# Advanced DevOps Lab
# Experiment 12

**Name: Yash Rahate**
**Class: D15B**
**Roll No.: 48**

## Aim:

To create an AWS Lambda function that logs the message "An Image has been added" when an object is uploaded to a specific S3 bucket.

## Theory:

AWS Lambda is a serverless compute service that allows you to run code without provisioning or managing servers. You can set up a Lambda function to automatically execute in response to various AWS events, such as the addition of an object to an S3 bucket.

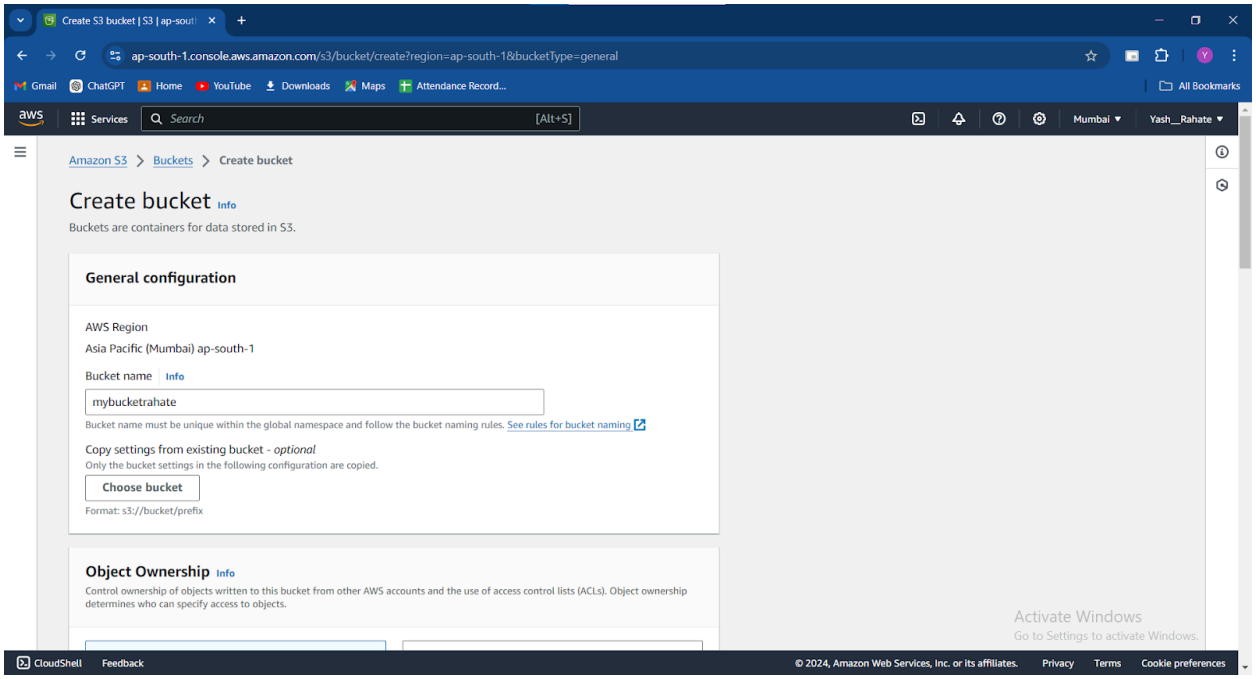
- **S3 Buckets**: Amazon S3 (Simple Storage Service) is an object storage service that provides scalable storage for files. You can set up triggers that notify other AWS services when specific events occur, such as uploading or deleting an object in an S3 bucket.
- **Lambda Function**: This is a small, self-contained piece of code that runs in response to events. In this case, the event will be the addition of an image to an S3 bucket. The function will log a specific message using CloudWatch Logs.
- **Event Source Mapping**: When configuring Lambda, you set S3 as the event source. The event mapping includes specifying which bucket and what type of event (such as the `s3:ObjectCreated:*` event) should trigger the Lambda function.
- **CloudWatch Logs**: AWS CloudWatch is a monitoring service where logs generated by Lambda functions are stored. The Lambda function in this case will log the message "An Image has been added" whenever an object is uploaded to the specified S3 bucket.

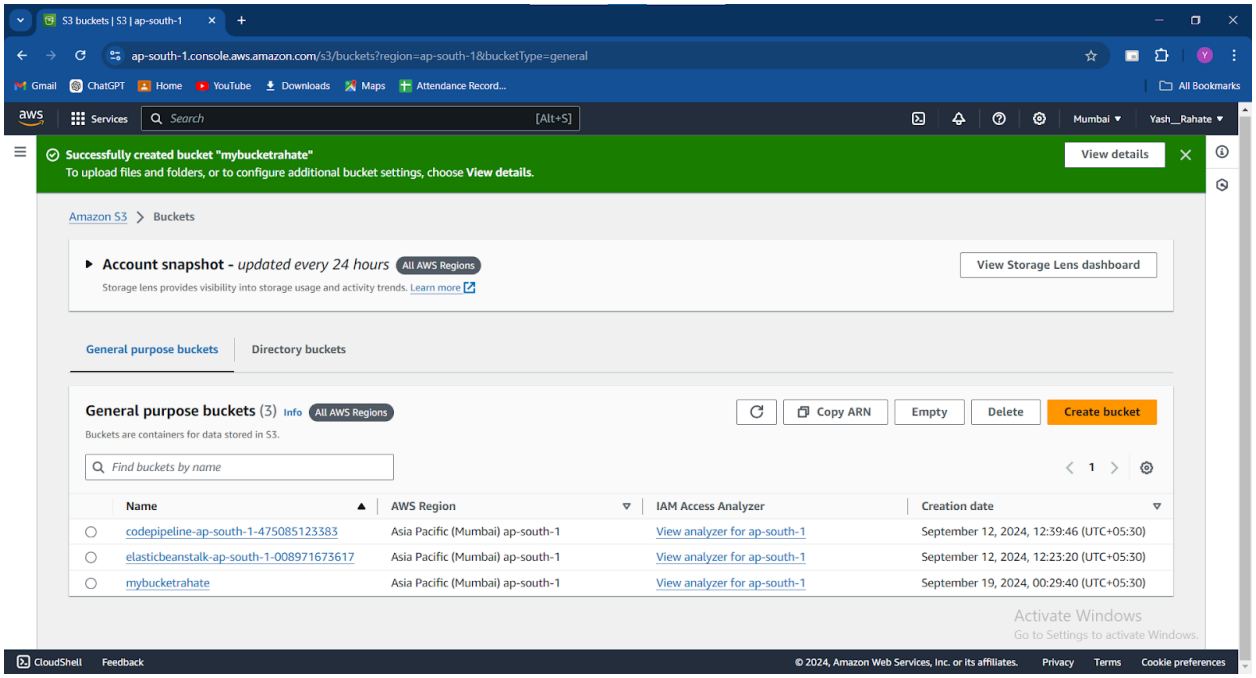
**Key Steps:**

Here are the steps to create a Lambda function that logs “An Image has been added” once an object is added to a specific S3 bucket in AWS Learner Lab:

### 1. Create an S3 Bucket

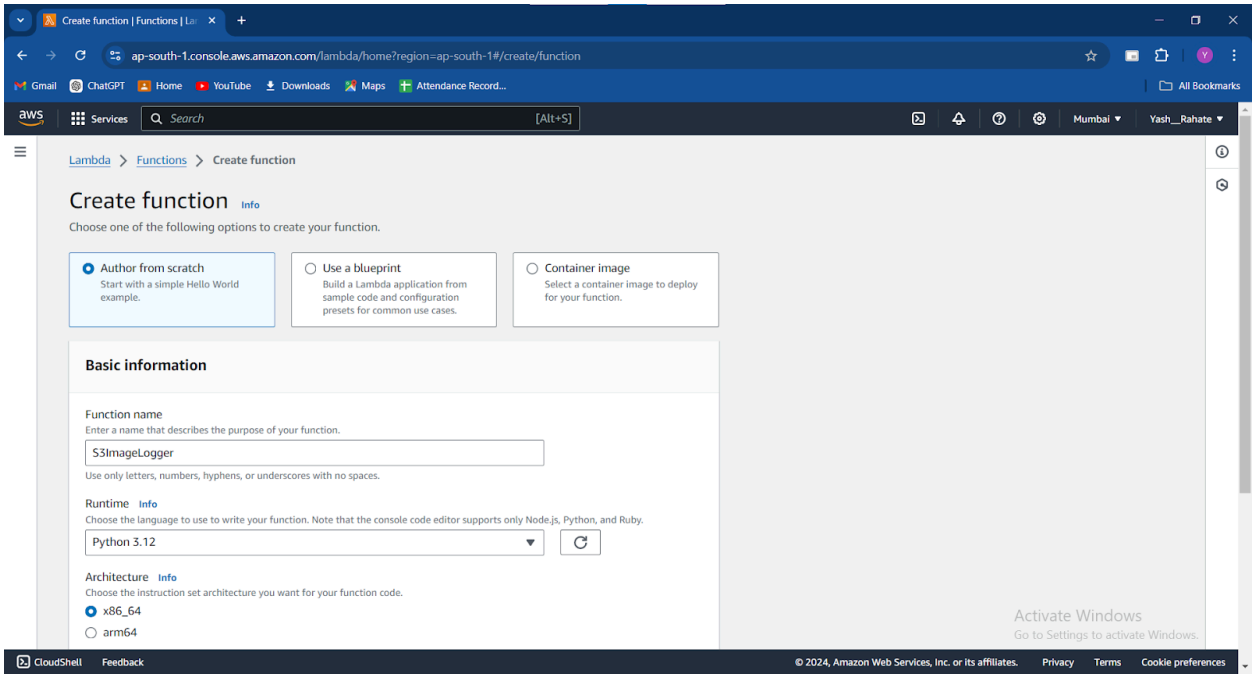
- Go to the AWS Management Console.
- Navigate to the S3 service.
- Click on "Create bucket."
- Enter a unique bucket name and choose a region.
- Configure other settings as needed and click "Create bucket."






## 2. Create a Lambda Function

- Go to the AWS Management Console.
- Navigate to the Lambda service.
- Click on "Create function."
- Choose "Author from scratch."
- Enter a name for your function, e.g., S3ImageLogger.
- Select a runtime (e.g., Python 3.x or Node.js).
- Click "Create function."



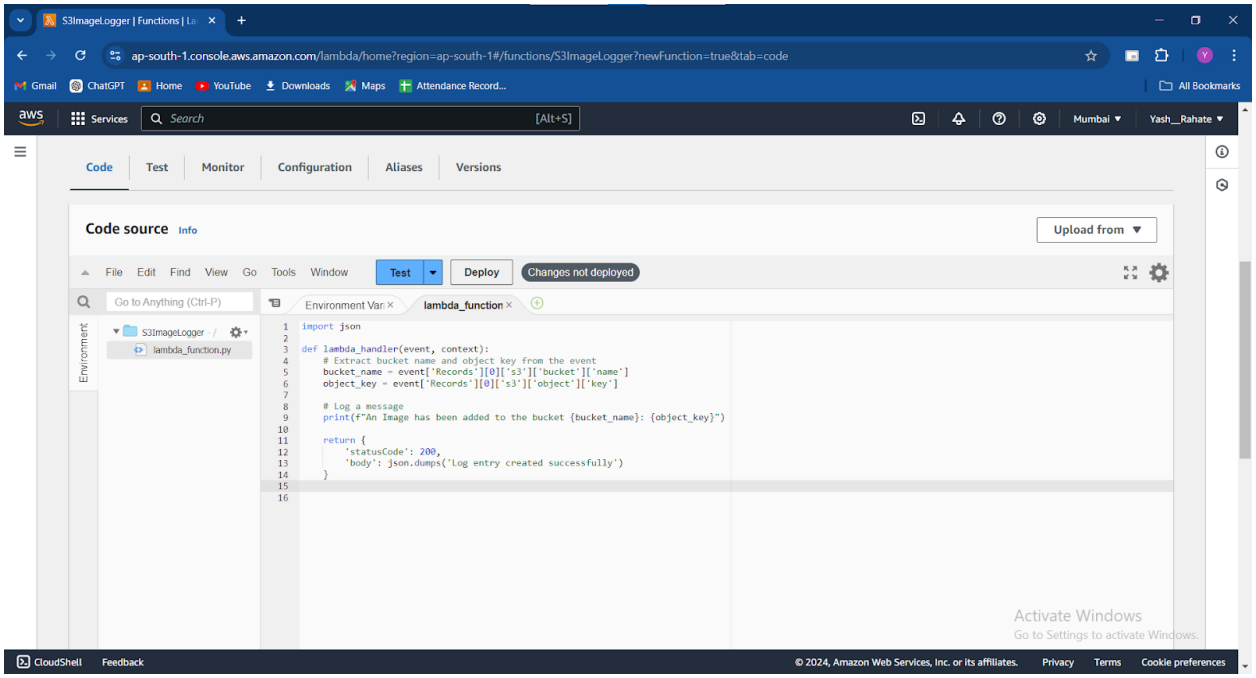

### 3. Write the Lambda Function Code

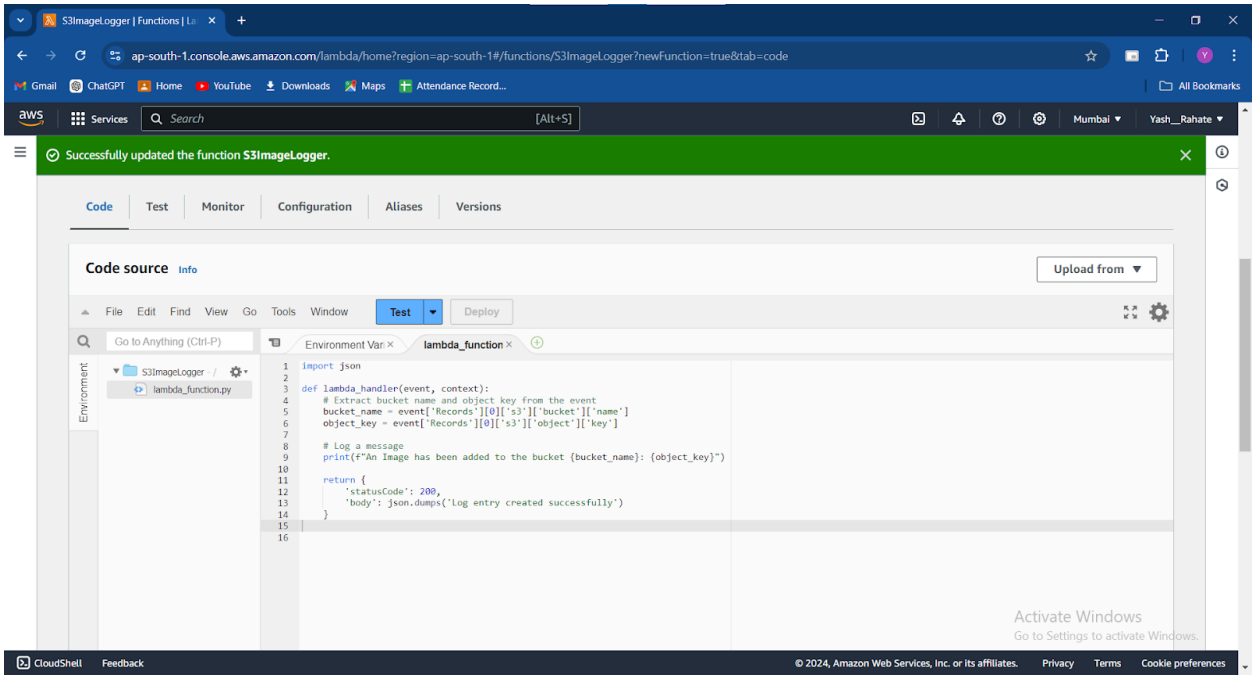
- In the Lambda function console, scroll down to the code editor.

Replace the default code with the following code snippet (assuming you're using Python):
python
Copy code

```
import json

def lambda_handler(event, context):
    # Extract bucket name and object key from the event
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    object_key = event['Records'][0]['s3']['object']['key']

    # Log a message
    print(f"An Image has been added to the bucket {bucket_name}: {object_key}")

    return {
        'statusCode': 200,
        'body': json.dumps('Log entry created successfully')
    }
```

- Click "Deploy" to save your changes.

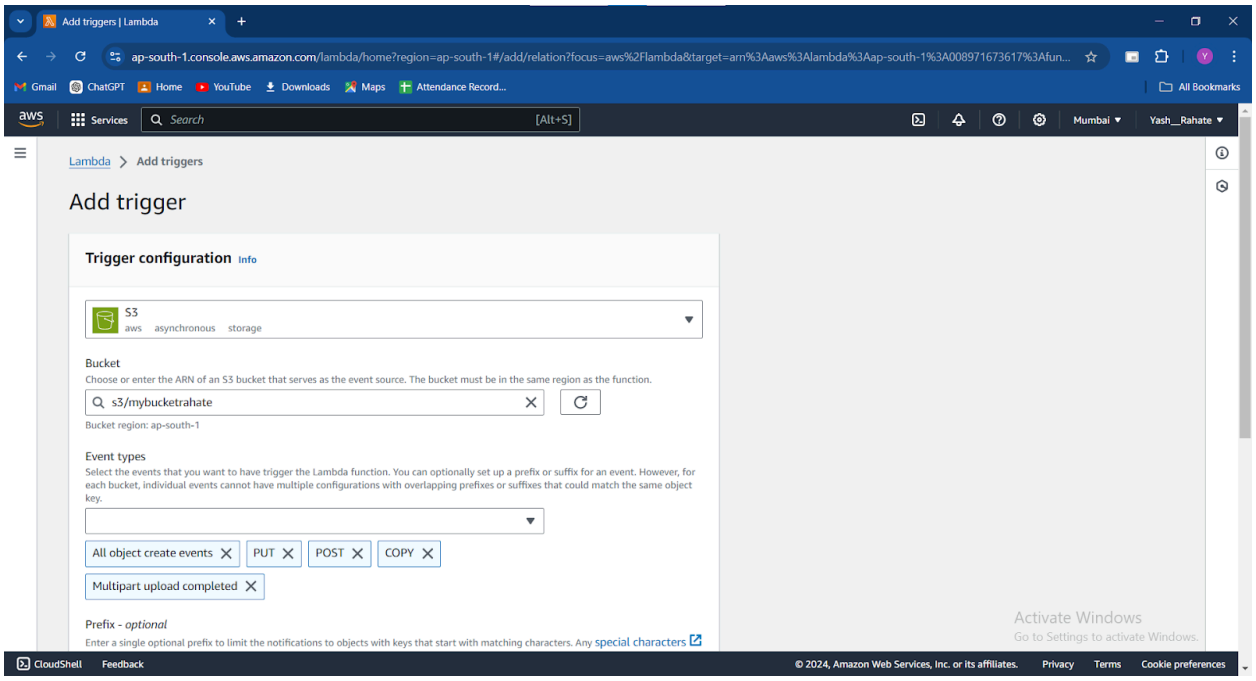



## 4. Set Up S3 Trigger for the Lambda Function

- Scroll down to the "Function overview" section in the Lambda console.
- Click on "Add trigger."
- Select "S3" from the list of triggers.
- Choose the S3 bucket you created earlier.
- In the "Event type" dropdown, select "All object create events."

- Optionally, specify a prefix or suffix to filter the events (e.g., for images only, you can use suffix .jpg, .png).
- Click "Add."

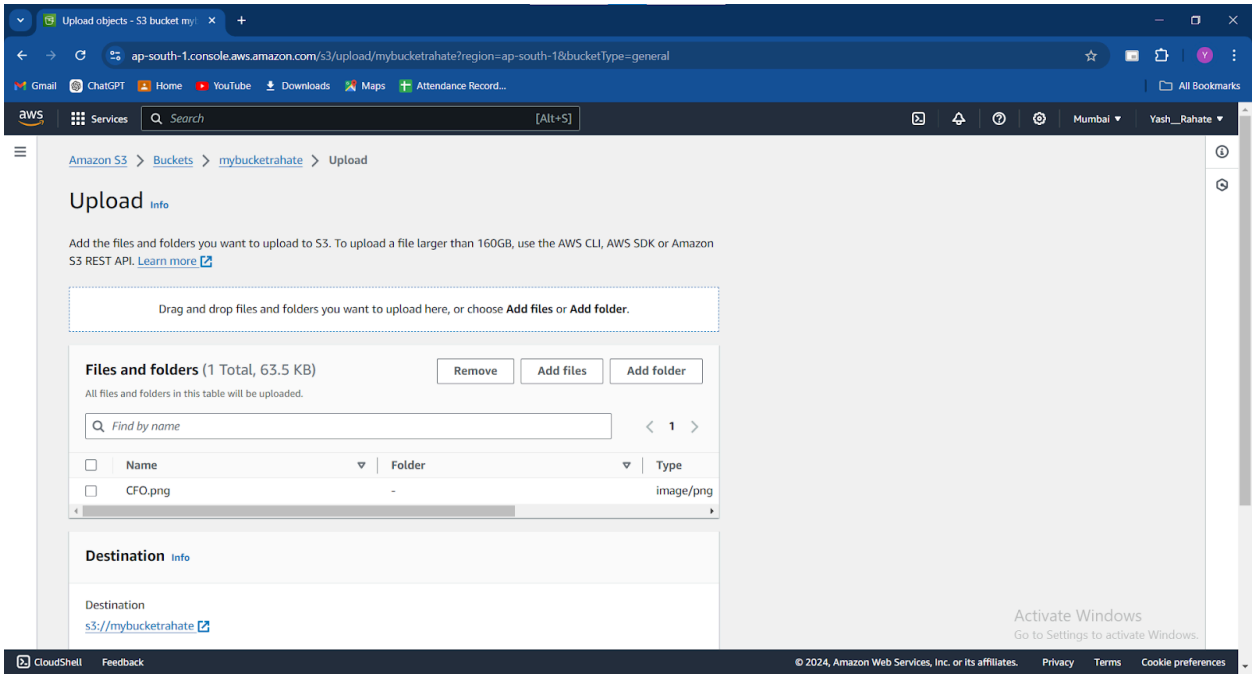


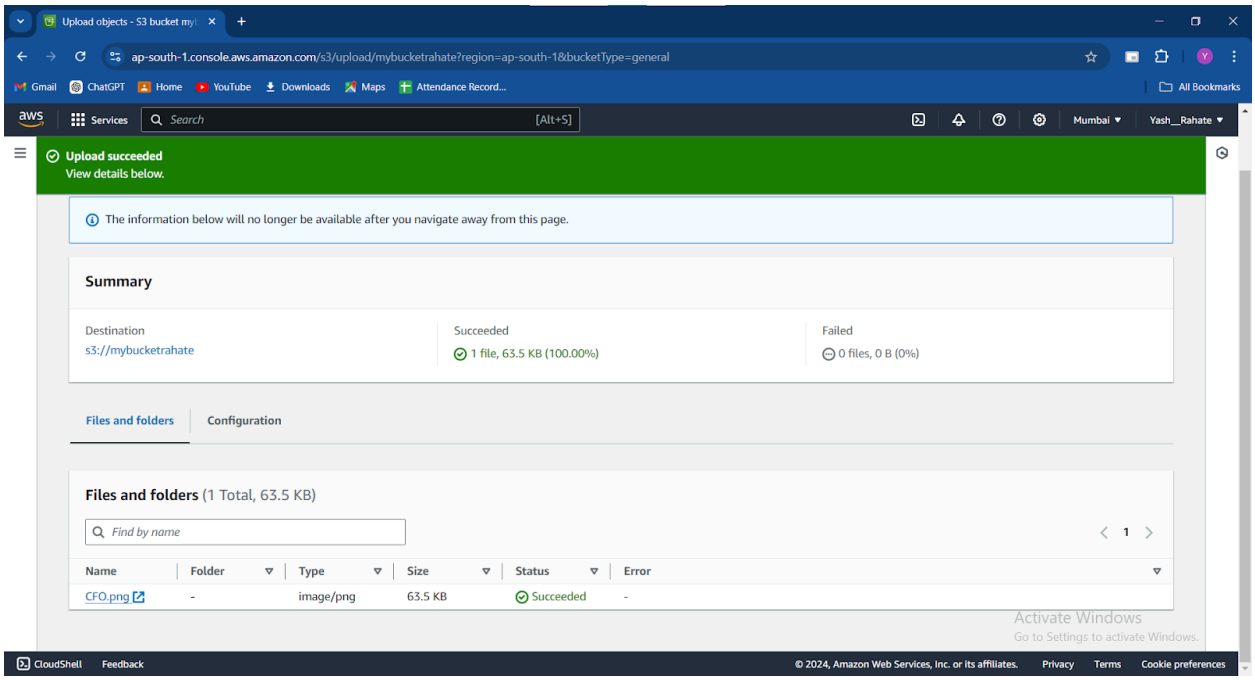

## 5. Grant Permissions to Lambda

- Navigate to the "Permissions" tab of your Lambda function.
- Ensure the Lambda function's execution role has the necessary permissions to access the S3 bucket.
- If needed, attach the AmazonS3ReadOnlyAccess policy or create a custom policy with the necessary permissions.

## 6. Test the Setup

- Upload an image file to your S3 bucket.
- Go to the "Monitoring" tab in your Lambda function to check the logs.
- Alternatively, use CloudWatch Logs to view the output and confirm that the message "An Image has been added" has been logged.

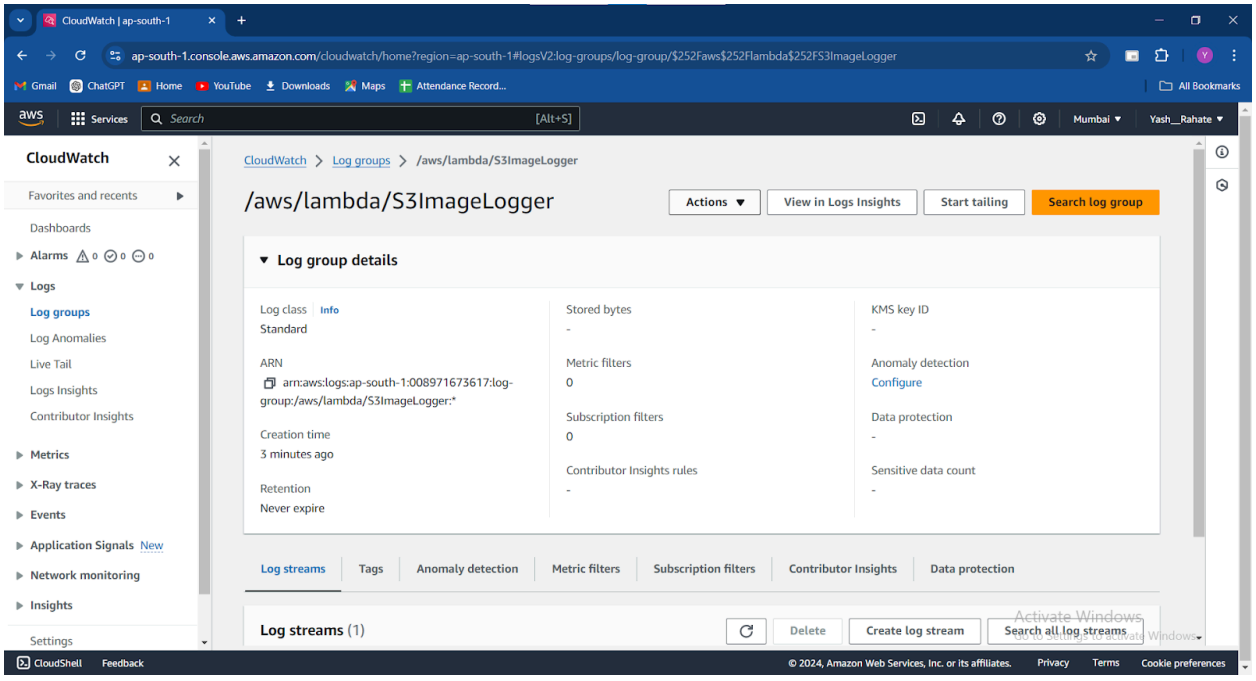


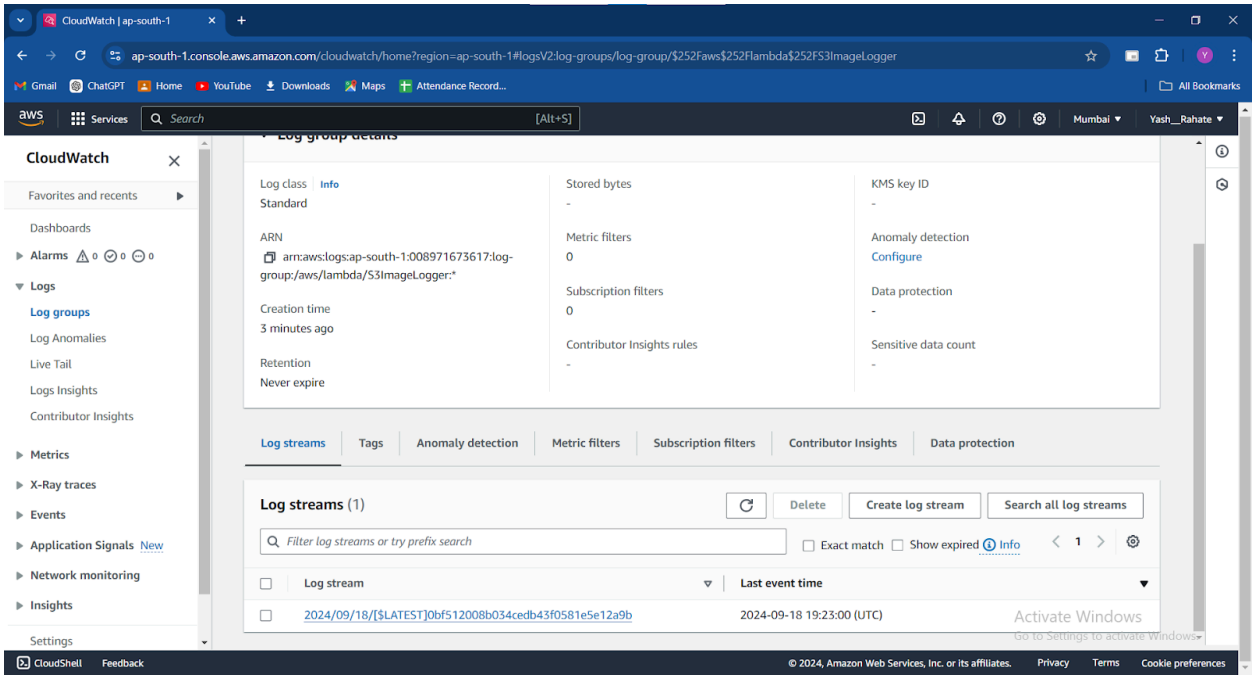



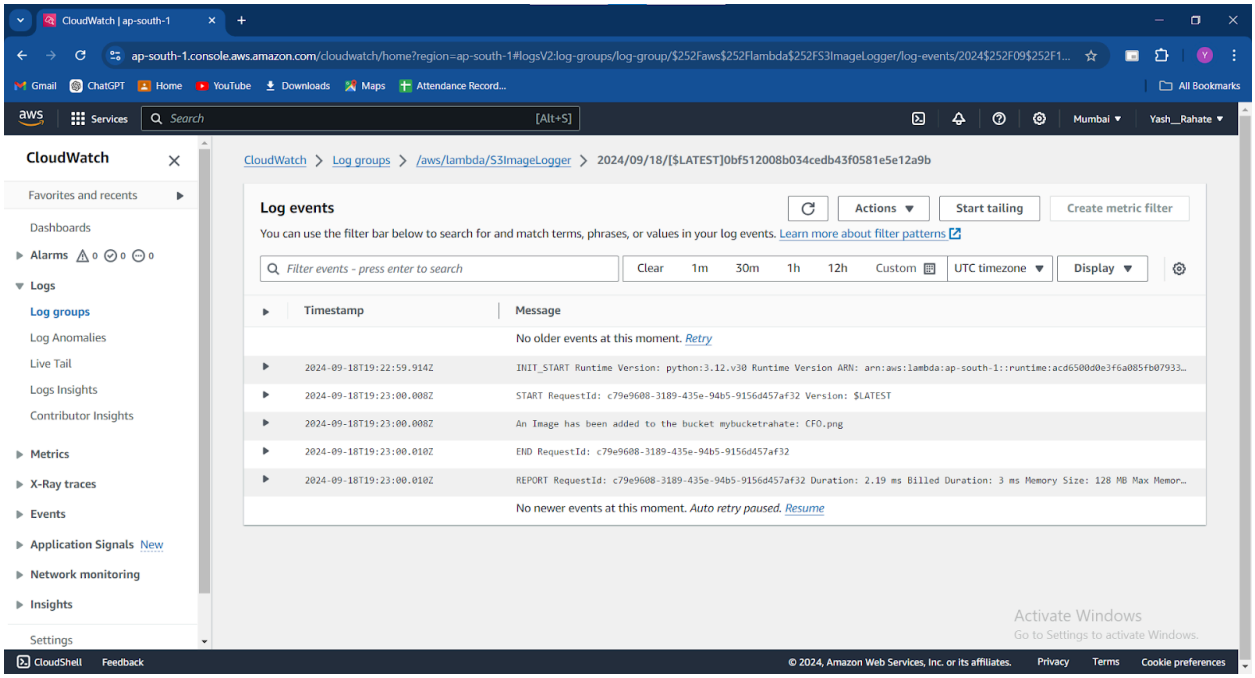

This setup should ensure that each time an image is uploaded to the specified S3 bucket, the Lambda function will log the appropriate message.

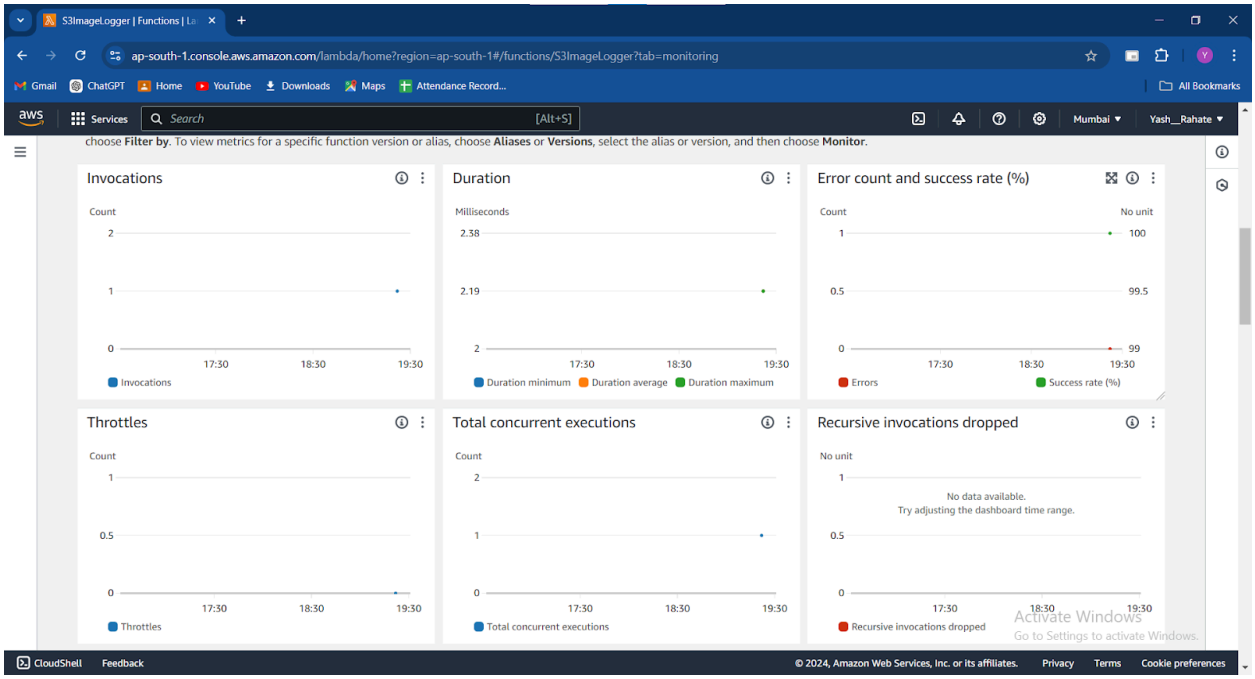
Image uploaded in S3 bucket

## 7. Using CloudWatch Logs to view the output

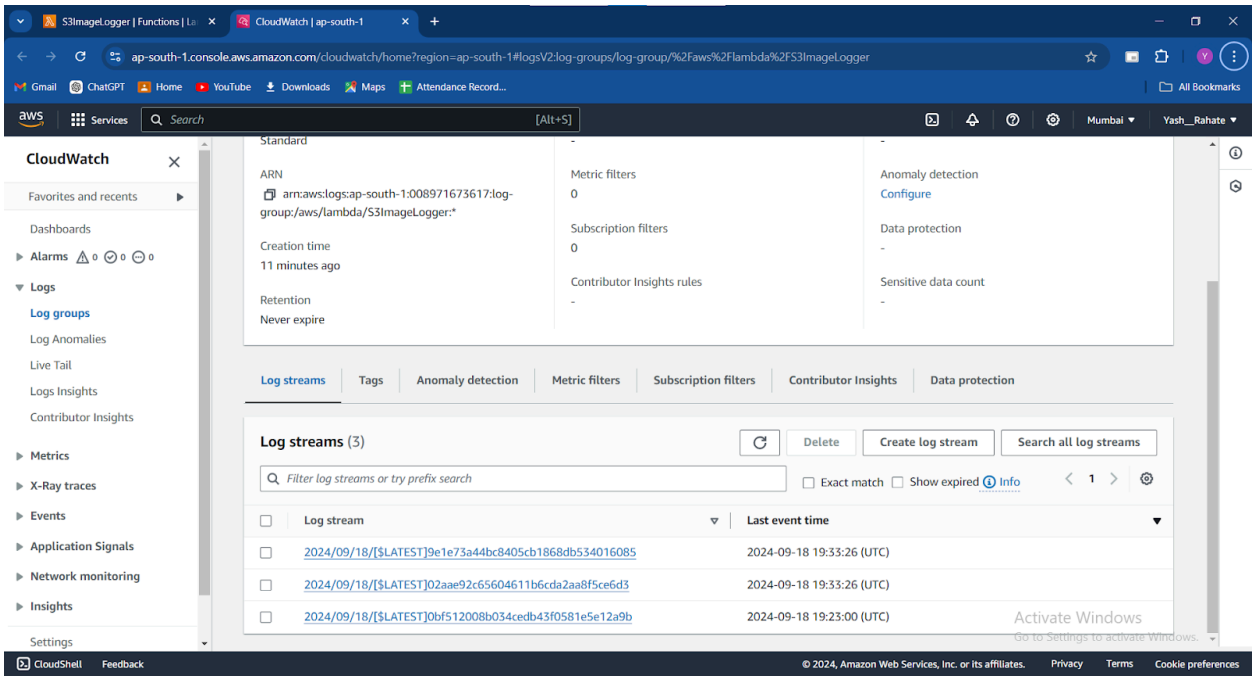



CloudWatch | ap-south-1
ap-south-1.console.aws.amazon.com/cloudwatch/home?region=ap-south-1#logsV2:log-groups/log-group/$252Faws$252Flambda$252FS3ImageLogger/log-events/2024$252F09$252F1...
Gmail ChatGPT Home YouTube Downloads Maps Attendance Record... All Bookmarks
aws Services Search [Alt+S] Mumbai Yash__Rahate
CloudWatch
Favorites and recents
Dashboards
Alarms 0 0 0
Logs
Log groups
Log Anomalies
Live Tail
Logs Insights
Contributor Insights
Metrics
X-Ray traces
Events
Application Signals New
Network monitoring
Insights
Settings
CloudWatch > Log groups > /aws/lambda/S3ImageLogger > 2024/09/18/[$LATEST]0bf512008b034cedb43f0581e5e12a9b
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
2024-09-18T19:22:59.914Z INIT_START Runtime Version: python:3.12.v30 Runtime Version ARN: arn:aws:lambda:ap-south-1::runtime:acd6500d0e3f6a085fb07933...
2024-09-18T19:23:00.008Z START RequestId: c79e9608-3189-435e-94b5-9156d457af32 Version: $LATEST
2024-09-18T19:23:00.008Z An Image has been added to the bucket mybucketrahate: CFO.png
2024-09-18T19:23:00.010Z END RequestId: c79e9608-3189-435e-94b5-9156d457af32
2024-09-18T19:23:00.010Z REPORT RequestId: c79e9608-3189-435e-94b5-9156d457af32 Duration: 2.19 ms Billed Duration: 3 ms Memory Size: 128 MB Max Memor...
No newer events at this moment. Auto retry paused. Resume
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


S3ImageLogger | Functions | La
ap-south-1.console.aws.amazon.com/lambda/home?region=ap-south-1#/functions/S3ImageLogger?tab=monitoring
Gmail ChatGPT Home YouTube Downloads Maps Attendance Record... All Bookmarks
aws Services Search [Alt+S] Mumbai Yash__Rahate
choose Filter by. To view metrics for a specific function version or alias, choose Aliases or Versions, select the alias or version, and then choose Monitor.
Invocations
Count
Invocations
Duration
Milliseconds
Duration minimum Duration average Duration maximum
Error count and success rate (%)
Count
No unit
Errors Success rate (%)
Throttles
Count
Throttles
Total concurrent executions
Count
Total concurrent executions
Recursive invocations dropped
No unit
No data available.
Try adjusting the dashboard time range.
Recursive invocations dropped
17:30 18:30 19:30
Activate Windows
Go to Settings to activate Windows.
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

After uploading 2 more images




## Conclusion:

This Lambda function demonstrates a simple use case of serverless computing. By connecting S3 and Lambda through event triggers, AWS provides a seamless way to automate tasks, such as logging when an image is uploaded to a bucket. This type of setup can be extended further for more complex workflows, like processing the image or sending notifications.