

Name: Akanksha Shinde Class: D15C Roll No: 53

Experiment 12

1) Create an S3 bucket .

Amazon S3 > Buckets > 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](#)

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

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

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 ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Services

Search

[Alt+S]

Successfully created bucket "komal-akankshabucket"
To upload files and folders, or to configure additional bucket settings, choose [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](#)

General purpose buckets

Directory buckets

General purpose buckets (1) Info All AWS Regions

Buckets are containers for data stored in S3.

Find buckets by name

Name	AWS Region	IAM Access Analyzer
<input type="radio"/> komal-akankshabucket	US East (N. Virginia) us-east-1	View analyzer for us-east-1

2) Create a Lambda Function.

Lambda > Functions > komal-akanksha

komal-akanksha

Function overview Info

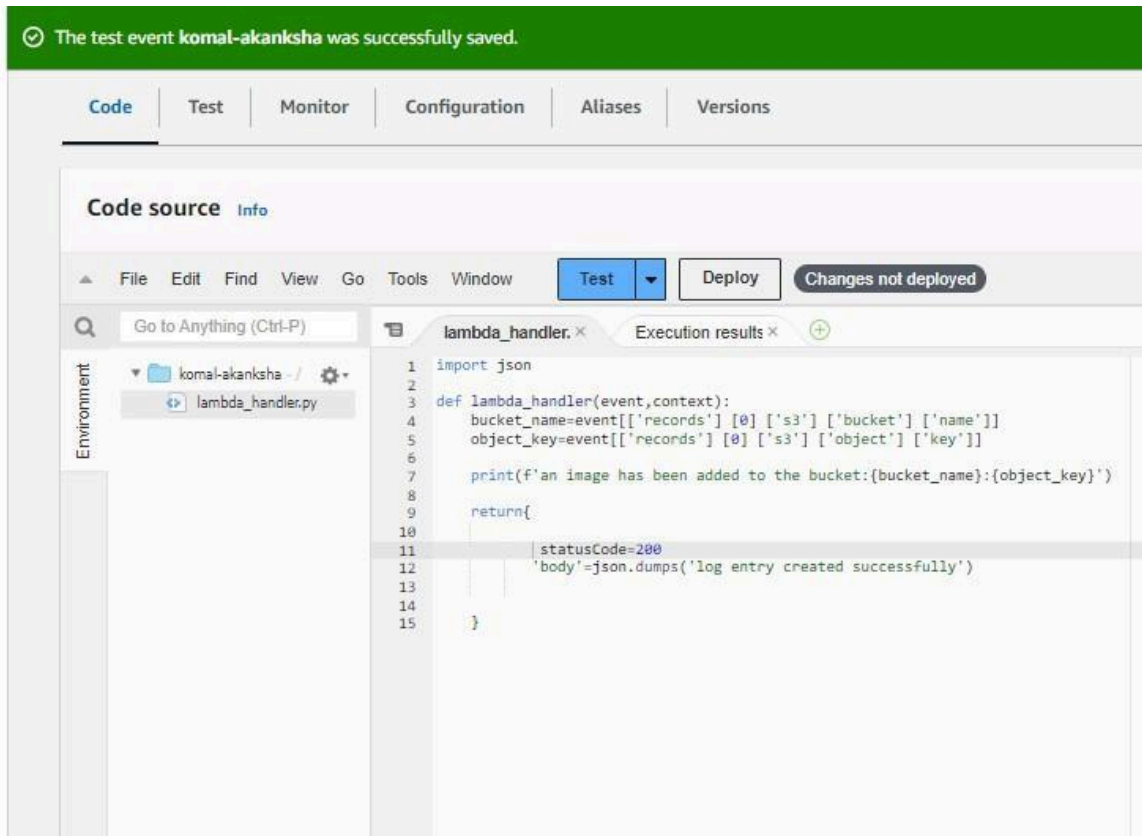
Diagram

Template

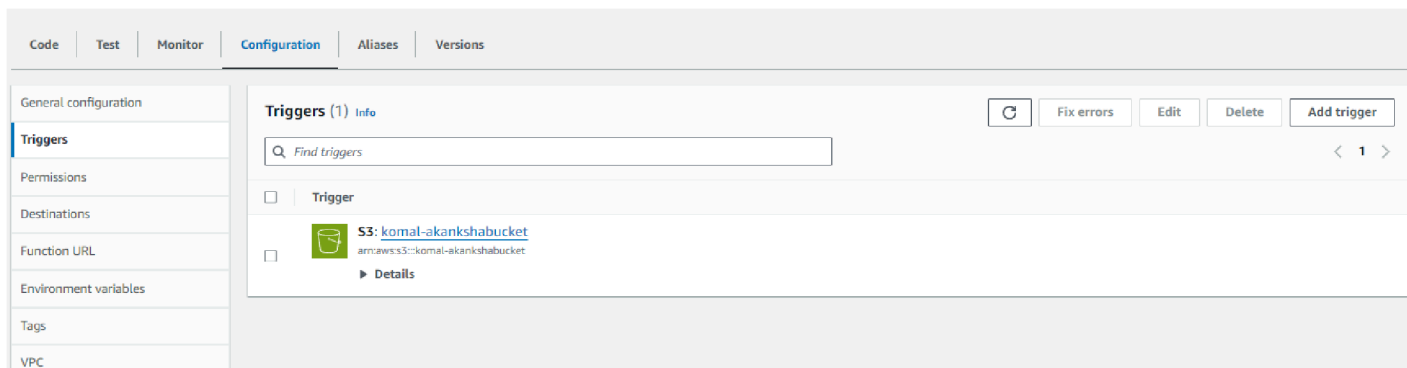
komal-akanksha

Layers (0)

+ Add trigger



3) Add triggers.



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.

✕ ↺

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](#).

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](#).

Recursive invocation

If your function writes objects to an S3 bucket, ensure that you are using different S3 buckets for input and output. Writing to the same bucket increases the risk of creating a recursive invocation, which can result in increased Lambda usage and increased costs. [Learn more](#)

- ☐ I acknowledge that using the same S3 bucket for both input and output is not recommended and that this configuration can cause recursive invocations, increased Lambda usage, and increased costs.

Lambda will add the necessary permissions for AWS S3 to invoke your Lambda function from this trigger. [Learn more](#) [about the Lambda permissions model](#).

[Cancel](#)

[Add](#)

4)Go to S3 and upload an image.

Amazon S3 > Buckets > komal-akankshabucket

komal-akankshabucket [Info](#)

[Objects](#) | [Properties](#) | [Permissions](#) | [Metrics](#) | [Management](#) | [Access Points](#)

Objects (0) [Info](#) [Refresh](#) [Copy S3 URI](#) [Copy URL](#) [Download](#) [Open](#)

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
No objects			
You don't have any objects in this bucket.			
Upload			

[Upload succeeded](#)
View details below.

Upload: status

[Info](#) The information below will no longer be available after you navigate away from this page.

Summary

Destination s3://komal-akankshabucket	Succeeded 1 file, 67.8 KB (100.00%)
--	--

[Files and folders](#) | [Configuration](#)

Files and folders (1 Total, 67.8 KB)

Name	Folder	Type	Size	Status	Error
prime.jpg	-	image/jpeg	67.8 KB	Succeeded	-

5) Check the log for Lambda Function.

CloudWatch > Log groups > /aws/lambda/komal-akanksha > 2024/10/07/[\$LATEST]4e4b07ea331e4a29b56bdfaea66affc8

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](#)

Q Filter events - press enter to search Clear 1m 30m 1h 12h

Timestamp	Message
No older events at this moment. Retry	
2024-10-07T11:25:03.572Z	INIT_START Runtime Version: python:3.9.v52 Runtime Version ARN: arn:aws:lambda:us-east-1::runtime:4b9806e1cdd0f084da9f06b0dce167a8f7569f1e856ebdae2e14f44bbaa6999
2024-10-07T11:25:03.670Z	START RequestId: a654b832-58e6-4bec-8b66-5998786c816d Version: \$LATEST
2024-10-07T11:25:03.687Z	END RequestId: a654b832-58e6-4bec-8b66-5998786c816d
2024-10-07T11:25:03.687Z	REPORT RequestId: a654b832-58e6-4bec-8b66-5998786c816d Duration: 17.14 ms Billed Duration: 18 ms Memory Size: 128 MB Max Memory Used: 30 MB Init Duration: 96.51 ms
No newer events at this moment. Auto retry paused. Resume	

Conclusion:

Integrating AWS Lambda with S3 enables automated, real-time processing of events like file uploads. In this scenario, a Lambda function is set up to log a message whenever an image is uploaded to a designated S3 bucket. This integration highlights the advantages of serverless computing by automating processes without the need for manual involvement or server management. By using AWS Lambda, developers can create event-driven workflows that scale efficiently, reducing operational complexity and enabling rapid deployment of solutions that react to specific events within cloud-based environments.