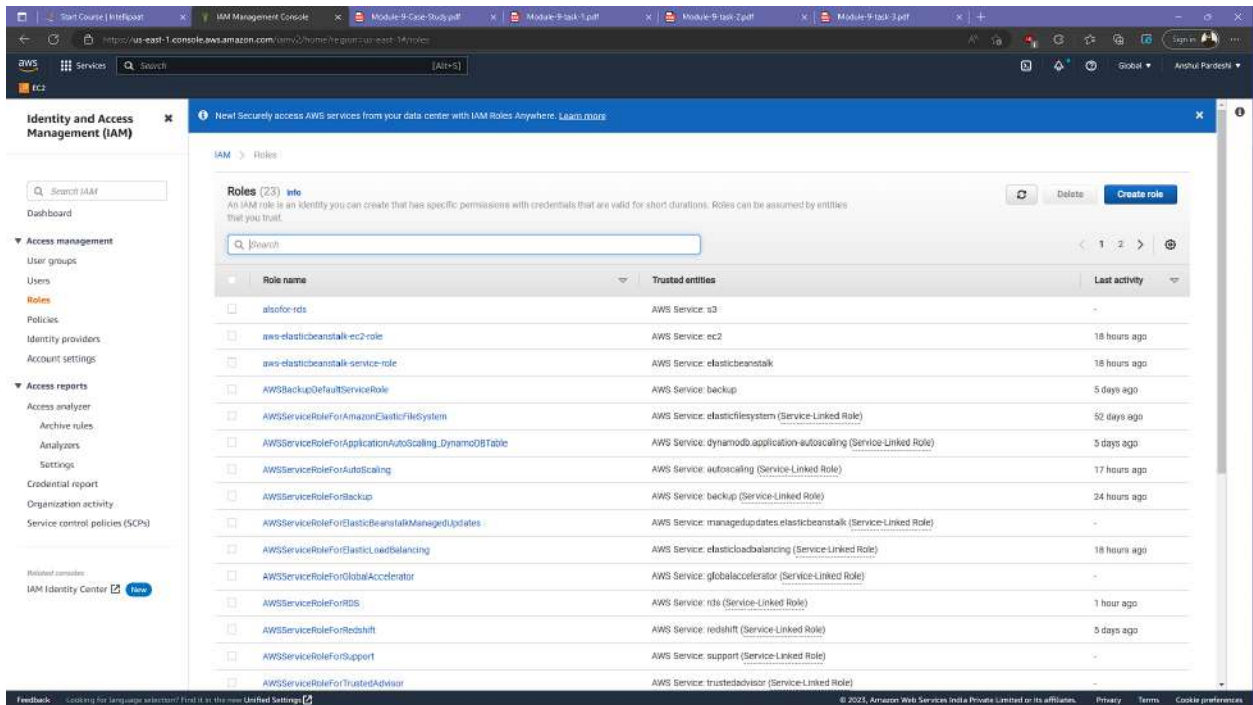


Module-9: Lambda Assignment - 1

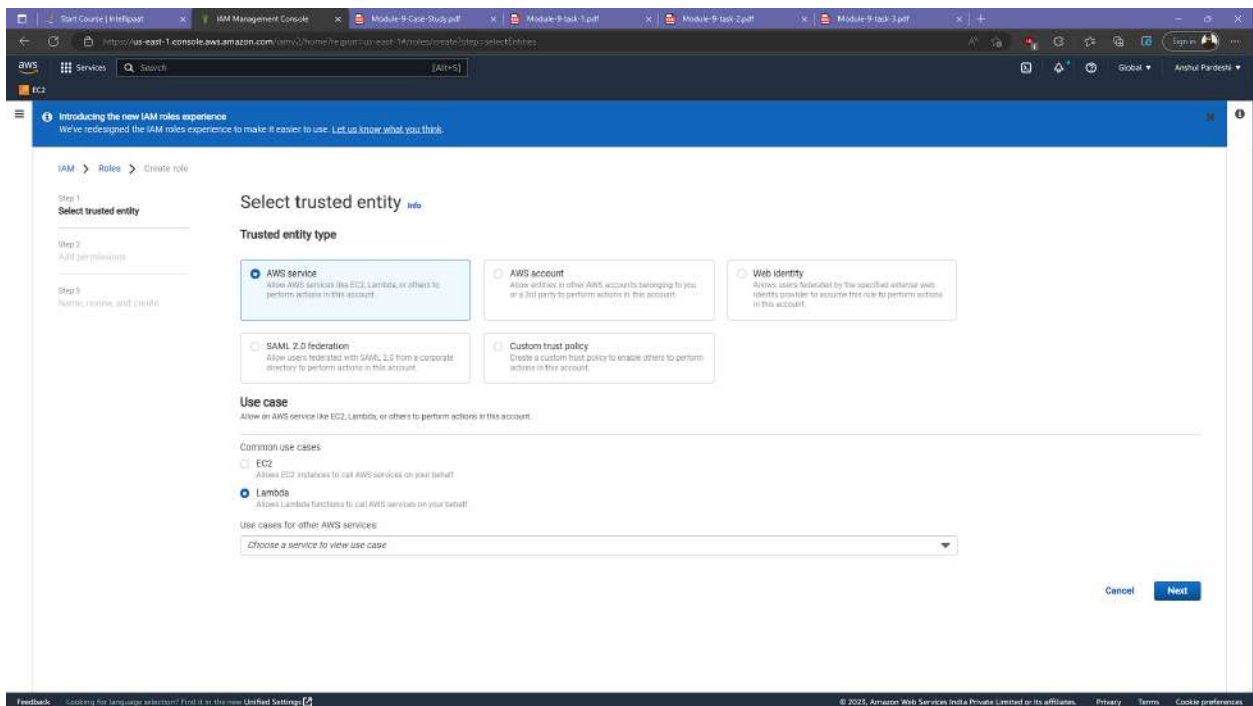
You have been asked to:

1. Create a sample Python Lambda function.
2. Set the Lambda Trigger as SQS and send msg to test invocations.

First create a IAM role which we will need later to attach to lambda.



Choose lambda as a service.



Give SQS Full Access and LambdaSQSExecution access.

The screenshot shows the 'Add permissions' step in the AWS IAM console. The left sidebar indicates the current step is 'Add permissions'. The main area is titled 'Add permissions' and shows 'Permissions policies (Selected 2/810)'. A search bar contains 'sqs' and shows 3 matches. The following table lists the selected policies:

Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonSQSFullAccess	AWS ma...	Provides full access to Amazon SQS via the AWS Management Console.
<input type="checkbox"/> AmazonSQSReadOnlyAccess	AWS ma...	Provides read only access to Amazon SQS via the AWS Management Console.
<input checked="" type="checkbox"/> AWSLambdaSQSQueueExec...	AWS ma...	Provides receive message, delete message, and read attribute access to SQS queues, and write permissions to CloudWatch logs.

Below the table, there is a section for 'Set permissions boundary - optional' with a description: 'Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission management to others.'

At the bottom right, there are buttons for 'Cancel', 'Previous', and 'Next'.

Name it and create a role.

The screenshot shows the 'Name, review, and create' step in the AWS IAM console. The left sidebar indicates the current step is 'Name, review, and create'. The main area is titled 'Name, review, and create' and shows the role name 'for_lambda_assignment' and the description 'Allows Lambda functions to call AWS services on your behalf.'

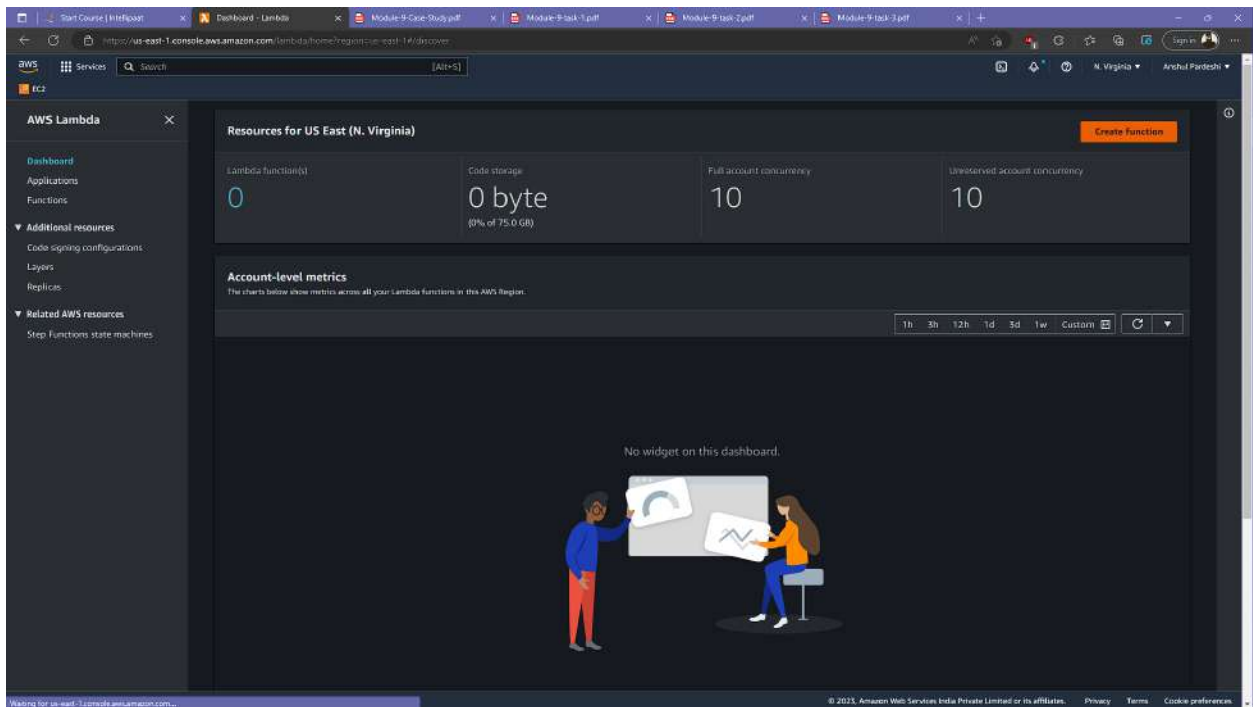
Below the description, there is a section for 'Step 1: Select trusted entities' with a JSON policy document:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "sts:AssumeRole"
8       ],
9       "Principal": {
10        "Service": [
11          "lambda.amazonaws.com"
12        ]
13      }
14    ]
15  }
16 }
```

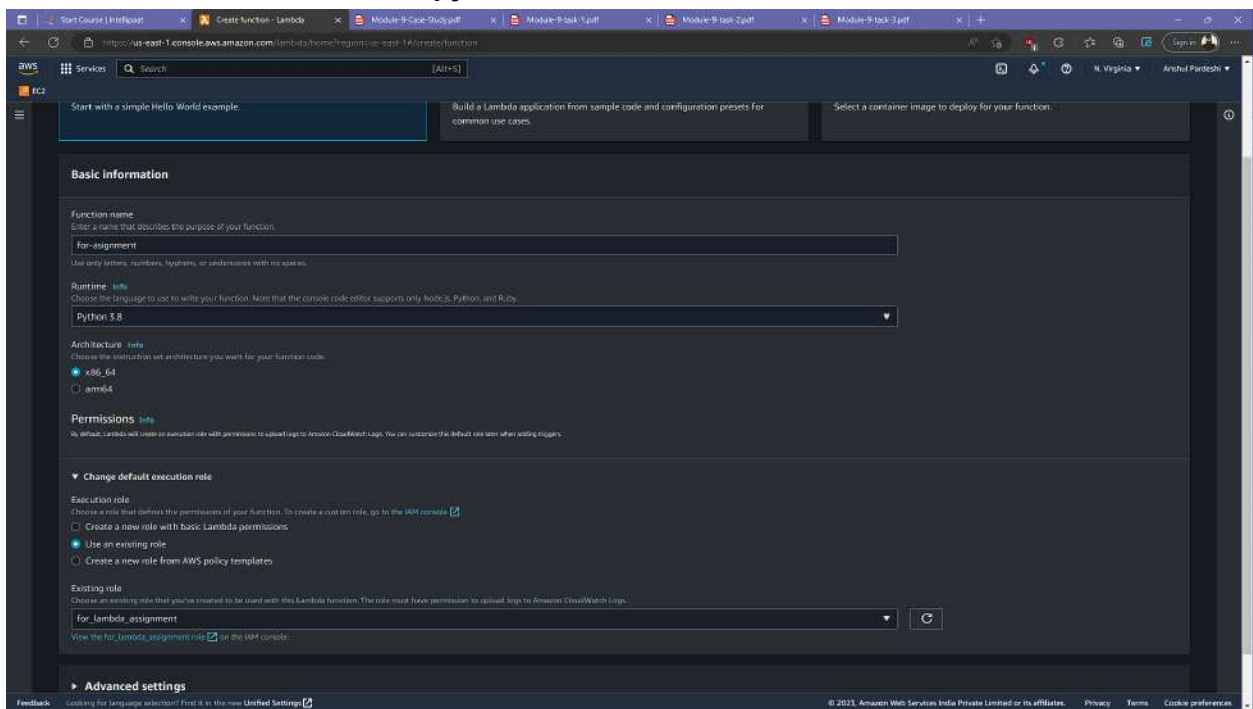
Below the JSON, there is a section for 'Step 2: Add permissions' with a table showing the permissions policy summary:

Policy name	Type	Attached as
AWSLambdaSQSQueueExecutionRole	AWS managed	Permissions policy
AmazonSQSFullAccess	AWS managed	Permissions policy

Lets create lambda function now.



Choose python 3.8 and let others be default.



Lambda function has been created.

The screenshot displays the AWS Lambda console for a function named 'for-assignment'. A green notification bar at the top states: 'Successfully created the function for-assignment. You can now change its code and configuration. To invoke your function with a test event, choose "Test".' The 'Function overview' section shows the function name, a 'Layers' button, and a '+ Add trigger' button. To the right, a 'Description' box indicates the function was last modified '3 seconds ago' and provides the 'Function ARN' and 'Function URL'. Below this, the 'Code source' section is active, showing a code editor with a JavaScript handler function. The code is as follows:

```
import json

def lambda_handler(event, context):
    # TODO: implement
    return {
        'statusCode': 200,
        'body': json.dumps('Hello from Lambda!')}


```

The bottom of the console shows the footer: '© 2021 Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences'.

Now Lets create a SQS resource.

The screenshot shows the AWS SQS console. A green notification bar at the top states: 'Queue for-assignment.mfo has been deleted successfully.' The 'Queues (0)' section is visible, with a search bar and a '+ Create queue' button. Below this, a table header is shown with columns: 'Name', 'Type', 'Created', 'Messages available', 'Messages in flight', 'Encryption', and 'Content-based deduplication'. The table currently contains no entries, and a 'Create queue' button is present at the bottom of the table area. The footer at the bottom of the console reads: '© 2021 Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences'.

Let time period BE 1 MINUTE FOR FASTER RESULTS.

The screenshot shows the 'Create queue' page in the AWS Management Console. The queue type is set to 'Standard'. The name is 'for-assignment'. Under the 'Configuration' section, the 'Message retention period' is set to 1 minute, which is highlighted by a red box. Other settings include a visibility timeout of 30 seconds, a delivery delay of 0 seconds, and a maximum message size of 256 KB.

Type
Choose the queue type for your application or cloud infrastructure.

☒ **Standard** info
At least-once delivery, message ordering isn't preserved.
• At least-once delivery
• Best-effort ordering

☐ **FIFO** info
First-in-first-out delivery, message ordering is preserved.
• First-in-first-out delivery
• Exactly-once processing

You can't change the queue type after you create a queue.

Name
for-assignment
A queue name is case-sensitive and can have up to 80 characters. You can use alphanumeric characters, hyphens (-), and underscores (_).

Configuration
Set the maximum message size, visibility to other consumers, and message retention. [Info](#)

Visibility timeout [Info](#)
30 Seconds
Should be between 0 seconds and 12 hours.

Message retention period [Info](#)
1 Minutes
Should be between 1 minute and 14 days.

Delivery delay [Info](#)
0 Seconds
Should be between 0 seconds and 15 minutes.

Maximum message size [Info](#)
256 KB
Should be between 1 KB and 256 KB.

Receive message wait time [Info](#)
0 Seconds
Should be between 0 and 20 seconds.

Let others things be default.

The screenshot shows the 'Encryption' and 'Access policy' sections of the queue configuration. Under 'Encryption', 'Server-side encryption' is set to 'Enabled' with the 'AWS Key Management Service key (SSE-KMS)' type. Under 'Access policy', the 'Basic' method is selected. The 'Define who can send messages to the queue' section has 'Only the queue owner' selected. The 'Define who can receive messages from the queue' section has 'Only the queue owner' selected. A JSON policy is displayed on the right.

Encryption
Amazon SQS provides in-transit encryption by default. To add at-rest encryption to your queue, enable server-side encryption. [Info](#)

Server-side encryption
☐ Disabled
☒ **Enabled**

Encryption key type
☒ **AWS Key Management Service key (SSE-KMS)**
An encryption key that Amazon SQS creates, manages, and uses for you.
☐ **AWS Key Management Service key (SSE-KMS)**
An encryption key provided by AWS Key Management Service (AWS KMS).

Access policy
Define who can access your queue. [Info](#)

Choose method
☒ **Basic**
Use simple options to define a basic access policy.
☐ **Advanced**
Use a JSON object to define an advanced access policy.

Define who can send messages to the queue
☒ **Only the queue owner**
Only the owner of the queue can send messages to the queue.
☐ **Only the specified AWS accounts, IAM users and roles**
Only the specified AWS account IDs, IAM users and roles can send messages to the queue.

Define who can receive messages from the queue
☒ **Only the queue owner**
Only the owner of the queue can receive messages from the queue.
☐ **Only the specified AWS accounts, IAM users and roles**
Only the specified AWS account IDs, IAM users and roles can receive messages from the queue.

JSON (read-only)

```
{
  "Version": "2008-10-17",
  "Id": "default_policy_ID",
  "Statement": [
    {
      "Sid": "owner_statement",
      "Effect": "Allow",
      "Principal": {
        "AWS": "*"
      },
      "Action": [
        "SQS:*"
      ],
      "Resource": "arn:aws:sqs:us-east-1:994272653802:for-assignment"
    }
  ]
}
```

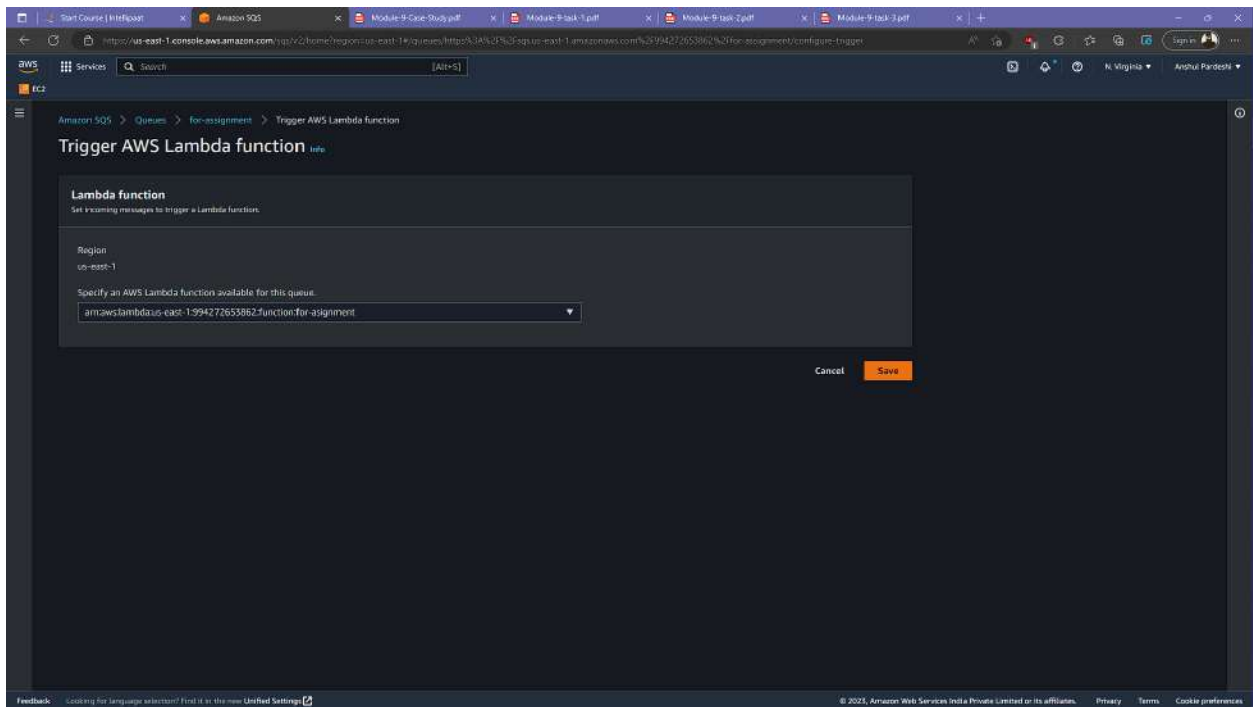
SQS has been created.

The screenshot shows the AWS Management Console for the 'us-east-1' region. A green notification banner at the top states: 'Queue for-assignment created successfully. You can now send and receive messages.' The main content area is titled 'for-assignment' and includes a 'Details' tab. The details section shows the queue's name as 'for-assignment', its type as 'Standard', and its ARN as 'arn:aws:sqs:us-east-1:994272653862:for-assignment'. It also shows the encryption key as 'Amazon SQS key (SSE-SQS)' and the URL as 'https://sqs.us-east-1.amazonaws.com/994272653862/for-assignment'. Below the details, there are tabs for 'SNS subscriptions', 'Lambda triggers', 'Dead-letter queue', 'Monitoring', 'Tagging', 'Access policy', 'Encryption', and 'Dead-letter queue redrive tasks'. The 'SNS subscriptions' tab is active, showing a table with no subscriptions. A 'Subscribe to Amazon SNS topic' button is visible.

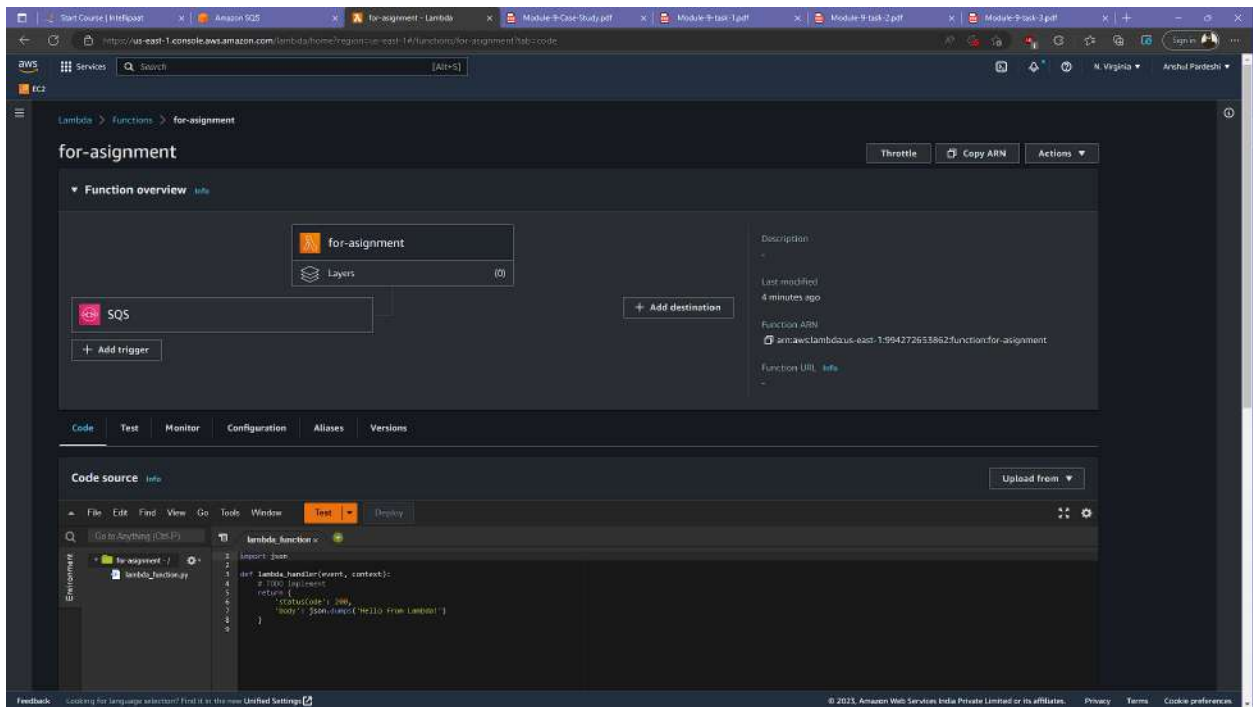
Now, assign a lambda trigger to it.

This screenshot shows the same AWS Management Console page, but with the 'Lambda triggers' tab selected. The 'Details' section remains the same. The 'Lambda triggers' section shows a table with no triggers. A 'Configure Lambda function trigger' button is visible. The notification banner at the top is still present.

Choose the Lambda Service that we just created.



SQS trigger has been applied.



Start Course | Introduction | Amazon SQS | for-assignment - Lambda | Module 9-Que Study.pdf | Module 9-task-1.pdf | Module 9-task-2.pdf | Module 9-task-3.pdf

https://us-east-1.console.aws.amazon.com/sqs/v2/home?region=us-east-1#/queues/http%3A%2F%2Fsqs.us-east-1.amazonaws.com%2F994272653862%2Ffor-assignment

Services Search [Alt+S]

EC2

Lambda function ant:aws:lambda:us-east-1:994272653862:function:for-assignment is triggered when a message arrives in this queue.

Amazon SQS > Queues > for-assignment

for-assignment Edit Delete Purge Send and receive messages Start DLQ redrive

Details info

Name for-assignment	Type Standard	ARN arn:aws:sqs:us-east-1:994272653862:for-assignment
Encryption Amazon SQS key (SSE-SQS)	URL https://sqs.us-east-1.amazonaws.com/994272653862/for-assignment	Dead-letter queue -

More

SMS subscriptions Lambda triggers Dead-letter queue Monitoring Tagging Access policy Encryption Dead-letter queue redrive tasks

Lambda triggers (1) info

Search triggers

UUID	ARN	Status	Last modified
4f832a17-14a4-4ee6-a0db-0cf1664f9107	ant:aws:lambda:us-east-1:994272653862:function:for-assignment	Creating (- 1 minute)	1/15/2023, 12:08:10 PM

Feedback Looking for language selection? Find it in the new Unified Settings

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Send message through SQS.

Start Course | Introduction | Amazon SQS | for-assignment - Lambda | Module 9-Que Study.pdf | Module 9-task-1.pdf | Module 9-task-2.pdf | Module 9-task-3.pdf

https://us-east-1.console.aws.amazon.com/sqs/v2/home?region=us-east-1#/queues/http%3A%2F%2Fsqs.us-east-1.amazonaws.com%2F994272653862%2Ffor-assignment/send-receive

Services Search [Alt+S]

EC2

Amazon SQS > Queues > for-assignment > Send and receive messages

Send and receive messages

Send messages to and receive messages from a queue.

Send message info

Clear content Send message

Message body
Enter the message to send to the queue.

Test Message To trigger Lambda Function

Delivery delay info

0 Seconds

Should be between 0 seconds and 15 minutes.

Message attributes - Optional info

Receive messages info

Edit poll settings Stop polling Poll for messages

Messages available 0 Polling duration 30 Maximum message count 10 Polling progress 0% (0 messages/poll)

Messages (0)

Search messages

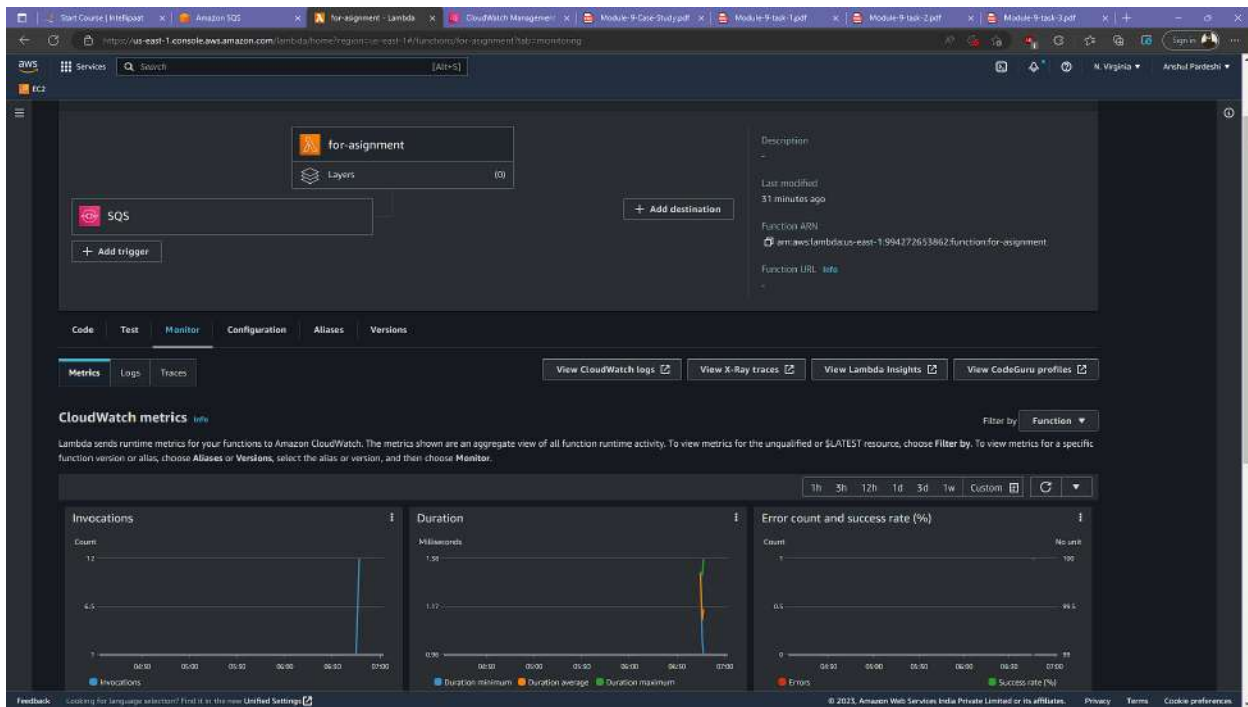
ID	Sent	Size	Receive count
No messages. To view messages in the queue, poll for messages.			

Poll for messages

Feedback Looking for language selection? Find it in the new Unified Settings

© 2023, Amazon Web Services India Private Limited or its affiliates. Privacy Terms Cookie preferences

Those messages triggered lambda.



You can see invocations in cloudwatch and logs.

