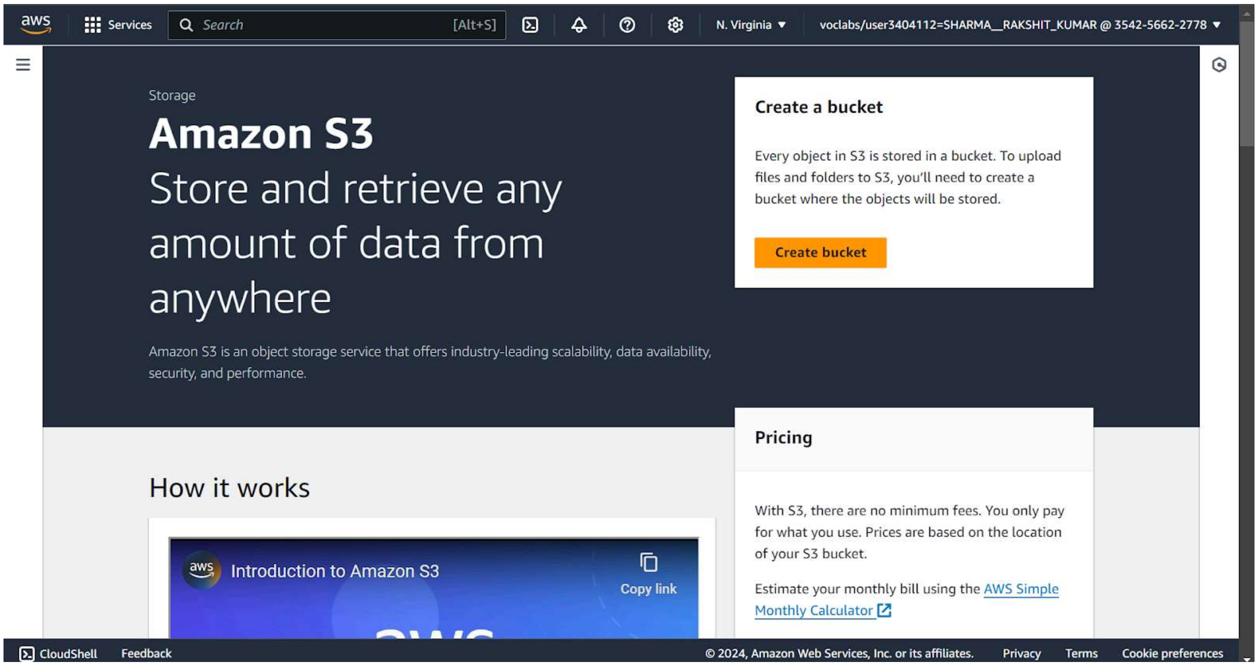
Aim: To create a Lambda function which will log “An Image has been added” once you add an object to a specific bucket in S3

1. Login to your AWS account. Search for S3 service, click on create bucket to make a new bucket.



aws
Services
Search [Alt+S]
N. Virginia ▼
voclabs/user3404112=SHARMA__RAKSHIT_KUMAR @ 3542-5662-2778 ▼

Storage

Amazon S3

Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

Pricing

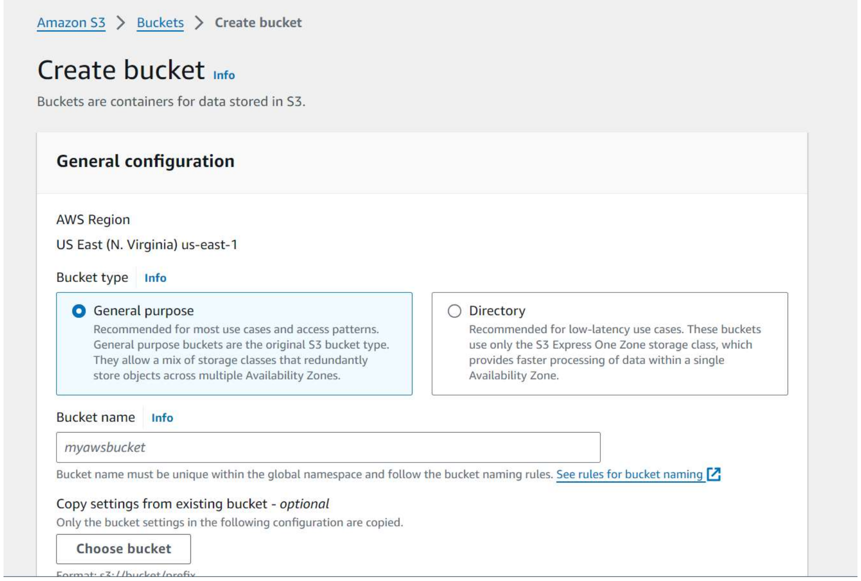
With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the AWS Simple Monthly Calculator

How it works

Introduction to Amazon S3
Copy link
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

2. Select the bucket type as general purpose, give your bucket an appropriate name. Uncheck the block all public access. Keep rest of the settings as default and click on create bucket.



Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region
US East (N. Virginia) us-east-1

Bucket type Info

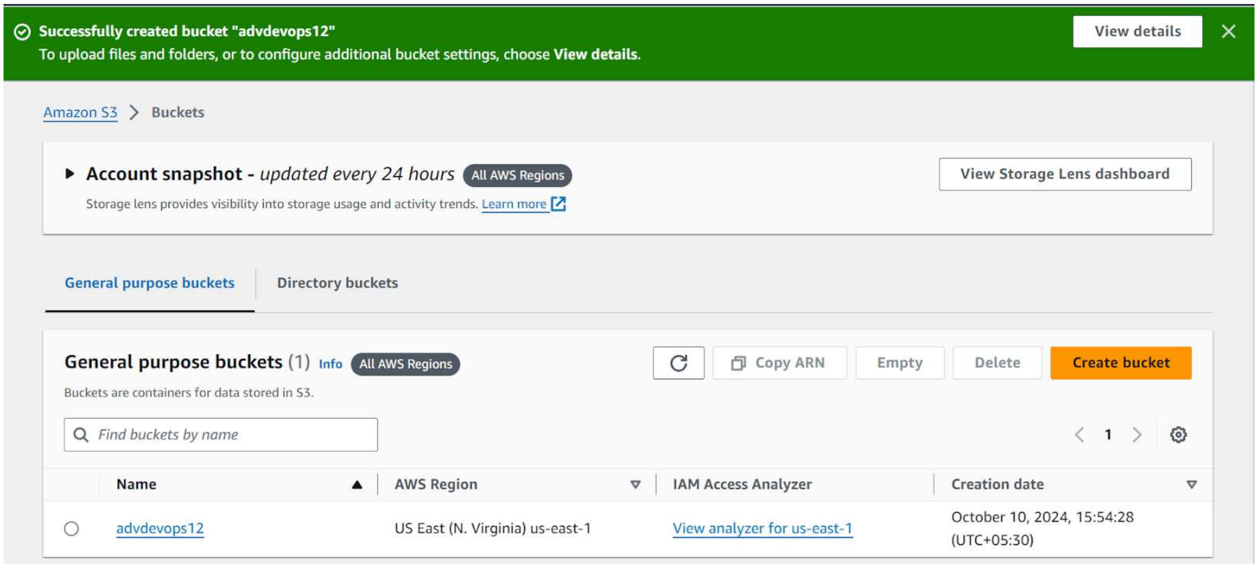
● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info
myawsbucket
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

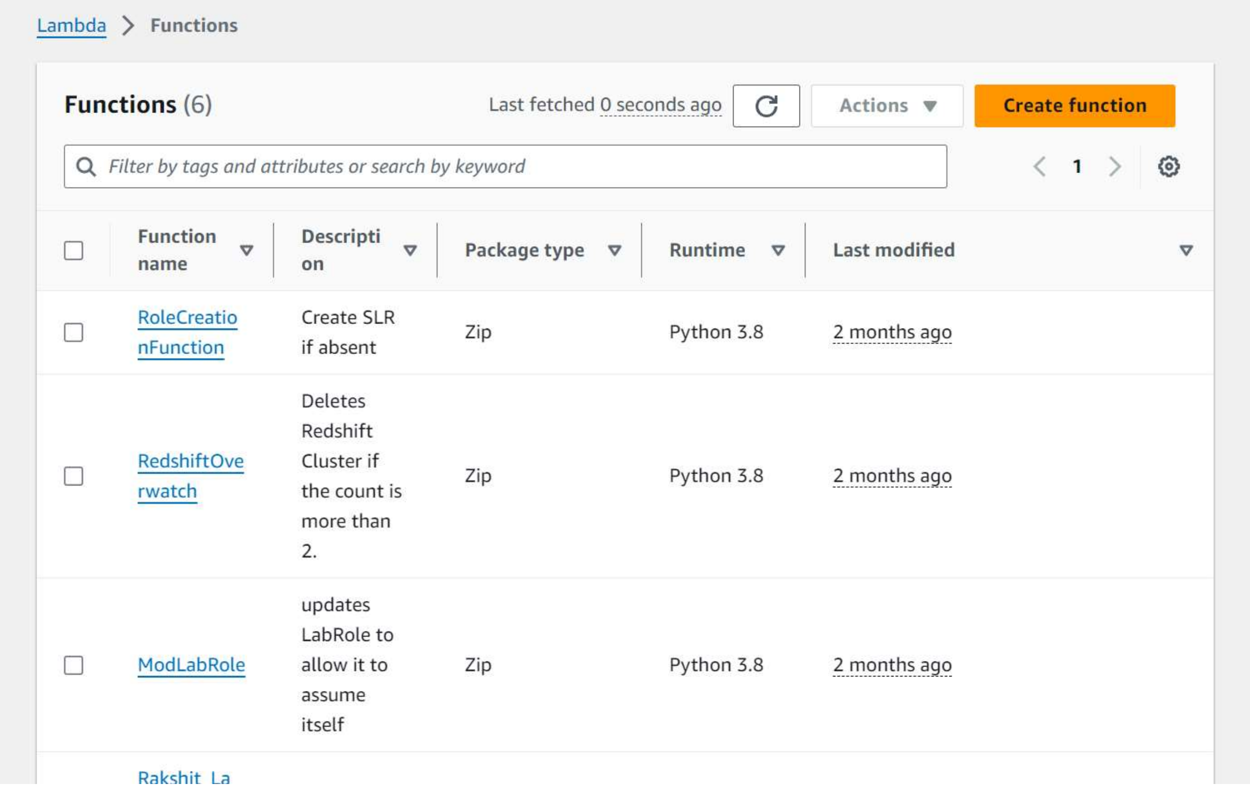


Successfully created bucket "advdevops12"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

View details
Amazon S3 > Buckets
Account snapshot - updated every 24 hours
All AWS Regions
Storage lens provides visibility into storage usage and activity trends. Learn more
View Storage Lens dashboard
General purpose buckets
Directory buckets
General purpose buckets (1) Info
All AWS Regions
Buckets are containers for data stored in S3.
Copy ARN
Empty
Delete
Create bucket
Find buckets by name
1

	Name	AWS Region	IAM Access Analyzer	Creation date
○	advdevops12	US East (N. Virginia) us-east-1	View analyzer for us-east-1	October 10, 2024, 15:54:28 (UTC+05:30)

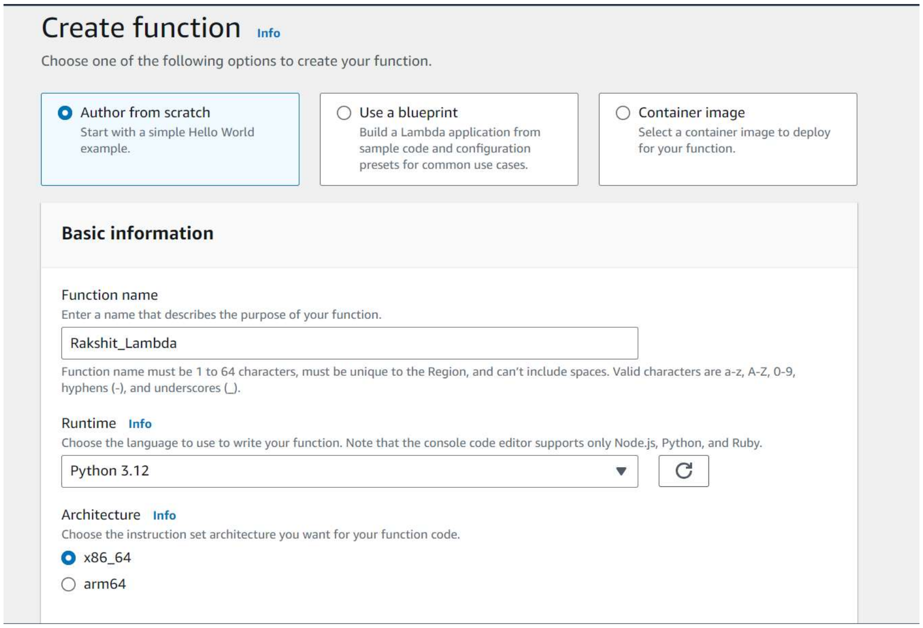
3. Search for lambda and open its console. Click on create function to make a new function.



Lambda > Functions
Functions (6)
Last fetched 0 seconds ago
Actions
Create function
Filter by tags and attributes or search by keyword
1

□	Function name	Description	Package type	Runtime	Last modified
□	RoleCreationFunction	Create SLR if absent	Zip	Python 3.8	2 months ago
□	RedshiftOverwatch	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.8	2 months ago
□	ModLabRole	updates LabRole to allow it to assume itself	Zip	Python 3.8	2 months ago
	Rakshit_La				

4. Give a name to your lambda function. Select the language you want to use to write the functions. We will use Python 3.12, Architecture x86. Select Execution role to Create a new role with basic Lambda permissions.



Create function Info

Choose one of the following options to create your function.

● 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.

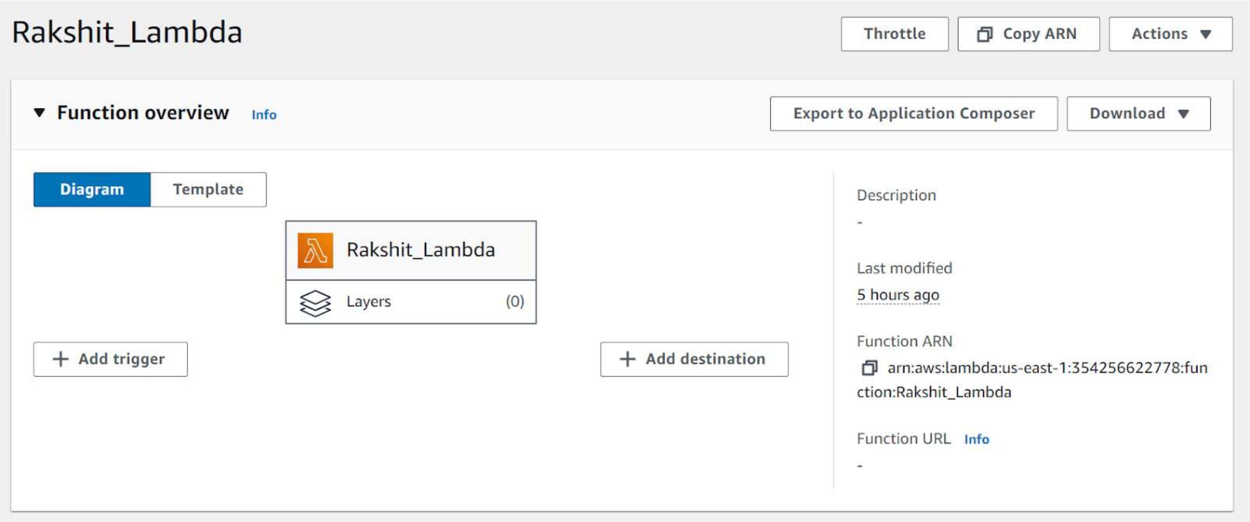
○ Container image
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
Rakshit_Lambda
Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

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 3.12

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



Rakshit_Lambda

Function overview Info

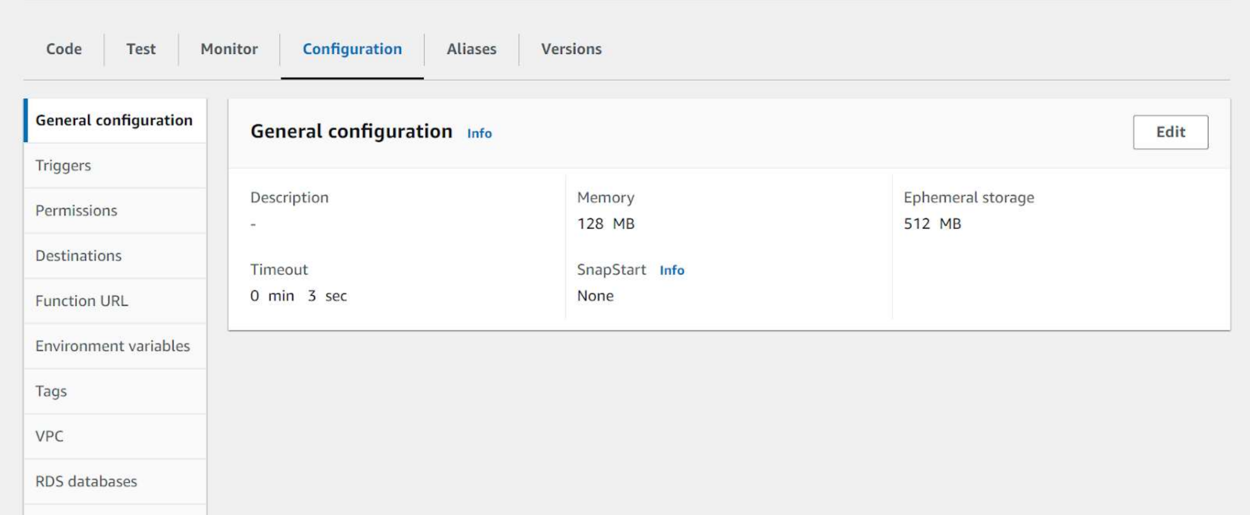
Description
-

Last modified
5 hours ago

Function ARN
arn:aws:lambda:us-east-1:354256622778:function:Rakshit_Lambda

Function URL Info
-

5. Scroll down and go to the configuration tab. In General configuration click on edit to change the configuration.



General configuration

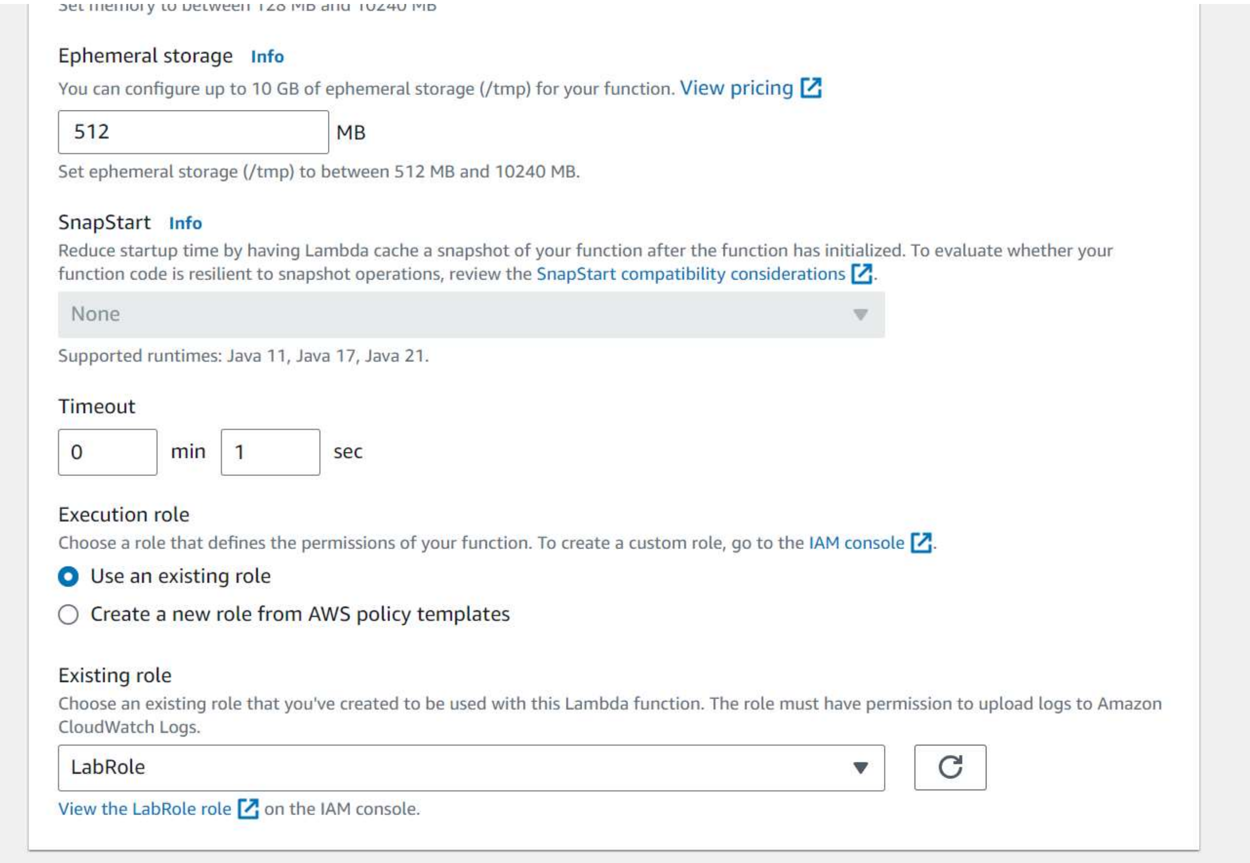
Description: -

Memory: 128 MB

Ephemeral storage: 512 MB

Timeout: 0 min 3 sec

SnapStart: None



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.

SnapStart Info

Reduce startup time by having Lambda cache a snapshot of your function after the function has initialized. To evaluate whether your function code is resilient to snapshot operations, review the SnapStart compatibility considerations.

None

Supported runtimes: Java 11, Java 17, Java 21.

Timeout

0 min 1 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

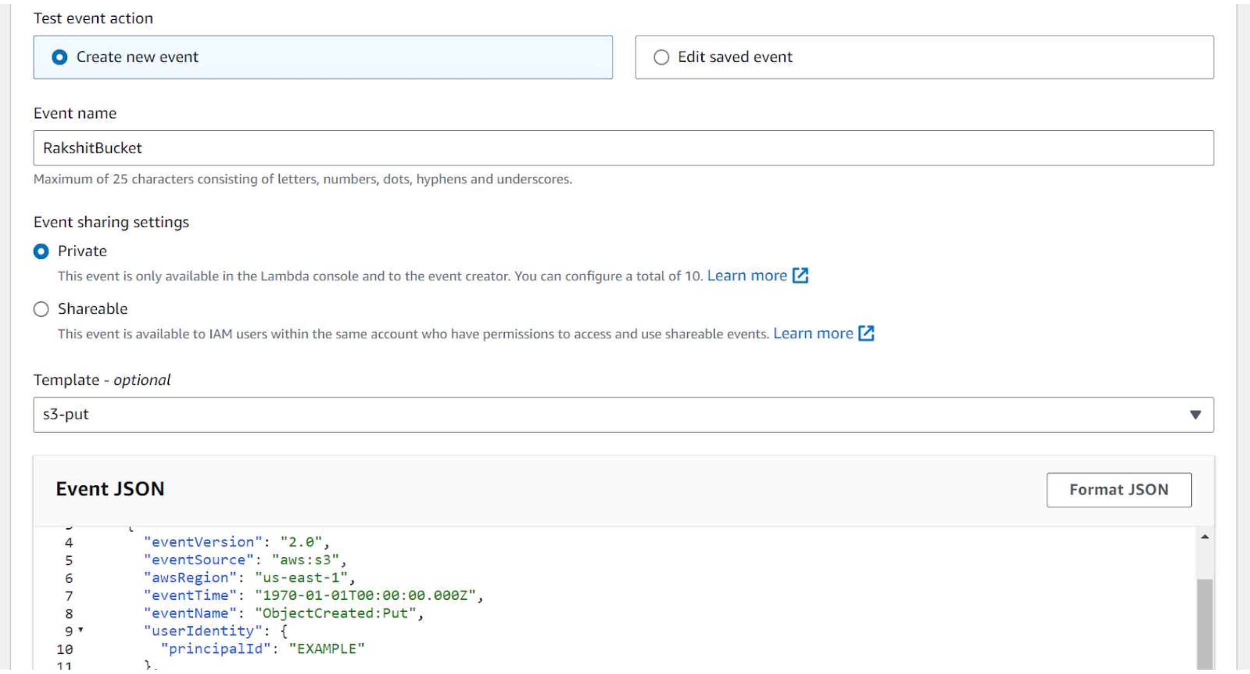
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

LabRole

View the LabRole role on the IAM console.

Here, you can enter a description and change Memory and Timeout. I’ve changed the Timeout period to 1 sec since that is sufficient for now

6. Now to create a new event, go to the test tab. Create a new event, give a name to the event and select Event Sharing to private, and select s3 put template.



Test event action

● Create new event ○ Edit saved event

Event name

RakshitBucket

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

● Private

This event is only available in the Lambda console and to the event creator. You can configure a total of 10. Learn more

○ Shareable

This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

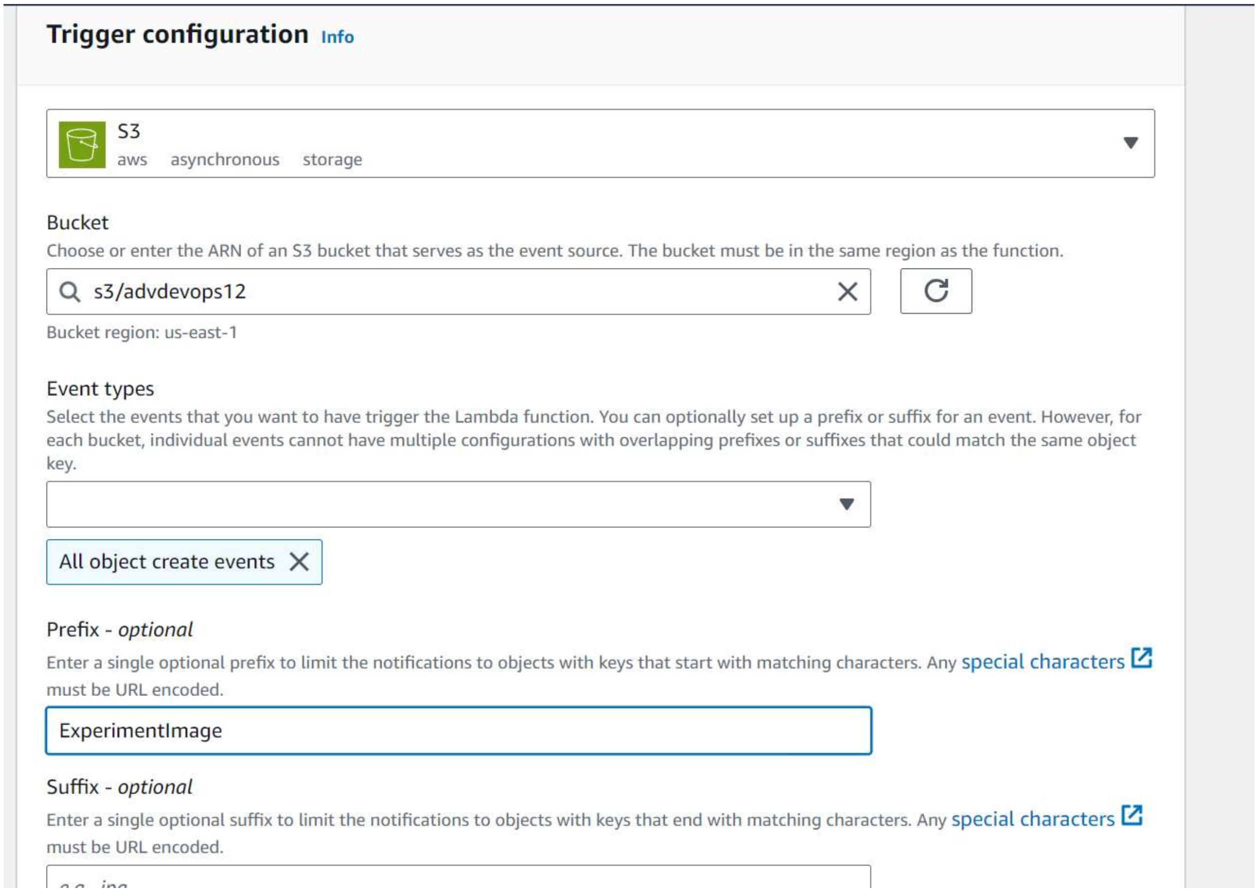
Template - *optional*

s3-put

Event JSON

```
      "eventVersion": "2.0",
      "eventSource": "aws:s3",
      "awsRegion": "us-east-1",
      "eventTime": "1970-01-01T00:00:00.000Z",
      "eventName": "ObjectCreated:Put",
      "userIdentity": {
        "principalId": "EXAMPLE"
      },
```

7. Now in the lambda function click on add trigger.



Trigger configuration Info

S3
aws asynchronous storage

Bucket

Choose or enter the ARN of an S3 bucket that serves as the event source. The bucket must be in the same region as the function.

s3/advdevops12

Bucket region: us-east-1

Event types

Select the events that you want to have trigger the Lambda function. You can optionally set up a prefix or suffix for an event. However, for each bucket, individual events cannot have multiple configurations with overlapping prefixes or suffixes that could match the same object key.

All object create events ✕

Prefix - *optional*

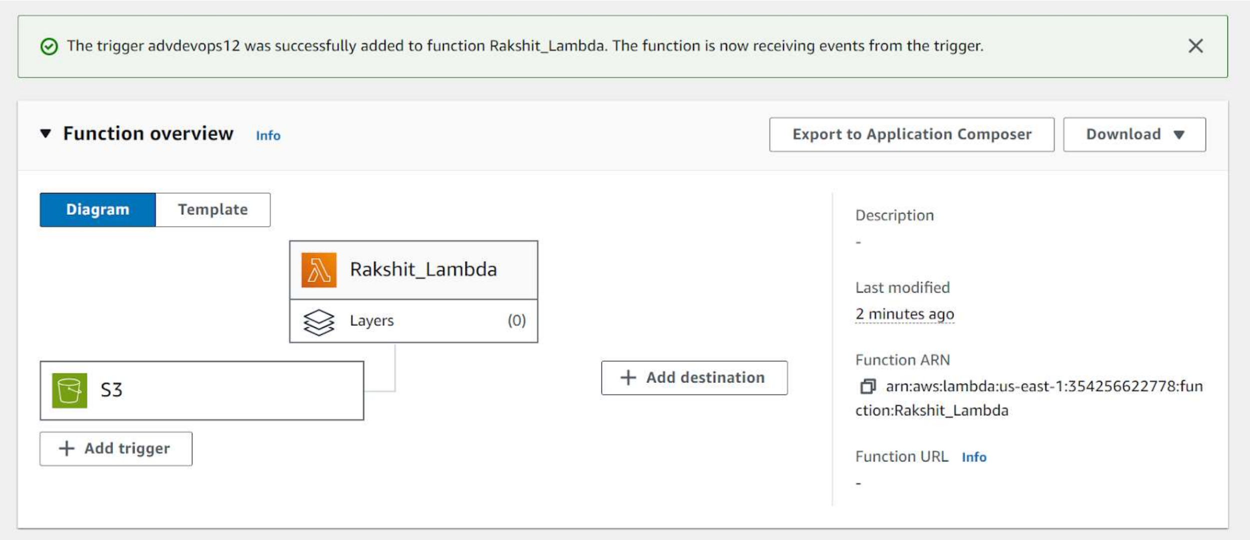
Enter a single optional prefix to limit the notifications to objects with keys that start with matching characters. Any special characters must be URL encoded.

ExperimentImage

Suffix - *optional*

Enter a single optional suffix to limit the notifications to objects with keys that end with matching characters. Any special characters must be URL encoded.

Select the source as s3, and search for the bucket that we created earlier. If you want you can add the prefix for the image. Keep the rest of the setting as default and add trigger.



The trigger advdevops12 was successfully added to function Rakshit_Lambda. The function is now receiving events from the trigger.

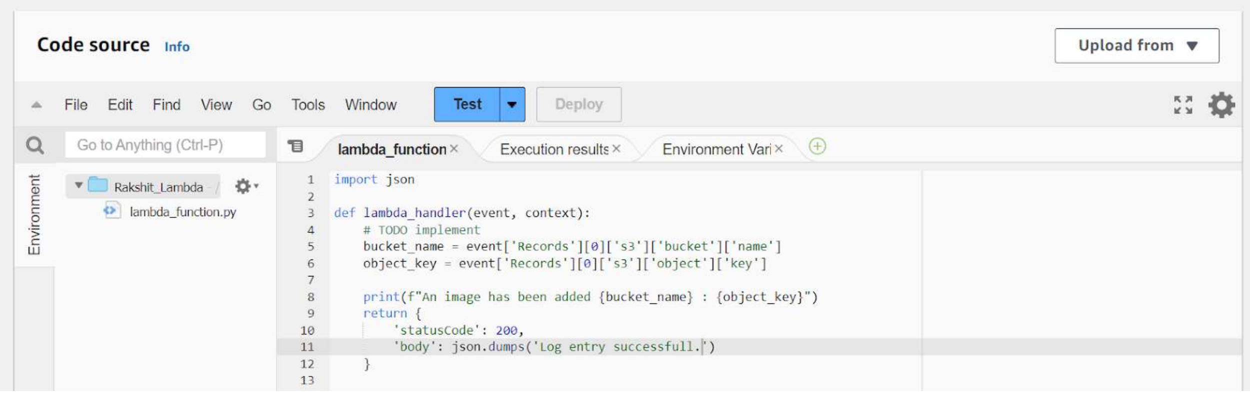
Description
-

Last modified
2 minutes ago

Function ARN
arn:aws:lambda:us-east-1:354256622778:function:Rakshit_Lambda

Function URL Info
-

8. Now Write code that logs a message like “An Image has been added” when triggered. Save the file and click on deploy.



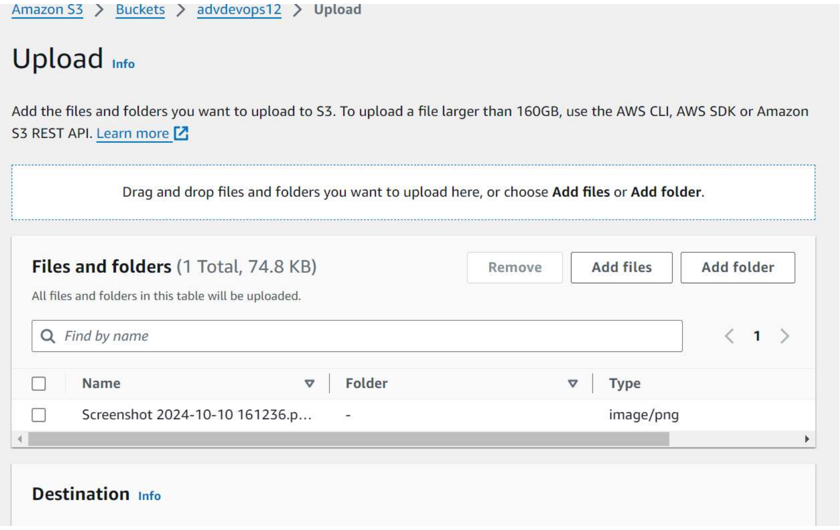
```python
import json

def lambda_handler(event, context):
    # TODO implement
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    object_key = event['Records'][0]['s3']['object']['key']

    print(f"An image has been added {bucket_name} : {object_key}")
    return {
        'statusCode': 200,
        'body': json.dumps('Log entry successfull.')
    }

```

9. Now we will upload an image to our bucket.



Amazon S3 > Buckets > advdevops12 > Upload

Upload Info

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

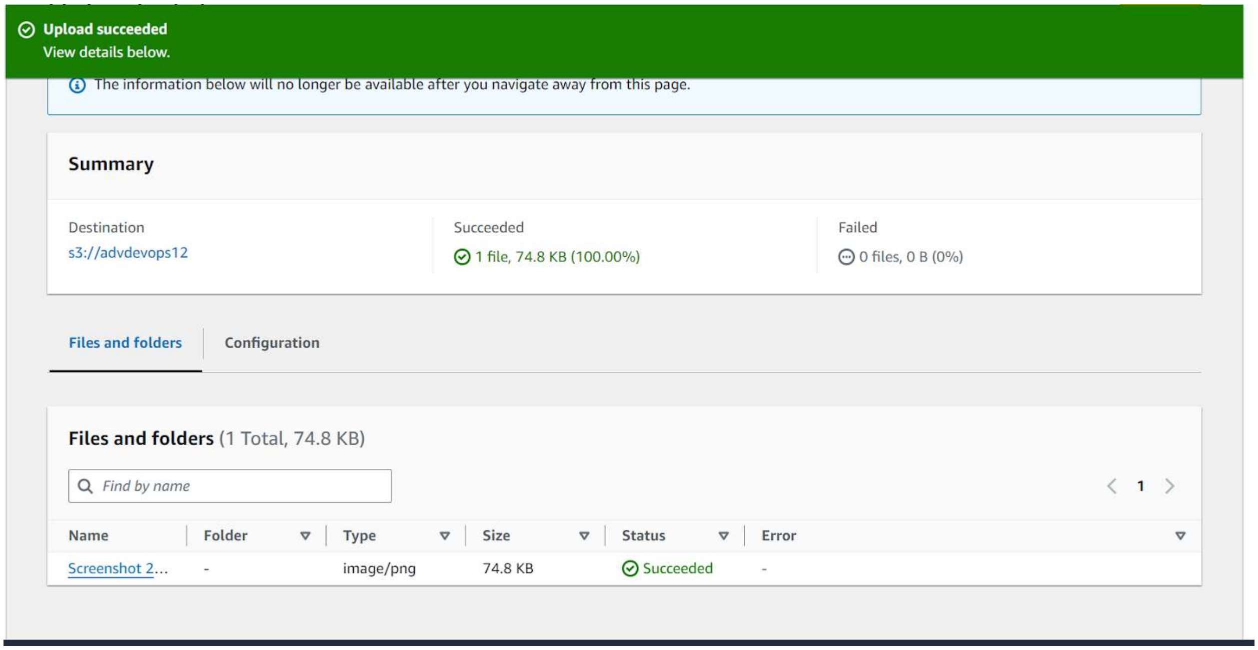
Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 Total, 74.8 KB)

All files and folders in this table will be uploaded.

□	Name	Folder	Type
□	Screenshot 2024-10-10 161236.p...	-	image/png

Destination Info



Upload succeeded
View details below.

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

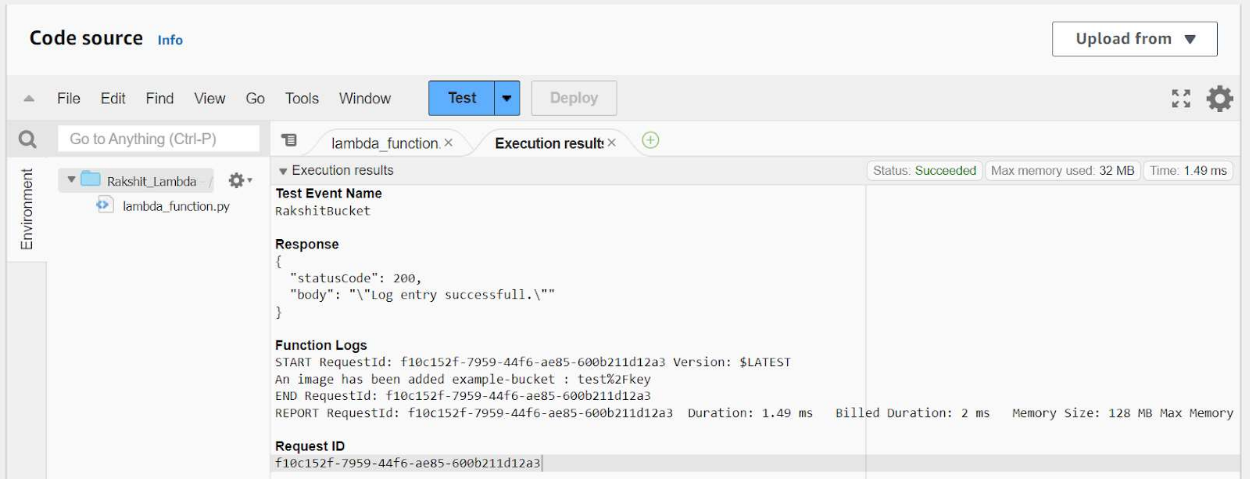
Summary

Destination	Succeeded	Failed
s3://advdevops12	✓ 1 file, 74.8 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 Total, 74.8 KB)

Name	Folder	Type	Size	Status	Error
Screenshot 2...	-	image/png	74.8 KB	✓ Succeeded	-

10. Now in lambda function, select the event which we created now and click on test. Check if it is logging the string that we added in our code.



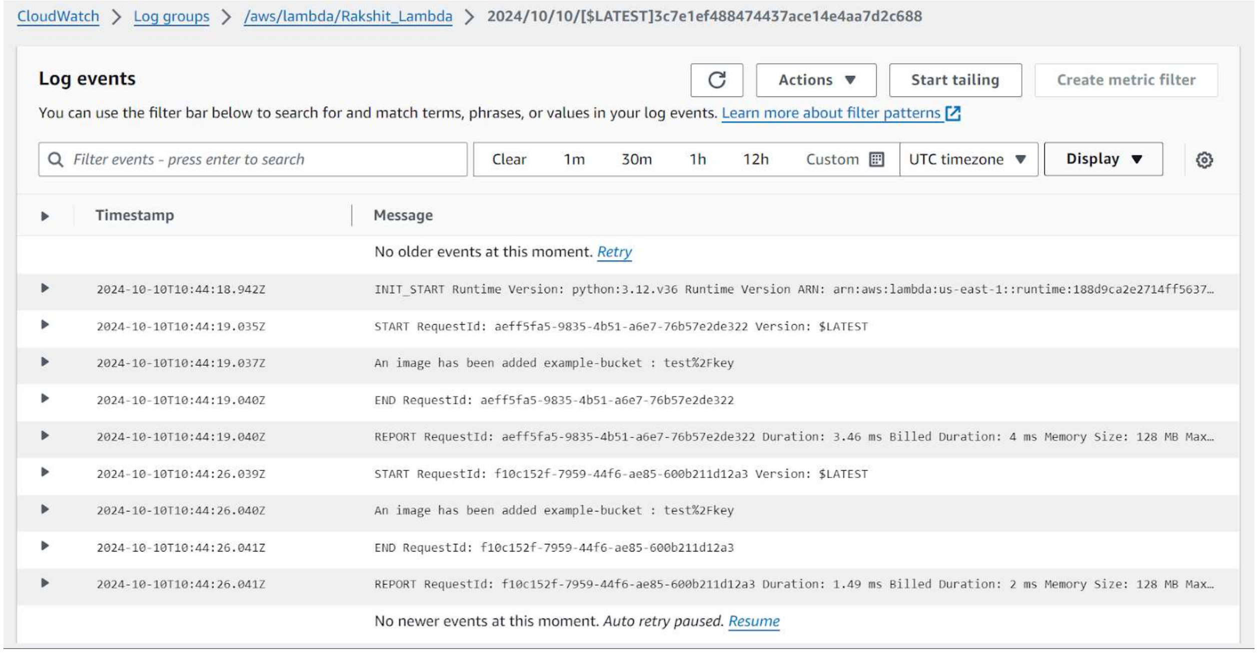
Test Event Name
```
RakshitBucket
```

Response
```
{
  "statusCode": 200,
  "body": "\"Log entry successfull.\""
}
```

Function Logs
```
START RequestId: f10c152f-7959-44f6-ae85-600b211d12a3 Version: $LATEST
An image has been added example-bucket : test%2Fkey
END RequestId: f10c152f-7959-44f6-ae85-600b211d12a3
REPORT RequestId: f10c152f-7959-44f6-ae85-600b211d12a3  Duration: 1.49 ms   Billed Duration: 2 ms   Memory Size: 128 MB Max Memory
```

Request ID
```
f10c152f-7959-44f6-ae85-600b211d12a3
```

11. Now Lets see the log on Cloud watch.To see it go to monitor section and then click on view cloudwatch logs.



CloudWatch > Log groups > /aws/lambda/Rakshit_Lambda > 2024/10/10/[$LATEST]3c7e1ef488474437ace14e4aa7d2c688
Log events
Actions ▼
Start tailing
Create metric filter
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 - press enter to search
Clear
1m
30m
1h
12h
Custom
UTC timezone ▼
Display ▼

Timestamp	Message
	No older events at this moment. *Retry*
2024-10-10T10:44:18.942Z	INIT_START Runtime Version: python:3.12.v36 Runtime Version ARN: arn:aws:lambda:us-east-1::runtime:188d9ca2e2714ff5637…
2024-10-10T10:44:19.035Z	START RequestId: aeff5fa5-9835-4b51-a6e7-76b57e2de322 Version: $LATEST
2024-10-10T10:44:19.037Z	An image has been added example-bucket : test%2Fkey
2024-10-10T10:44:19.040Z	END RequestId: aeff5fa5-9835-4b51-a6e7-76b57e2de322
2024-10-10T10:44:19.040Z	REPORT RequestId: aeff5fa5-9835-4b51-a6e7-76b57e2de322 Duration: 3.46 ms Billed Duration: 4 ms Memory Size: 128 MB Max…
2024-10-10T10:44:26.039Z	START RequestId: f10c152f-7959-44f6-ae85-600b211d12a3 Version: $LATEST
2024-10-10T10:44:26.040Z	An image has been added example-bucket : test%2Fkey
2024-10-10T10:44:26.041Z	END RequestId: f10c152f-7959-44f6-ae85-600b211d12a3
2024-10-10T10:44:26.041Z	REPORT RequestId: f10c152f-7959-44f6-ae85-600b211d12a3 Duration: 1.49 ms Billed Duration: 2 ms Memory Size: 128 MB Max…
	No newer events at this moment. *Auto retry paused.* *Resume*

Conclusion:

In this experiment, we successfully created a Lambda function that logs "An Image has been added" when an object is uploaded to a specific S3 bucket. We started by creating an S3 bucket and setting up a Lambda function with Python 3.12. We configured a trigger for the Lambda function to monitor the S3 bucket and added code to log the message when an image is uploaded. After deploying the code and testing it by uploading an image, we verified the logs in CloudWatch. This experiment demonstrated the use of AWS Lambda for event-driven automation, specifically integrating Lambda with S3.