

Lesson 08 Demo 01

Creating Lambda Function and CloudWatch

Objective: To create an AWS Lambda function, deploy code to it, test the function, and view its CloudWatch logs

Tools: AWS Lambda, AWS S3, and AWS CloudWatch

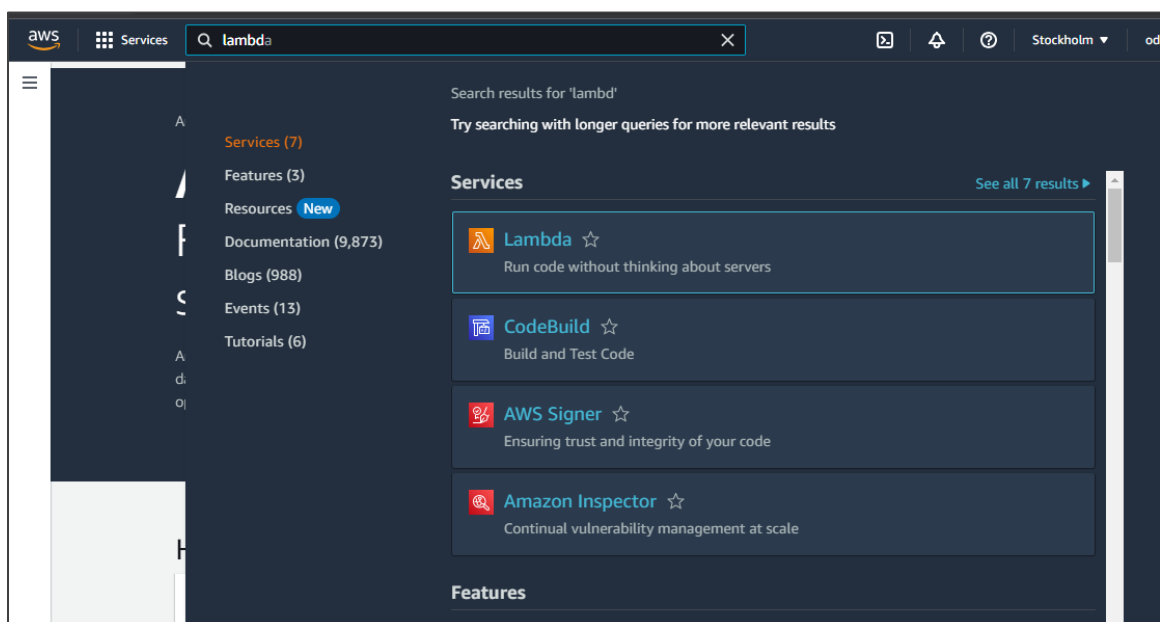
Prerequisites: An AWS account with access to its lab

Steps to be followed:

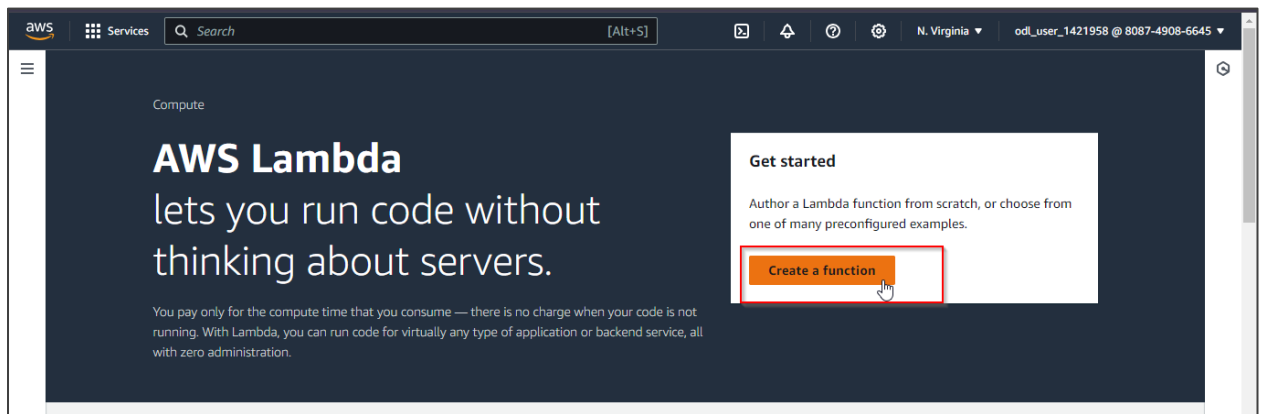
1. Create a Lambda service
2. Deploy code to Lambda function
3. Test the Lambda function
4. View CloudWatch logs

Step 1: Create a Lambda service

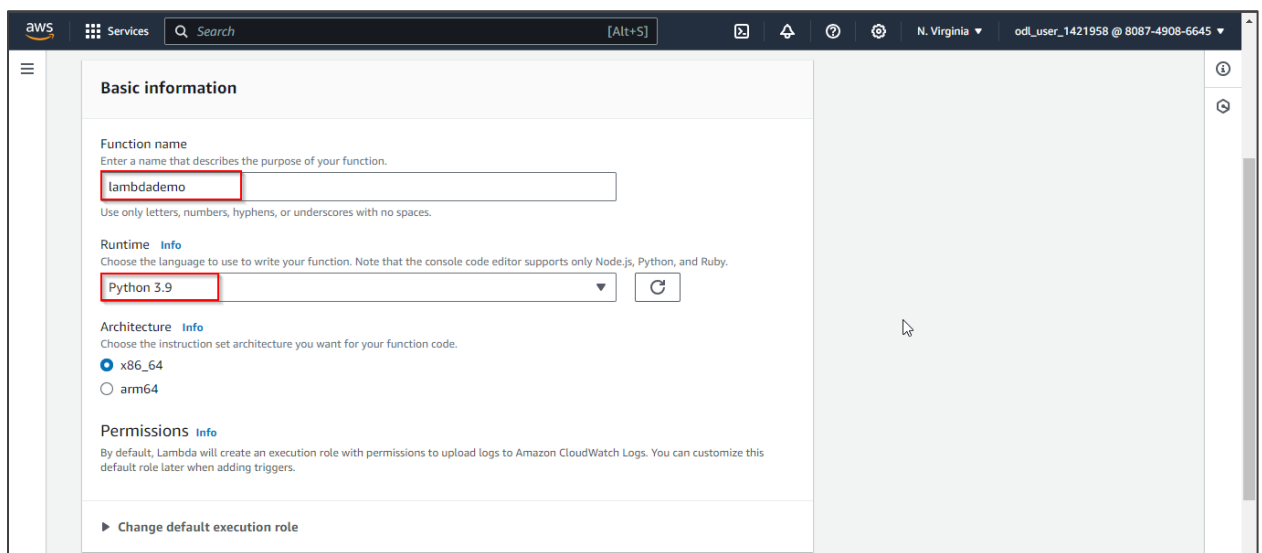
1.1 On the AWS Management Console, search for and click **Lambda**



1.2 Click on the **Create a function**



1.3 Enter the **Function name** as **lambd_demo** and choose the **Runtime** as **Python 3.9**



1.4 Click on **Change default execution role**, select **Create a new role from AWS policy templates**, enter the Role name as **Eventdemo**, and choose the **Amazon S3 object** as the Policy templates.

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.

Role name
Enter a name for your new role.
Eventdemo

Policy templates - optional Info
Choose one or more policy templates.
Amazon S3 object read-only permissions

1.5 Click on the **Create function**

Create a new role from AWS policy templates

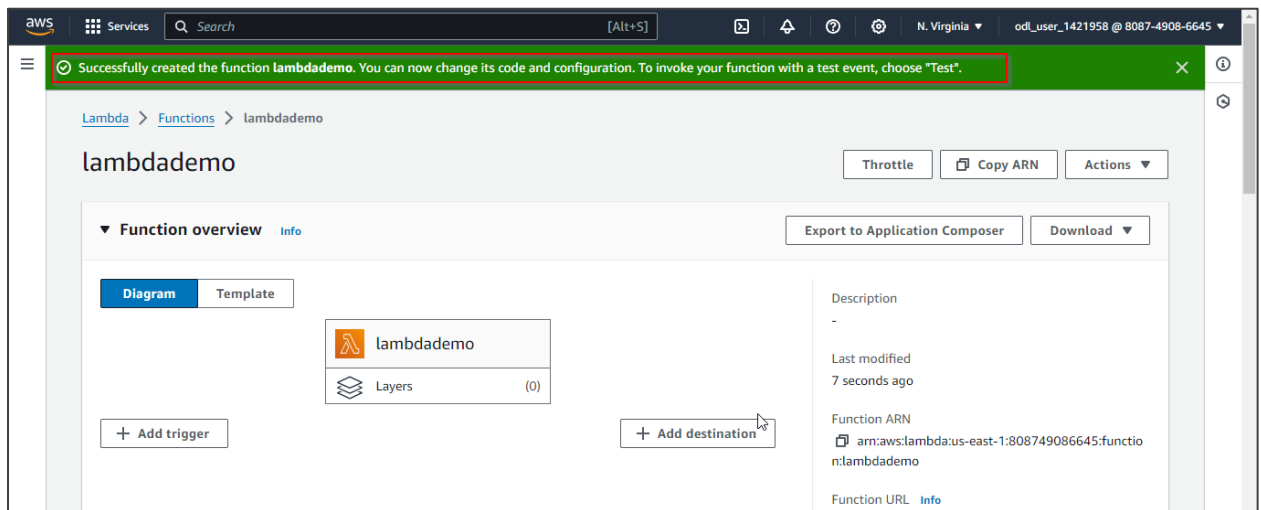
Role creation might take a few minutes. Please do not delete the role or edit the trust or permissions policies in this role.

Role name
Enter a name for your new role.
Eventdemo

Policy templates - optional Info
Choose one or more policy templates.
Amazon S3 object read-only permissions

Advanced settings

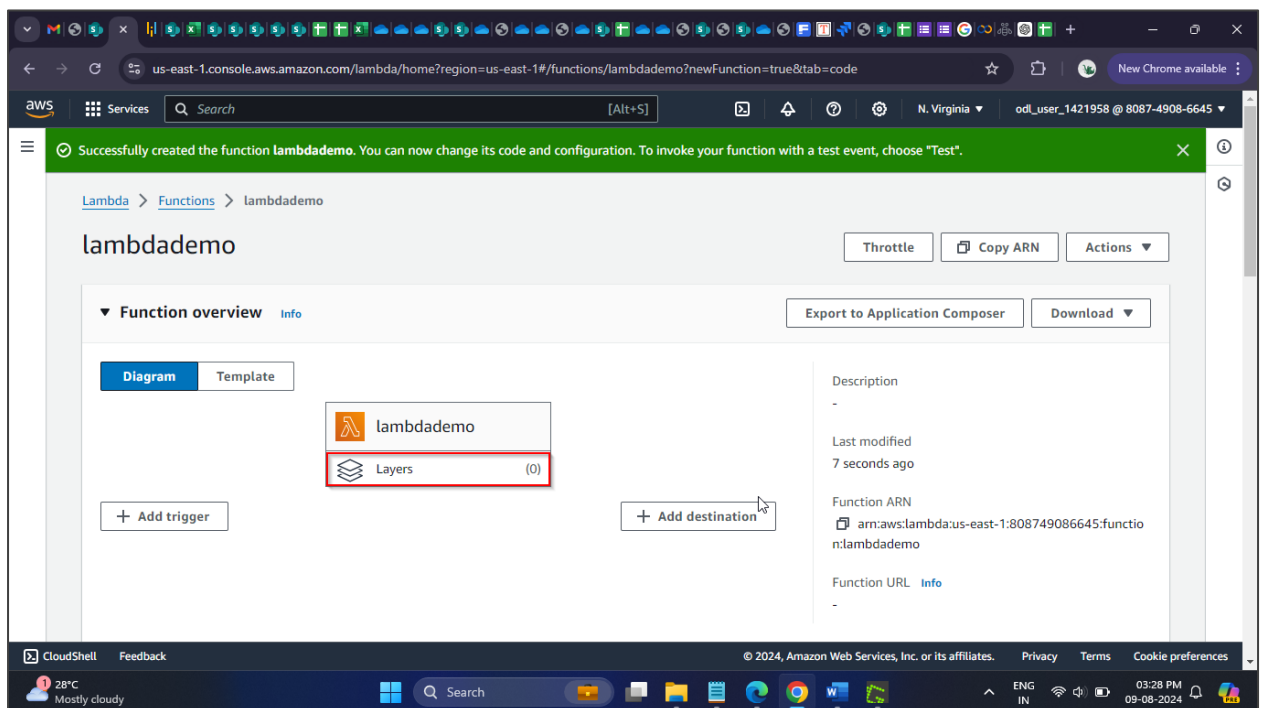
Cancel **Create function**



The Lambda function has been successfully created.

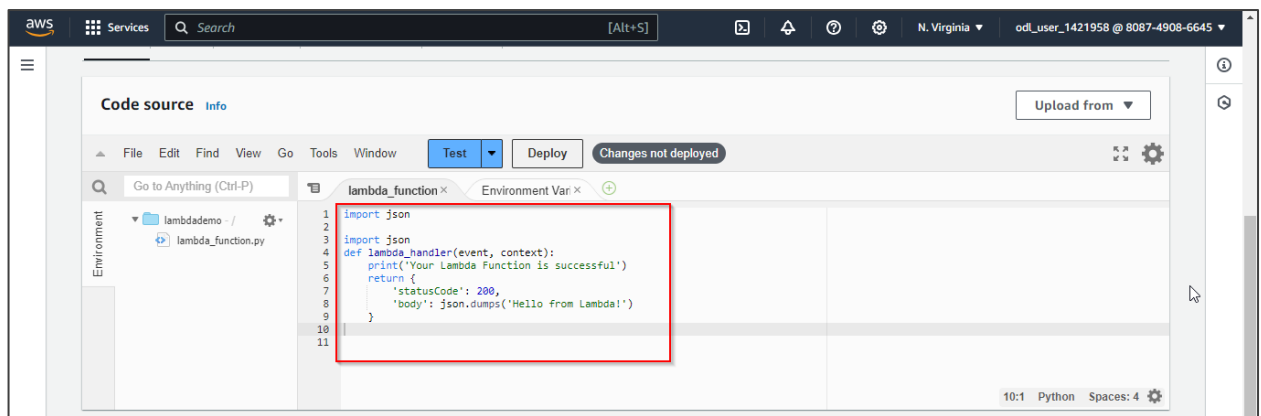
Step 2: Deploy code to Lambda Function

2.1 Click on the Layers

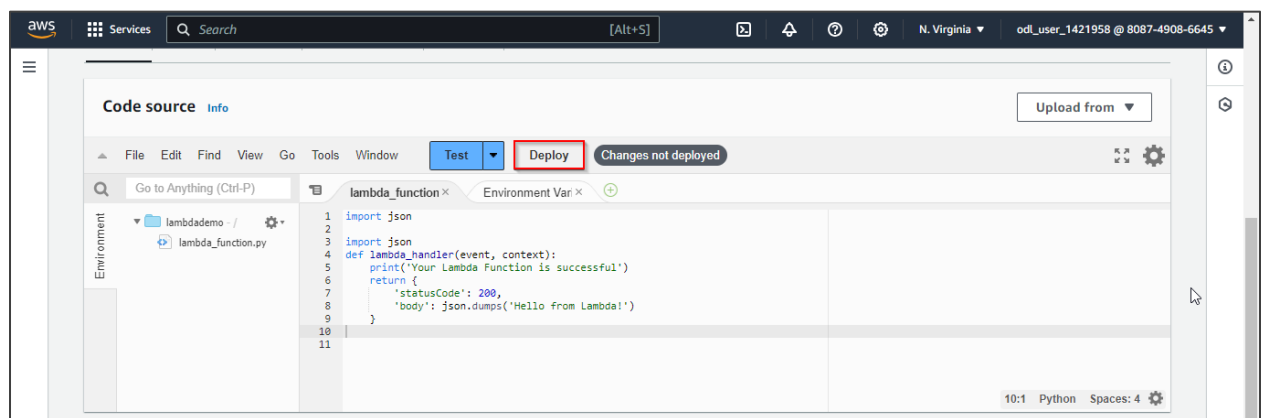


2.2 Enter the following code into the **lambda_function** editor:

```
import json
def lambda_handler(event, context):
    print('Your Lambda Function is successful')
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')
    }
```



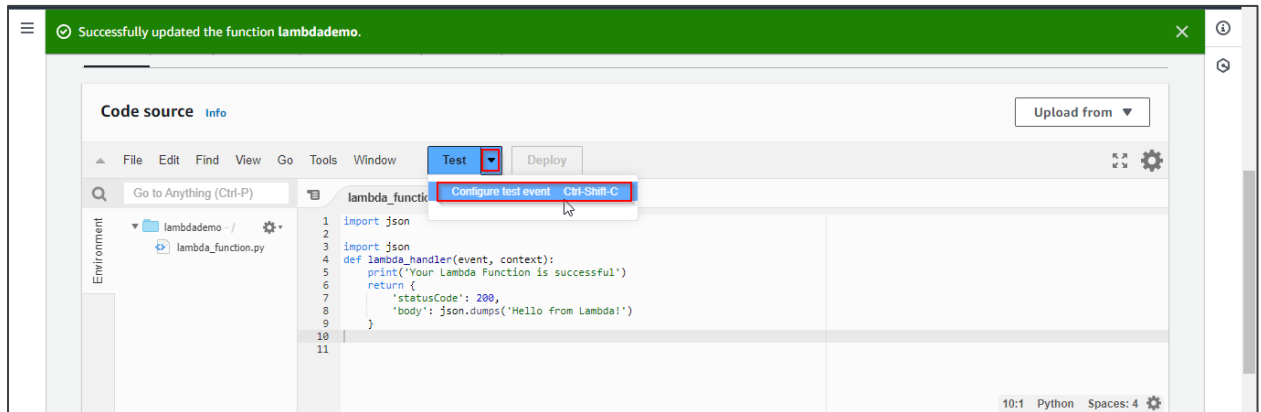
2.3 Click on the **Deploy** option



The Lambda function has been successfully updated.

Step 3: Test the Lambda Function

3.1 Click on the **Test** option and select **Configure test event**



3.2 Enter the **Event name** as **lamdademo** and click on the **Save** button

Configure test event

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action

☒ Create new event
 ☐ Edit saved event

Event name

lamdademo

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](#)

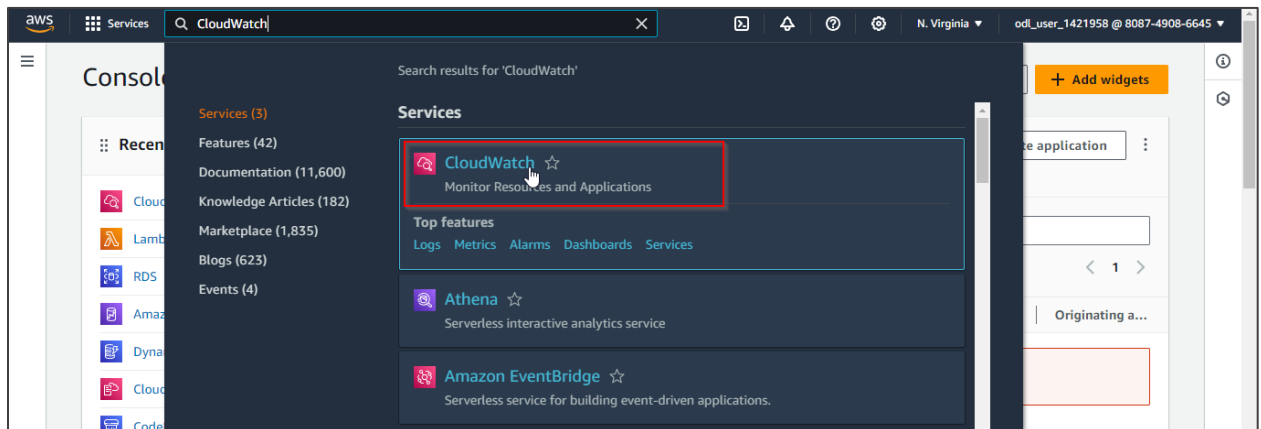
☐ Shareable
 This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

Template - optional

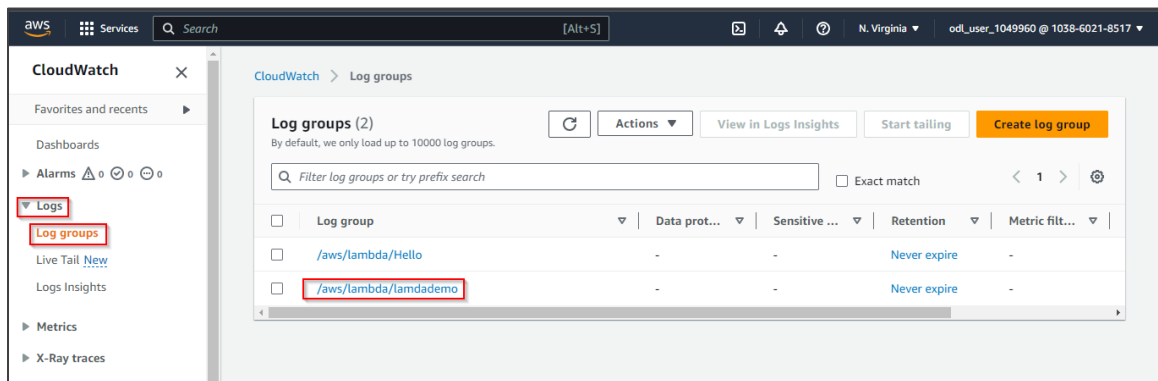
hello-world

Step 4: View CloudWatch logs

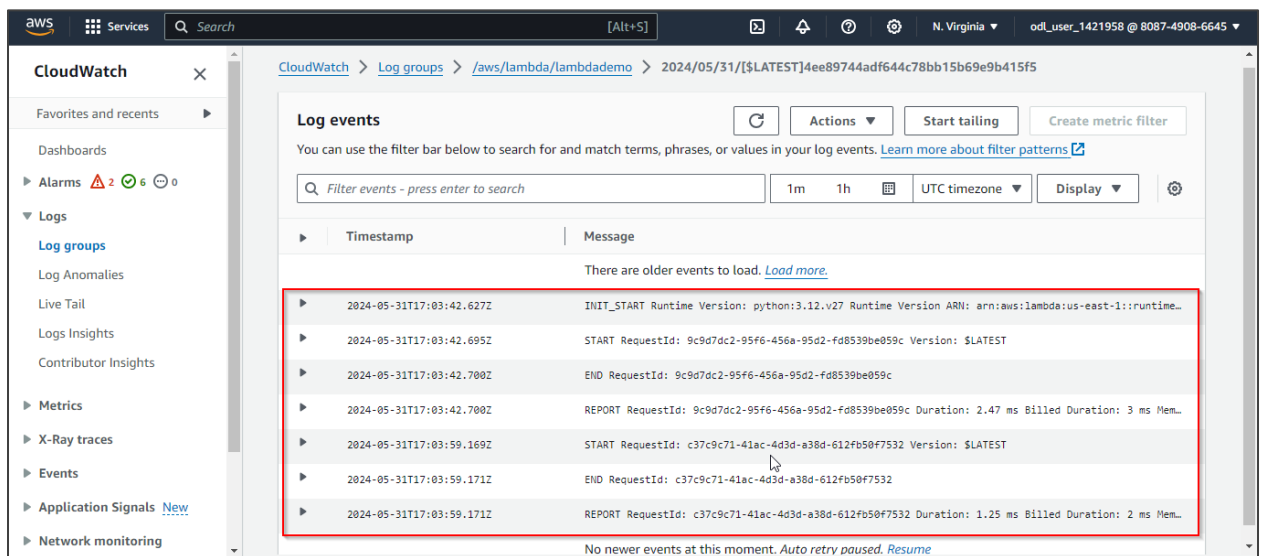
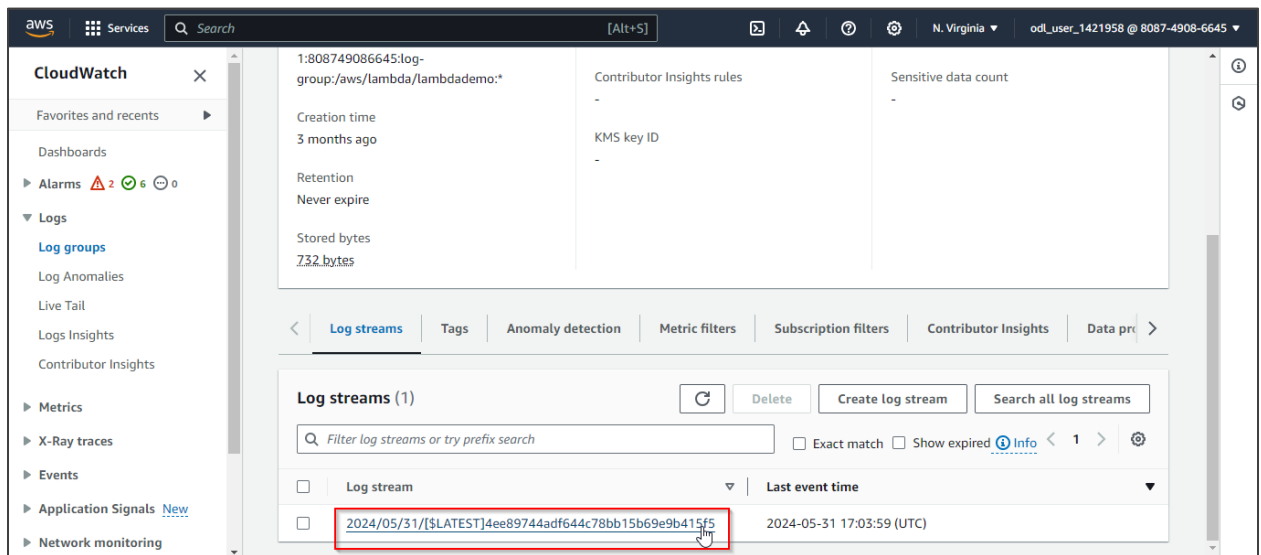
4.1 On the AWS Management Console, search and select **CloudWatch**



4.2 Click on **Log groups** under **Logs** and click on the `/aws/lambda/lamdademo` option



4.3 Click on any **Log streams** to view detailed CloudWatch logs for the Lambda function



The above-mentioned are the CloudWatch logs.

By following these steps, you have successfully created an end-to-end process for working with Lambda functions, from creation to deployment, testing, and log analysis.