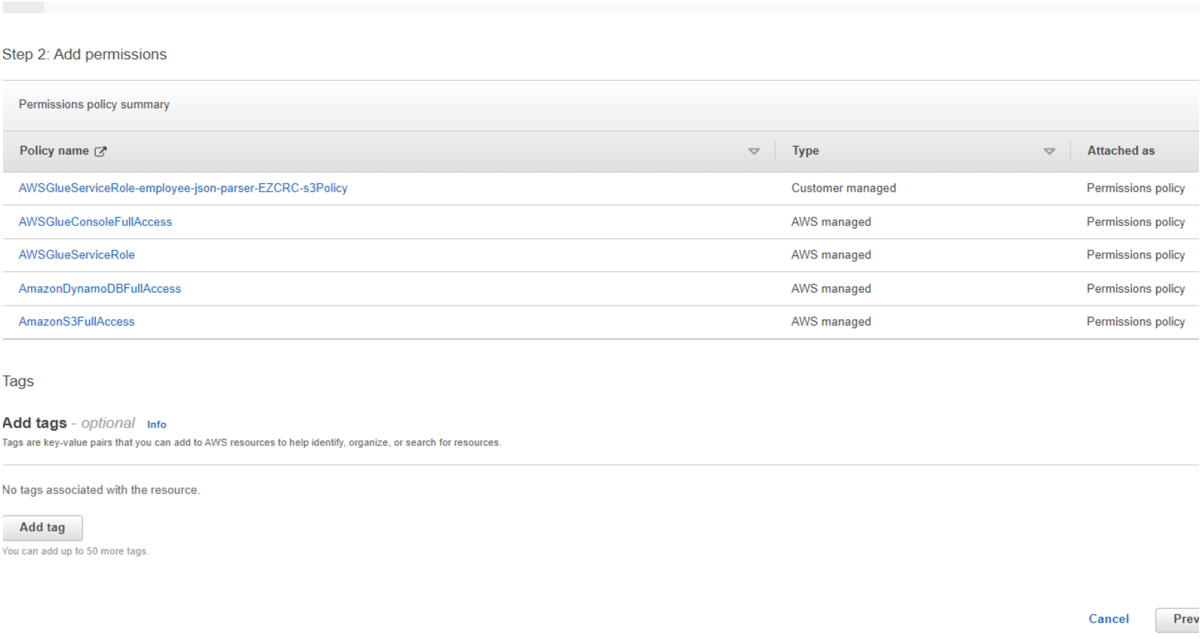
Common use cases

○ EC2
Allows EC2 instances to call AWS services on your behalf.

○ Lambda
Allows Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:
Glue

● Glue
Allows Glue to call AWS services on your behalf.



Step 2: Add permissions

Permissions policy summary

Policy name	Type	Attached as
AWSGlueServiceRole-employee-json-parser-EZCRC-s3Policy	Customer managed	Permissions policy
AWSGlueConsoleFullAccess	AWS managed	Permissions policy
AWSGlueServiceRole	AWS managed	Permissions policy
AmazonDynamoDBFullAccess	AWS managed	Permissions policy
AmazonS3FullAccess	AWS managed	Permissions policy

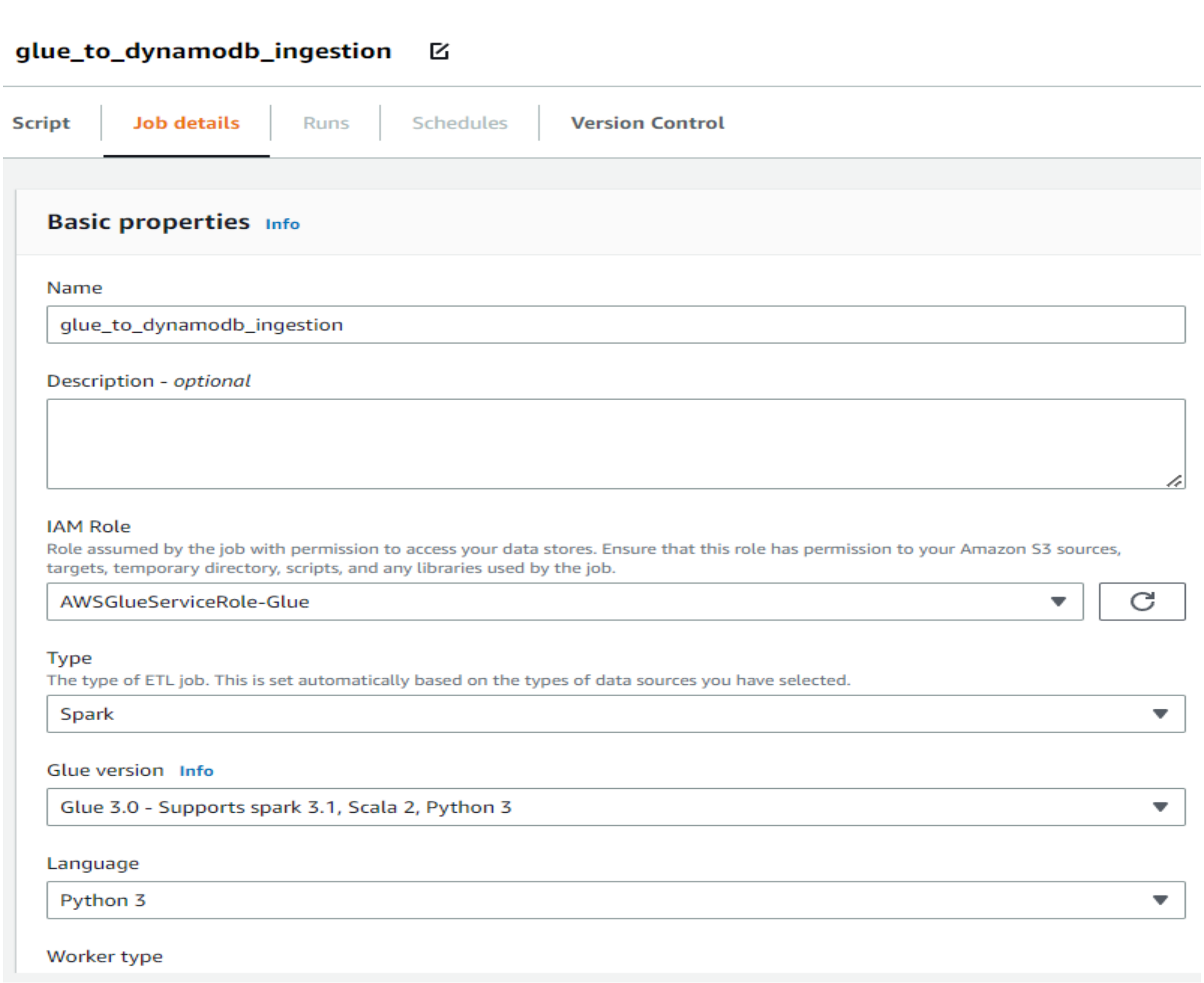
Tags

Add tags - *optional* Info

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.

You can add up to 50 more tags.



glue_to_dynamodb_ingestion

Basic properties Info

Name

glue_to_dynamodb_ingestion

Description - *optional*

IAM Role

Role assumed by the job with permission to access your data stores. Ensure that this role has permission to your Amazon S3 sources, targets, temporary directory, scripts, and any libraries used by the job.

AWSGlueServiceRole-Glue

Type

The type of ETL job. This is set automatically based on the types of data sources you have selected.

Spark

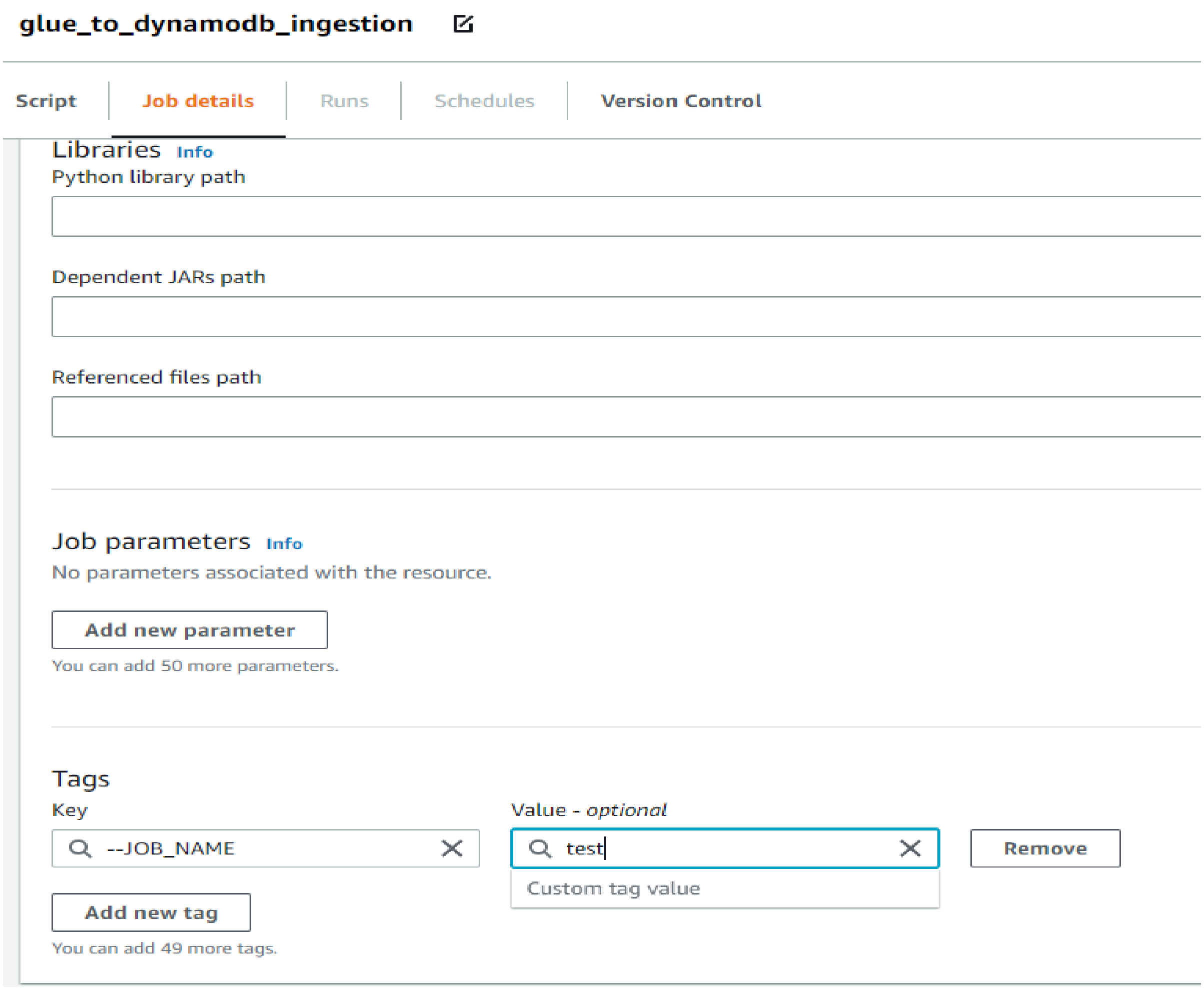
Glue version Info

Glue 3.0 - Supports spark 3.1, Scala 2, Python 3

Language

Python 3

Worker type



glue_to_dynamodb_ingestion

Libraries

Python library path

Dependent JARs path

Referenced files path

Job parameters

No parameters associated with the resource.

You can add 50 more parameters.

Tags

Key: --JOB_NAME

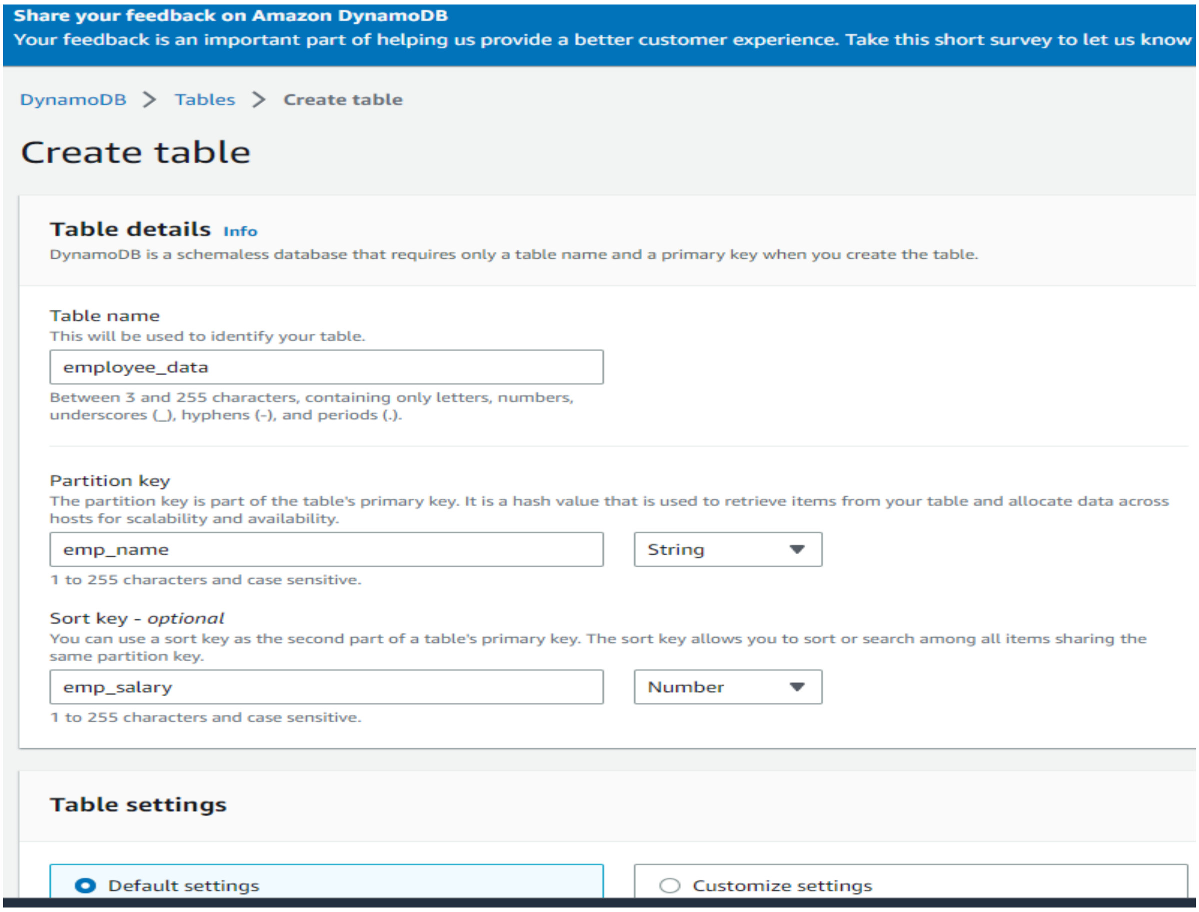
Value - *optional*: test

Custom tag value

You can add 49 more tags.

DynamoDB:

Create DynamoDB table:



Share your feedback on Amazon DynamoDB
Your feedback is an important part of helping us provide a better customer experience. Take this short survey to let us know

DynamoDB > Tables > Create table

Create table

Table details Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name
This will be used to identify your table.
employee_data
Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

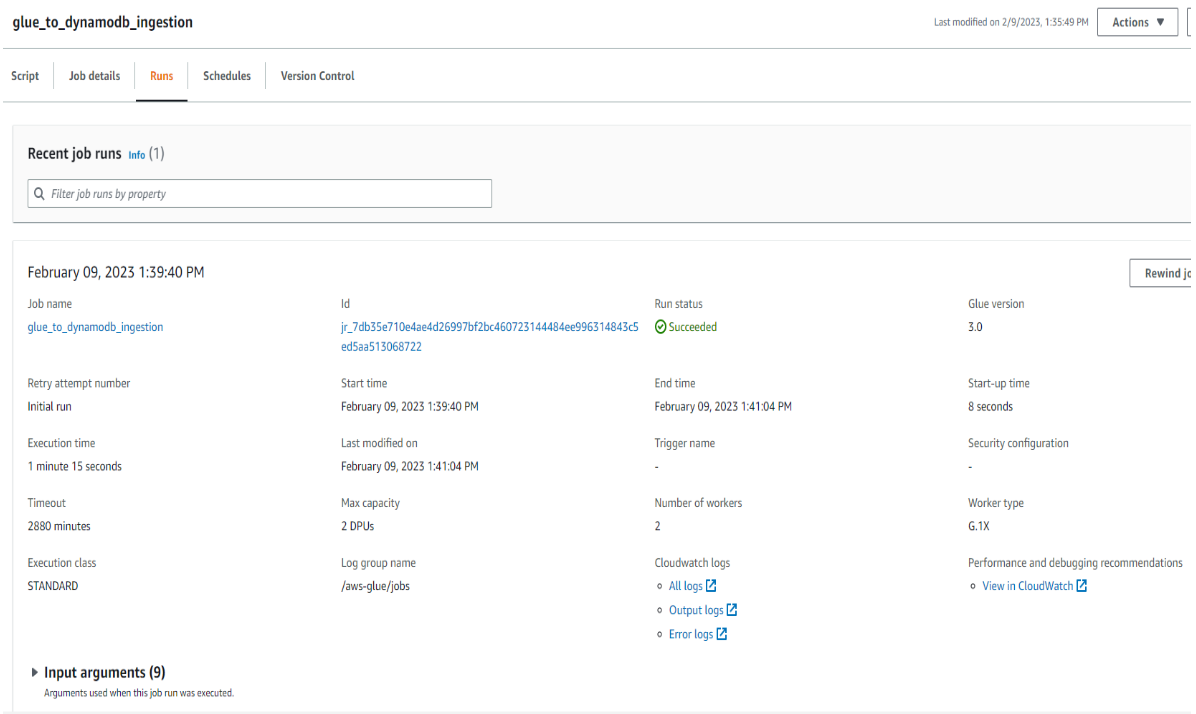
Partition key
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.
emp_name — String ▼
1 to 255 characters and case sensitive.

Sort key - *optional*
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.
emp_salary — Number ▼
1 to 255 characters and case sensitive.

Table settings

● Default settings ○ Customize settings

Run Glue Job:



glue_to_dynamodb_ingestion

Last modified on 2/9/2023, 1:35:49 PM

Recent job runs Info (1)

Filter job runs by property

February 09, 2023 1:39:40 PM

Job name: glue_to_dynamodb_ingestion
Id: jr_7db35e710e4ae4d26997bf2bc460723144484ee996314843c5ed5aa513068722
Run status: ✓ Succeeded
Glue version: 3.0

Retry attempt number: Initial run
Start time: February 09, 2023 1:39:40 PM
End time: February 09, 2023 1:41:04 PM
Start-up time: 8 seconds

Execution time: 1 minute 15 seconds
Last modified on: February 09, 2023 1:41:04 PM
Trigger name: -
Security configuration: -

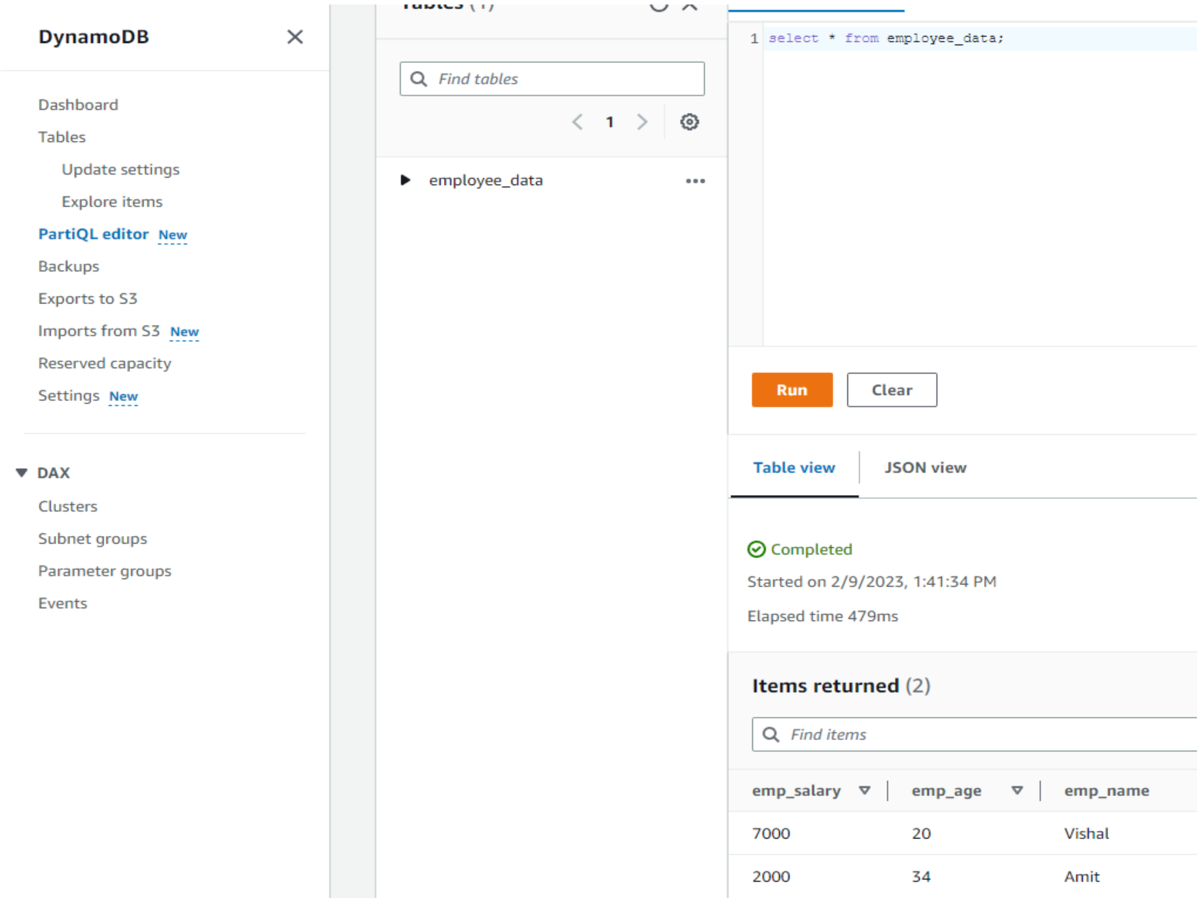
Timeout: 2880 minutes
Max capacity: 2 DPUs
Number of workers: 2
Worker type: G.1X

Execution class: STANDARD
Log group name: /aws-glue/jobs
Cloudwatch logs:
- All logs
- Output logs
- Error logs

Performance and debugging recommendations:
- View in CloudWatch

▶ Input arguments (9)
Arguments used when this job run was executed.

Check the data in DynamoDB:



```
select * from employee_data;
```

Completed

Started on 2/9/2023, 1:41:34 PM

Elapsed time 479ms

Items returned (2)

emp_salary	emp_age	emp_name
7000	20	Vishal
2000	34	Amit