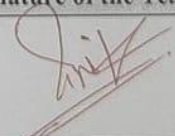


Experiment 8

Title:	Exploring AWS Lambda: Serverless Event Handling		LO2
Student Name	Mark Lopez	Roll no.	9913
Sign here to indicate that you have read all relevant material provided/ available on Moodle/ Classroom while performing and writing this experiment			

Rubrics:

Criteria	Excellent	Good	Satisfactory	Poor	Total Marks
Understanding of AWS Lambda (1)	Demonstrates comprehensive understanding of AWS Lambda's core concepts, capabilities, and application in various scenarios. (4)	Shows a good understanding of AWS Lambda but lacks detail in some areas. (3)	Basic understanding demonstrated, with significant gaps in comprehending its capabilities and applications. (2)	Minimal or incorrect understanding of AWS Lambda's core concepts and capabilities. (1)	4
Creating, Deploying, and Managing Lambda Functions (R2)	Successfully creates, deploys, and manages Lambda functions with no errors, and optimizes resource use. (4)	Creates and deploys Lambda functions with minor errors or inefficiencies in resource use. (3)	Manages to create and deploy Lambda functions but with significant errors or inefficiencies. (2)	Fails to correctly create or deploy Lambda functions. (1)	4
Handling and Processing Messages with Lambda (R3)	Flawlessly handles and processes messages from SQS queues through triggered Lambda functions, demonstrating a deep understanding of event-driven architecture. (4)	Adequately handles and processes messages, but with minor misunderstandings or inefficiencies in the process. (3)	Manages to handle and process messages, but with significant errors or misunderstandings. (2)	Fails to correctly handle or process messages from SQS queue (1)	4
Post lab Completion and Functionality (R4)	Fully completes the task with all functionalities working correctly and efficiently. Demonstrates a thorough understanding and application of AWS services relevant to the task. (6)	Completes most of the task with minor parts missing or not fully functional. Shows a good understanding of AWS services with minor inaccuracies in application. (4-5)	Completes the basic requirements of the task but lacks some functionalities or has notable errors. Basic understanding of AWS services with several inaccuracies. (2-3)	Fails to complete the task or the completed parts have significant errors. Limited or incorrect application of AWS (1)	6
Timeliness of Submission (R5)	On time (2)	1-week late (1)	2-weeks late (0)	More than 2 weeks late (Deduct up to 5 marks)	2
Total Marks					15
Date of Performance		Date of Submission		Signature of the Teacher	
					

Lambda > Functions > ProcessSQSMessages

ProcessSQSMessages

ThrottleCopy ARNActions

Function overviewInfo

DiagramTemplate

ProcessSQSMessages

Layers(0)

SQS

+ Add trigger

+ Add destination

Export to Infrastructure ComposerDownload

Description-

Last modified5 days ago

Function ARNarn:aws:lambda:ap-south-1:202533503126:function:ProcessSQSMessages

Function URLInfo

CodeTestMonitorConfigurationAliasesVersions

Code sourceInfo

Upload from

EXPLORER

PROCESSSQSMESSAGES

lambda_function.py

InfoTutorials

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more

Start tutorial

Event JSON

Format JSON

```
1 {
2   "Records": [
3     {
4       "messageId": "19dd0b57-b21e-4ac1-bd88-01bbb068cb78",
5       "receiptHandle": "MessageReceiptHandle",
6       "body": "Hello from SQS!",
7       "attributes": {
8         "ApproximateReceiveCount": "1",
9         "SentTimestamp": "1523232000000",
10        "SenderId": "123456789012",
11        "ApproximateFirstReceiveTimestamp": "1523232000001"
12      },
13      "messageAttributes": {},
14      "md5OfBody": "{{md5_of_body}}",
15      "eventSource": "aws:sqs",
16      "eventSourceARN": "arn:aws:sqs:us-east-1:123456789012:MyQueue",
17      "awsRegion": "us-east-1"
18    }
19  ]
20 }
```

1:1 JSON Spaces: 2

Code sourceInfo

Upload from

EXPLORER

PROCESSSQSMESSAGES

lambda_function.py

lambda_function.py

1 import json
2 def lambda_handler(event, context):
3 for record in event['Records']:
4 message_body = record['body']
5 print(f'Processing message: {message_body}')
6 # Add your processing logic here
7 return {
8 'statusCode': 200,
9 'body': json.dumps('Success')
10 }

DEPLOY

Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+I)

TEST EVENTS [SELECTED: TEST_1]

PROBLEMS

OUTPUT

CODE REFERENCE LOG

TERMINAL

Execution Results

Status: Succeeded

Test Event Name: test_1

Response:

```
{
  "statusCode": 200,
  "body": "\\\"Success\\\""}

```

Function Logs:

START RequestId: f10ea8a1-6370-48b5-a4f6-74e36618ba39 Version: \$LATEST

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](#)

Clear

1m

30m

1h

12h

Custom (5m)

UTC timezone

Display

Timestamp	Message
No more records within selected time range Retry	
2025-04-29T16:09:58.587Z	INIT_START Runtime Version: python:3.12.v63 Runtime Version ARN: arn:aws:lambda:ap-south-1::runtime:a9ad87288e268d48b360b9ad249ea7a1ad429d836646f1b75871d93349...
2025-04-29T16:09:58.681Z	START RequestId: 5943bf53-22aa-4d1c-b581-b934bb3dc915 Version: \$LATEST
2025-04-29T16:09:58.682Z	Processing message: Hello from SQS!
2025-04-29T16:09:58.685Z	END RequestId: 5943bf53-22aa-4d1c-b581-b934bb3dc915
2025-04-29T16:09:58.685Z	REPORT RequestId: 5943bf53-22aa-4d1c-b581-b934bb3dc915 Duration: 2.13 ms Billed Duration: 3 ms Memory Size: 128 MB Max Memory Used: 35 MB Init Duration: 89.70 ...
2025-04-29T16:11:15.642Z	START RequestId: f10ea8a1-6370-48b5-a4f6-74e36618ba39 Version: \$LATEST
2025-04-29T16:11:15.642Z	Processing message: Hello from SQS!
2025-04-29T16:11:15.648Z	END RequestId: f10ea8a1-6370-48b5-a4f6-74e36618ba39
2025-04-29T16:11:15.648Z	REPORT RequestId: f10ea8a1-6370-48b5-a4f6-74e36618ba39 Duration: 5.14 ms Billed Duration: 6 ms Memory Size: 128 MB Max Memory Used: 35 MB
No more records within selected time range Auto retry paused . Resume	

ChatConnect Pro

Join a Chat Room

Enter a room ID to start messaging

Join

ChatConnect Pro

Room: 100

Connected

09:46 PM

mark

tsting lambda

09:46 PM

mark

this is mark lopes

09:46 PM

mark

Hello testing for dynamo db at 21:46

mark

Type a message...

Send

<input type="checkbox"/>	RoomId (String) ▾	Timestamp (Number) ▾	Message ▾	Sender ▾
<input type="checkbox"/>	100	1745812933	jkbfdj	Pratyay
<input type="checkbox"/>	100	1745812934	jkbfdj	Pratyay
<input type="checkbox"/>	100	1745813035	Flood	mark
<input type="checkbox"/>	100	1745815229	Hi	Mark
<input type="checkbox"/>	100	1745815233	hello	mogu
<input type="checkbox"/>	100	1745815243	Mofu mofu	Mark
<input type="checkbox"/>	100	1745815594	uvu	mogu
<input type="checkbox"/>	100	1745818695	tsting lambda	mark
<input type="checkbox"/>	100	1745818833	this is mark lopes	mark
<input type="checkbox"/>	100	1745943381	Hello testing for dynamo db at 21:46	mark

☰ [Lambda](#) > [Functions](#) > InsertChatMessage

InsertChatMessage

Throttle

 Copy ARN

Actions ▾

▼ Function overview [Info](#)

Diagram

Template



InsertChatMessage



Layers

(0)



API Gateway

+ Add trigger

+ Add destination

Export to Infrastructure Composer

Download ▾


Description

-

Last modified

5 days ago

Function ARN

 arn:aws:lambda:ap-south-1:202533503126:function:InsertChatMessage

Function URL [Info](#)

-

Code

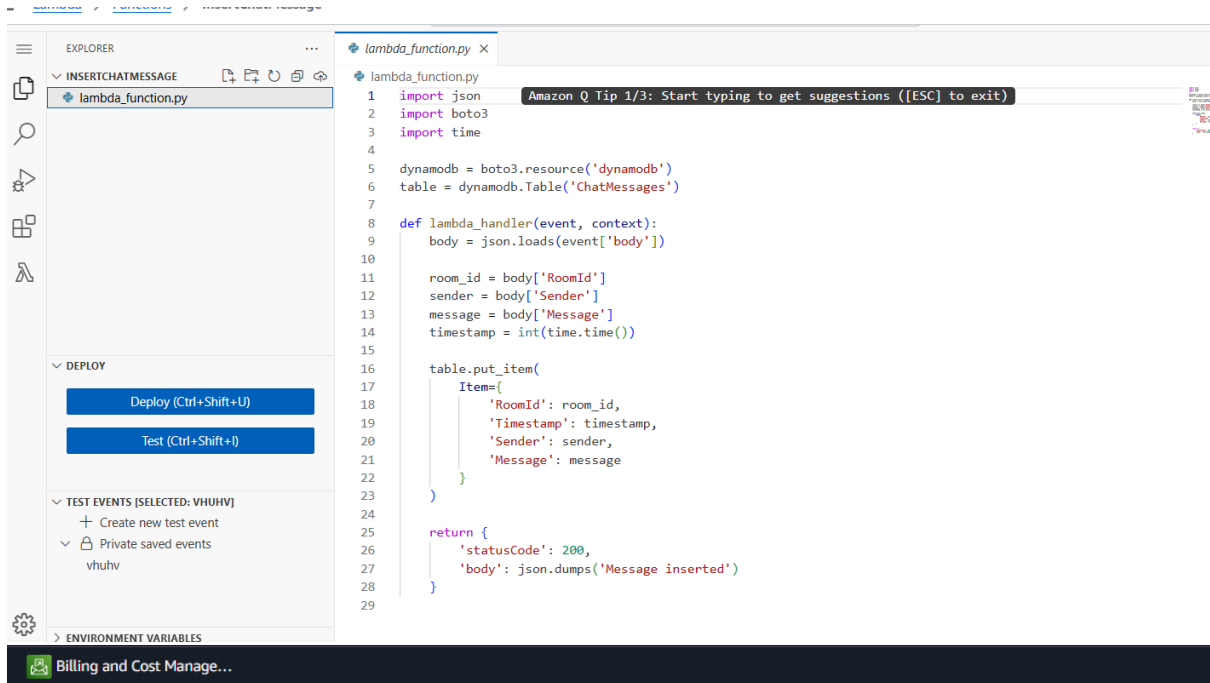
Test

Monitor

Configuration

Aliases

Versions





[Lambda](#) > [Functions](#) > GetChatMessages


GetChatMessages

▼ Function overview [Info](#)

[Diagram](#) | [Template](#)

 **GetChatMessages**

 Layers (0)


 **API Gateway**

[+ Add trigger](#)

[+ Add destination](#)

Desc
-

Last
5 day

Func
 tChar

Func
-

[Code](#) | [Test](#) | [Monitor](#) | [Configuration](#) | [Aliases](#) | [Versions](#)

Code source

EXPLORE

GETCHATMESSAGES

lambda_function.py

DEPLOY

Deploy (Ctrl+Shift+U)

Test (Ctrl+Shift+I)

TEST EVENTS [NONE SELECTED]

Create new test event

ENVIRONMENT VARIABLES

lambda_function.py

```
1 import json
2 import boto3
3 from boto3.dynamodb.conditions import Key
4 from decimal import Decimal
5
6 dynamodb = boto3.resource('dynamodb')
7 table = dynamodb.Table('ChatMessages') # Make sure this matches your actual table name
8
9 class DecimalEncoder(json.JSONEncoder):
10     def default(self, obj):
11         if isinstance(obj, Decimal):
12             # Convert Decimal to standard number types (int or float)
13             if obj % 1 == 0:
14                 return int(obj)
15             else:
16                 return float(obj)
17         return super(DecimalEncoder, self).default(obj)
18
19 def lambda_handler(event, context):
20     # Check if queryStringParameters exists and has RoomId
21     if 'queryStringParameters' not in event or not event['queryStringParameters'] or 'RoomId' not in event
22         return {
23             'statusCode': 400,
24             'headers': {
25                 'Access-Control-Allow-Origin': '*', # Adjust for security
26                 'Content-Type': 'application/json'
27             },
28             'body': json.dumps({'error': 'RoomId query parameter is required'})
29         }
30
31     room_id = event['queryStringParameters']['RoomId']
```

Billing and Cost Management

API Gateway > APIs > ChatAPI (g7cnmc9bpf)

API Gateway

APIs

Custom domain names

Domain name access associations

VPC links

API: ChatAPI ... (g7cnmc9bpf)

Develop

Routes

Authorization

Integrations

CORS

Reimport

Export

Deploy

Stages

Monitor

Metrics

Logging

ChatAPI

Stage: - Deploy

Edit

API details

API ID g7cnmc9bpf

Description No Description

ARN arn:aws:apigateway:ap-south-1::apis/g7cnmc9bpf

Protocol HTTP

IP address type IPv4

Created 2025-04-24

Default endpoint Enabled https://g7cnmc9bpf.execute-api.ap-south-1.amazonaws.com

Stages for ChatAPI (1)

Find resources

Stage name Invoke URL Attached deployment Auto deploy Last updated

default https://g7cnmc9bpf.execute-api.ap-south-1.amazonaws.com djch2a enabled 2025-04-24

Tags (0)

Find resources

Manage tags

Key Value

CloudShell Feedback © 2025, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShellFeedback

Billing and Cost Manage...

API Gateway > APIs > Routes - ChatAPI (g7cnmc9bpf)

Q

API Gateway

APIs

Custom domain names

Domain name access associations

VPC links

API: ChatAPI ... (g7cnmc9bpf)

▼ Develop

Routes

Authorization

Integrations

CORS

Reimport

Export

▼ Deploy

Stages

▼ Monitor

Metrics

Logging

Routes

Routes for ChatAPI

Create

Search

▼ /GetChatMessages

GET

▼ /InsertChatMessage

POST

Stage: -

Deploy

Route details

GET /GetChatMessages (ID: 1adhv3o)

Delete

Edit

ARN

arn:aws:apigateway:ap-south-1::apis/g7cnmc9bpf/routes/1adhv3o

Authorization

Authorizers protect your API against unauthorized requests. Routes with no authorization attached are open.

No authorizer attached to this route.

Attach authorization

Integration

The integration is the backend resource that this route calls when it receives a request.

bhombt6

Configure

CloudShellFeedback

Billing and Cost Manage...

API Gateway > APIs > Routes - ChatAPI (g7cnmc9bpf)

Q

API Gateway

APIs

Custom domain names

Domain name access associations

VPC links

API: ChatAPI ... (g7cnmc9bpf)

▼ Develop

Routes

Authorization

Integrations

CORS

Reimport

Export

▼ Deploy

Stages

▼ Monitor

Metrics

Logging

Routes

Routes for ChatAPI

Create

Search

▼ /GetChatMessages

GET

▼ /InsertChatMessage

POST

Stage: -

Deploy

Route details

POST /InsertChatMessage (ID: c6keltg)

Delete

Edit

ARN

arn:aws:apigateway:ap-south-1::apis/g7cnmc9bpf/routes/c6keltg

Authorization

Authorizers protect your API against unauthorized requests. Routes with no authorization attached are open.

No authorizer attached to this route.

Attach authorization

Integration

The integration is the backend resource that this route calls when it receives a request.

ipx6jvl

Configure