

● Beginner – Hands-On Lambda Labs

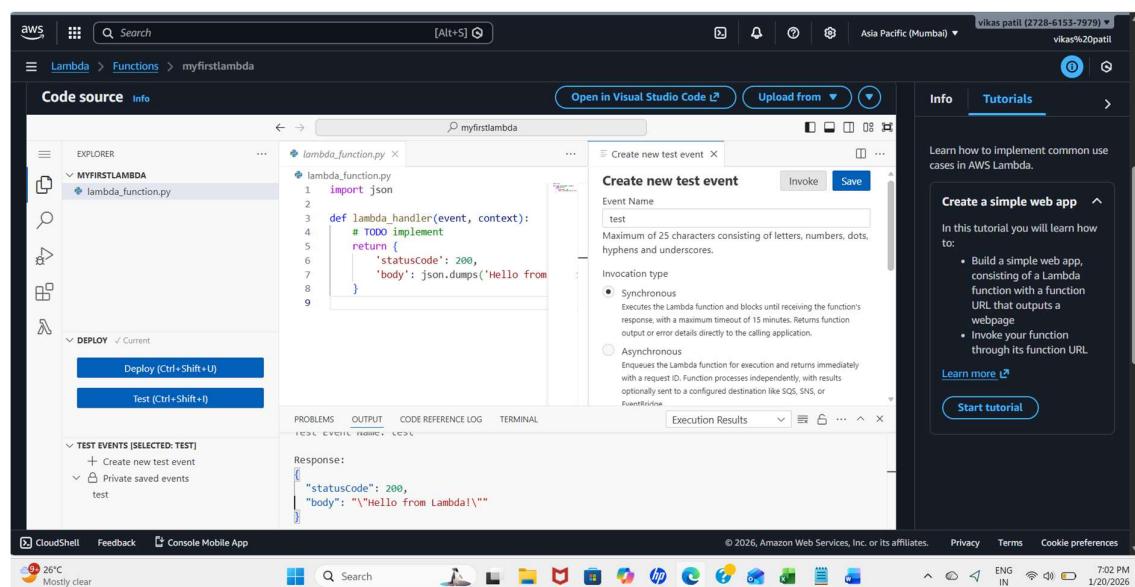
1 Hello World Lambda

Task:

- Create a Lambda function using **Python**
- Return "Hello from AWS Lambda" when invoked

Skills practiced:

- Lambda creation
- Basic handler function
- Test event



2. Lambda Triggered by S3 Upload

Task:

- Create an S3 bucket
- Trigger Lambda when a file is uploaded
- Print the file name and bucket name in logs

Skills practiced:

- S3 event triggers
- IAM permissions
- CloudWatch Logs

✓ PRACTICAL LAB: Lambda Triggered by S3 Upload

🎯 Goal

When a file is uploaded to **S3**, a **Lambda function** should run and **log the bucket name and file name**.

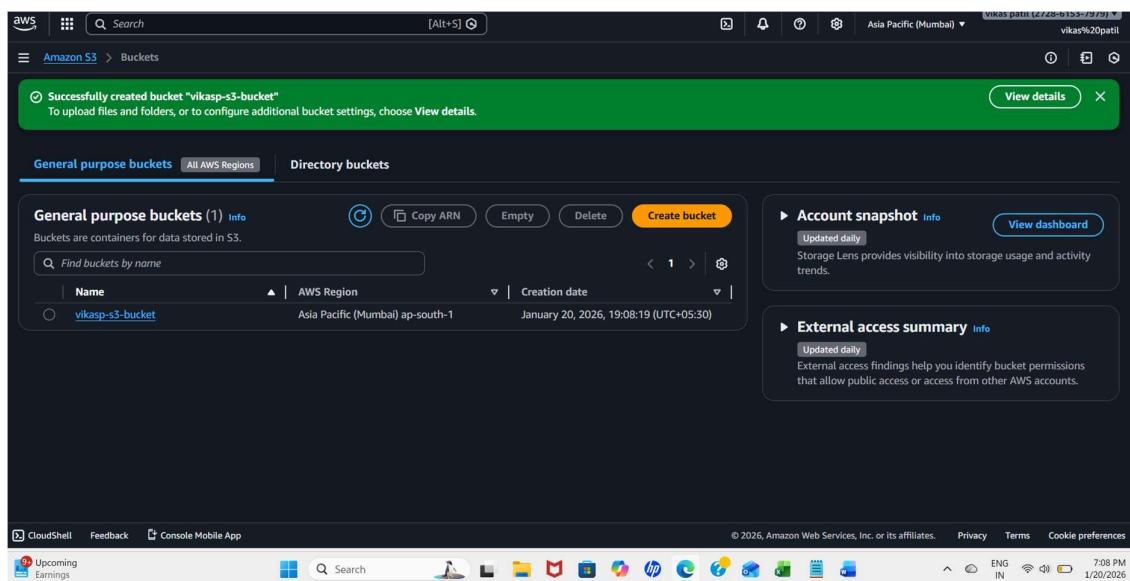
📦 Architecture

User uploads file → S3 Bucket → Lambda Function → CloudWatch Logs

✍ STEP-BY-STEP PROCESS

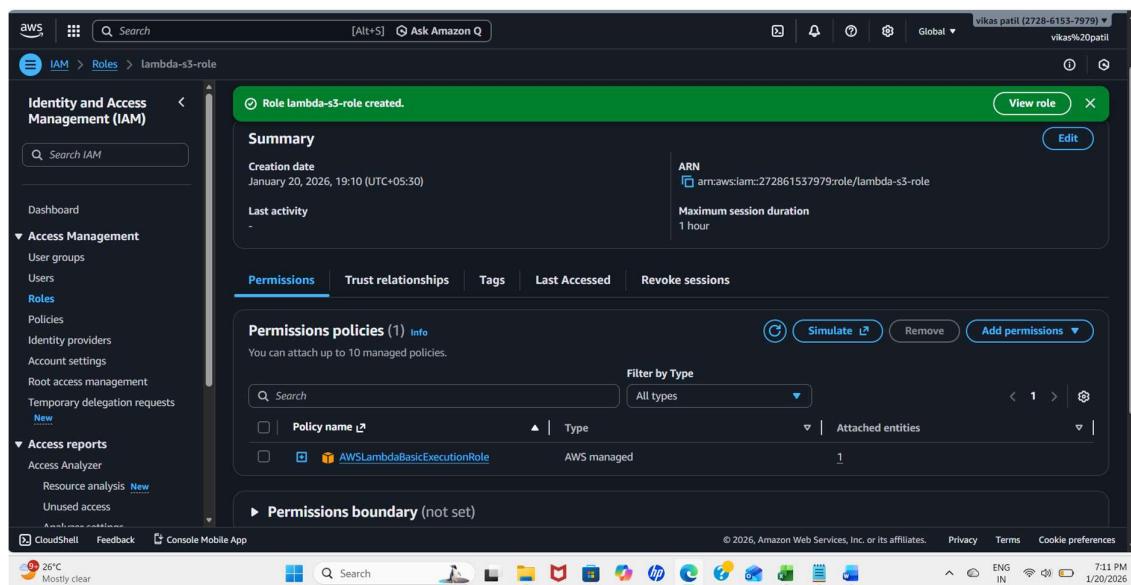
◆ STEP 1: Create S3 Bucket

1. Go to **AWS Console** → **S3**
2. Click **Create bucket**
3. Bucket name:
vikasp-s3-bucket
4. Region: Same as Lambda
5. Leave defaults → **Create bucket**



◆ STEP 2: Create IAM Role for Lambda

1. Go to **IAM** → **Roles** → **Create role**
2. Select **AWS service**
3. Choose **Lambda**
4. Attach policies:
 - **AWSLambdaBasicExecutionRole**
5. Role name:
`lambda-s3-role`
6. Click **Create role**



The screenshot shows the AWS IAM Roles page. The left sidebar is collapsed. The main area displays a green success message: "Role lambda-s3-role created." Below this, the "Summary" section shows the creation date (January 20, 2026, 19:10 (UTC+05:30)), ARN (arn:aws:iam::272861537979:role/lambda-s3-role), and maximum session duration (1 hour). The "Permissions" tab is selected, showing one attached policy: "AWSLambdaBasicExecutionRole" (AWS managed). The "Permissions policies" table has one row. The bottom of the page includes standard AWS navigation links and a footer with copyright information and system status.

◆ STEP 3: Create Lambda Function

1. Go to **AWS Lambda** → **Create function**
2. Choose **Author from scratch**
3. Function name:
`s3-trigger-lambda`
4. Runtime:
`Python 3.10`
5. Execution role:
 - Choose **Use existing role**
 - Select `lambda-s3-role`
6. Click **Create function**

Created s3-trigger-lambda function

◆ STEP 4: Add Lambda Code

Replace default code with this 

```
def lambda_handler(event, context):

    for record in event['Records']:

        bucket = record['s3']['bucket']['name']

        file_name = record['s3']['object']['key']

        print(f"File uploaded: {file_name}")

        print(f"Bucket name: {bucket}")



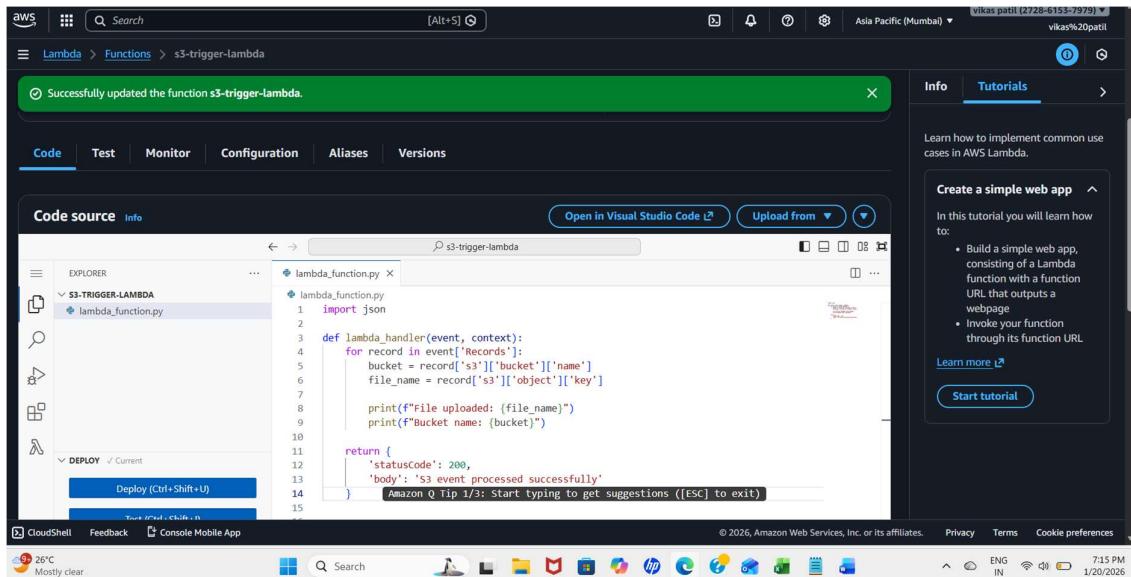
    return {

        'statusCode': 200,

        'body': 'S3 event processed successfully'

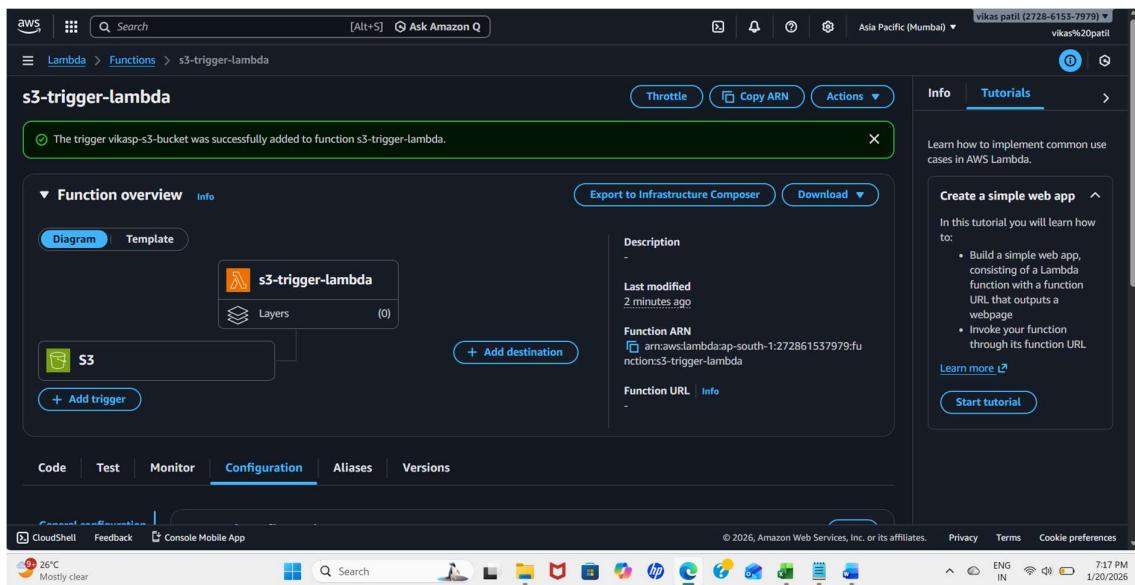
    }
```

 Click Deploy



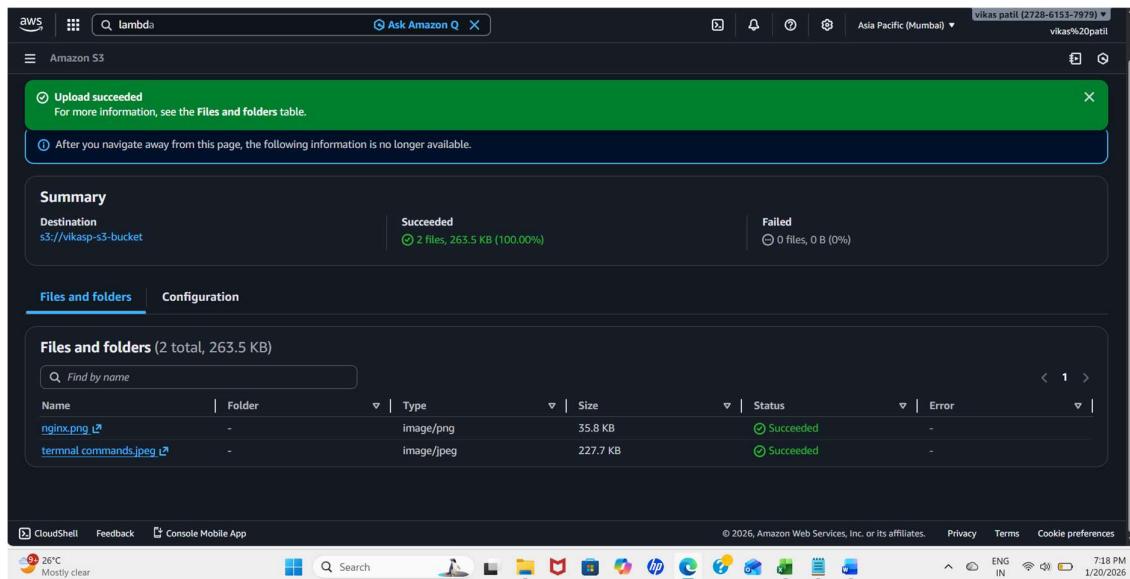
◆ STEP 5: Configure S3 Trigger

1. Go to **Lambda** → **Configuration** → **Triggers**
2. Click **Add trigger**
3. Select **S3**
4. Choose your bucket
5. Event type:
 PUT (Object created)
6. Check **Acknowledgement**
7. Click **Add**



◆ STEP 6: Upload File to S3

1. Go to **S3** → **Your bucket**
2. Click **Upload**
3. Upload any file (e.g. test.txt)
4. Click **Upload**



aws | λ lambda | Ask Amazon Q | Amazon S3 | Asia Pacific (Mumbai) | vikas patil (2728-6153-7979) | vikas%20patil

Upload succeeded
For more information, see the [Files and folders](#) table.

After you navigate away from this page, the following information is no longer available.

Summary		Succeeded		Failed	
Destination	2 files, 263.5 KB (100.00%)	0 files, 0 B (0%)			
s3://vikasp-s3-bucket					

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

Files and folders (2 total, 263.5 KB)

Name	Folder	Type	Size	Status	Error
nginx.png	-	image/png	35.8 KB	Succeeded	-
terminal commands.jpeg	-	image/jpeg	227.7 KB	Succeeded	-

CloudShell Feedback Console Mobile App | © 2026, Amazon Web Services, Inc. or its affiliates. | Privacy | Terms | Cookie preferences | 26°C Mostly clear | ENG IN | 7:18 PM | 1/20/2026

◆ STEP 7: Verify Output (MOST IMPORTANT)

1. Go to **Lambda** → **Monitor** → **View CloudWatch logs**
2. Open latest log stream
3. You should see:

File uploaded: nginx.png

File uploaded: terminal commands.jpeg

Bucket name: vikasp-s3-bucket

 **SUCCESS!**

CloudWatch Log management /aws/lambda/s3-trigger-lambda 2026/01/20/[LATEST]af9c3ad619746c2be8e07709875f0c

Log events

Timestamp Message

No older events at this moment. *Retry*

2026-01-20T13:48:15.871Z INIT_START Runtime Version: python:3.10.v109 Runtime Version ARN: arn:aws:lambda:ap-south-1:runtime:c3d97831572c48429cac556b7614e40...

2026-01-20T13:48:15.953Z START RequestId: 4b60492f-4aa4-4052-b1f1-ea436d5d738f Version: \$LATEST

2026-01-20T13:48:15.953Z File uploaded: nginx.png

2026-01-20T13:48:15.953Z Bucket name: vikasp-s3-bucket

2026-01-20T13:48:15.954Z END RequestId: 4b60492f-4aa4-4052-b1f1-ea436d5d738f

2026-01-20T13:48:15.954Z REPORT RequestId: 4b60492f-4aa4-4052-b1f1-ea436d5d738f Duration: 1.39 ms Billed Duration: 80 ms Memory Size: 128 MB Max Memory Used:...

No newer events at this moment. *Auto retry paused. Resume*

CloudWatch Log management /aws/lambda/s3-trigger-lambda 2026/01/20/[LATEST]69d4cb1e3ac7406b8e4d0a0c587d1f9

Log events

Timestamp Message

No older events at this moment. *Retry*

2026-01-20T13:48:15.853Z INIT_START Runtime Version: python:3.10.v109 Runtime Version ARN: arn:aws:lambda:ap-south-1:runtime:c3d97831572c48429cac556b7614e40...

2026-01-20T13:48:15.940Z START RequestId: ce5db7b7-d011-498c-98f5-4aeee3f39185 Version: \$LATEST

2026-01-20T13:48:15.940Z File uploaded: terminal+commands.jpeg

2026-01-20T13:48:15.940Z Bucket name: vikasp-s3-bucket

2026-01-20T13:48:15.942Z END RequestId: ce5db7b7-d011-498c-98f5-4aeee3f39185

2026-01-20T13:48:15.942Z REPORT RequestId: ce5db7b7-d011-498c-98f5-4aeee3f39185 Duration: 1.69 ms Billed Duration: 85 ms Memory Size: 128 MB Max Memory Used:...

No newer events at this moment. *Auto retry paused. Resume*

🧠 WHAT I PRACTICED

- ✓ Lambda creation
- ✓ IAM roles
- ✓ S3 event triggers
- ✓ CloudWatch logs
- ✓ Serverless workflow