

Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

AWS Management Console | Create function - Lambda | NewJeans (뉴진스) 'Attention' | +

← → C 🔍 us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/create/function?firstrun=true

oracle class JLP_Kyat AWS Academy Cloud aws educate aws rampup shebuild schedule VGW SharePoint Teamup AWS Solutions Arch... JP sound

aws Services Search [Alt+S] N. Virginia vocabs/user2211147=Nwe_Nwe_Htay_Win @ 6754-3515-0030

Lambda > Functions > Create function

Create function Info

AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch Start with a simple Hello World example.

Use a blueprint Build a Lambda application from sample code and configuration presets for common use cases.

Container image Select a container image to deploy for your function.

Basic information

Function name Enter a name that describes the purpose of your function.
myFunctionName

Runtime Info Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Node.js 16.x

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

Type here to search

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

In 4 days ENG 6:44 PM US 11/16/2022

AWS Management Console | Create function - Lambda | NewJeans (뉴진스) 'Attention' | +

lecture.com/courses/29564/assignments/249244?module_item_id=2466886

S Academy Cloud aws educate aws rampup shebuild schedule VGW SharePoint Teamup

gnments

id Lab - Implementing a Serverless Architecture with AWS Lambda

Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

No Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab End Lab 2:57 Instructions Grades Ad

EN_US

Fire hose streams. This IAO provides you with a pre-written Lambda function, so you will use the **Author from scratch** option.

7. Configure the following settings:

- Function name: Load-Inventory
- Runtime: Python 3.7
- Expand **Choose or create an execution role**.
- Execution role: **Use an existing role**
- Existing role: **Lambda-Load-Inventory-Role**

This role gives the Lambda function permissions so that it can access Amazon S3 and DynamoDB.

8. Choose **Create function**

9. Scroll down to the **Code source** section, and in the **Environment** pane, choose `lambda_function.py`.

AWS Management Console | Create function - Lambda | NewJeans (뉴진스) 'Attention' | +

← → C 🔍 us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/create/function

oracle class JLP_Kyat AWS Academy Cloud aws educate aws rampup shebuild schedule VGW SharePoint Teamup AWS Solutions Arch... JP sound

aws Services Search [Alt+S] N. Virginia vocabs/user2211147=Nwe_Nwe_Htay_Win @ 6754-3515-0030

Lambda > Functions > Create function

Create function Info

AWS Serverless Application Repository applications have moved to [Create application](#).

Author from scratch Start with a simple Hello World example.

Use a blueprint Build a Lambda application from sample code and configuration presets for common use cases.

Basic information

Function name Enter a name that describes the purpose of your function.
Load-Inventory

Runtime Info Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.7

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

Type here to search

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

In 4 days ENG 6:45 PM US 11/16/2022

AWS Management Console | **NewJeans (뉴진스) Hyp... | +**

lecture.com/courses/29564/assignments/249244?module_item_id=2466886

S Academy Clou... aws educate aws rampup shebuild schedule VGW SharePoint Teamup

gnments

Lab - Implementing a Serverless Architecture with AWS Lambda

Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

No Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab End Lab 2:57 Instructions Grades

EN_US Files README Terminal Source

Previous screen This lab provides you with a pre-written Lambda function, so you will use the **Author from scratch** option.

7. Configure the following settings:

- Function name: Load-Inventory
- Runtime: Python 3.7
- Choose or create an execution role.
- Execution role: Use an existing role
- Existing role: Lambda-Load-Inventory-Role

This role gives the Lambda function permissions so that it can access Amazon S3 and DynamoDB.

8. Choose **Create function**

9. Scroll down to the **Code source** section, and in the **Environment** pane, choose `lambda_function.py`.

Create function

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

© 2022, Amazon Web Services, Inc. or its affiliates.

In 4 days ENG 6:46 PM US 11/16/2022

AWS Management Console | **NewJeans (뉴진스) Hyp... | +**

lecture.com/courses/29564/assignments/249244?module_item_id=2466886

S Academy Clou... aws educate aws rampup shebuild schedule VGW SharePoint Teamup

gnments

Lab - Implementing a Serverless Architecture with AWS Lambda

Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

No Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab End Lab 2:56

EN_US Files README

Previous screen This role gives the Lambda function permissions so that it can access Amazon S3 and DynamoDB.

8. Choose **Create function**

9. Scroll down to the **Code source** section, and in the **Environment** pane, choose `lambda_function.py`.

10. In the code editor, delete all the code.

11. In the **Code source** editor, copy and paste the following code:

```
# Load-Inventory Lambda Function
#
```

Load-Inventory - Lambda

Successfully created the function Load-Inventory. You can now change its code and configuration. To invoke your function, choose "Test".

Lambda > Functions > Load-Inventory

Load-Inventory

Throttle Copy ARN

Function overview Info

Load-InVENTORY	Description
Layers (0)	Last modified 2 seconds ago
+ Add trigger	Function ARN am:aws:lambda:us-east-1:50030:function:Load-InVENTORY
+ Add destination	Function URL Info

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

© 2022, Amazon Web Services, Inc. or its affiliates.

Privacy

In 4 days ENG 6:46 PM US 11/16/2022

The screenshot shows the AWS Management Console interface. On the left, a sidebar displays course information: 'com/courses/29564/assignments/249244?module_item_id=2466886'. The main content area is titled 'ule 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda'. It includes sections for 'Due Date', 'Points', and 'Submitting an external tool'. A 'Submit' button is present. Below this, there's a 'JS' dropdown menu with options: 'Expand > Choose or create an execution role.', 'Execution role: Use an existing role.', and 'Existing role: Lambda:Load-Inventory-Role.' A note states: 'This role gives the Lambda function permissions so that it can access Amazon S3 and CloudWatch Metrics.' Step 8, 'Choose Create function', is highlighted. Step 9, 'Scroll down to the Code source section, and in the Environment pane, choose lambda_function.', is also visible. Step 10, 'In the code editor, delete all the code.', is mentioned. Step 11, 'In the Code source editor, copy and paste the following code:', is followed by a code snippet:

```
# Load-Inventory Lambda function
#
# This role gives the Lambda function permissions so that it can access Amazon S3 and CloudWatch Metrics.
```

The right side of the screenshot shows the AWS Lambda console for the 'Load-Inventory' function. It displays a success message: 'Successfully created the function Load-Inventory. You can now change its code and configuration. To invoke your function, choose "Test".'. The 'Code source' tab is selected, showing the 'lambda_function' file with the provided code. The AWS Lambda interface includes tabs for 'File', 'Edit', 'Find', 'View', 'Go', 'Tools', 'Window', 'Test', and 'Deploy'.

The screenshot shows the AWS Management Console interface again. The 'Code source' editor has been updated with the following code:

```
# Load-Inventory Lambda function
#
# This function is triggered by an object being created in an Amazon S3 bucket.
# The file is downloaded and each line is inserted into a DynamoDB table.
import json, urllib, boto3, csv
# Connect to S3 and DynamoDB
```

The right side of the screenshot shows the AWS Lambda console for the 'Load-Inventory' function. It displays a success message: 'Successfully updated the function Load-Inventory.'. The 'Code source' tab is selected, showing the 'lambda_function' file with the updated code. The AWS Lambda interface includes tabs for 'File', 'Edit', 'Find', 'View', 'Go', 'Tools', 'Window', 'Test', and 'Deploy'.

Click Deploy after pasting the code

```
# Load-Inventory Lambda function
#
# This function is triggered by an object being created in an Amazon S3 bucket.
# The file is downloaded and each line is inserted into a DynamoDB table.
import json, urllib, boto3, csv
# Connect to S3 and DynamoDB
```

```

s3 = boto3.resource('s3')
dynamodb = boto3.resource('dynamodb')
# Connect to the DynamoDB tables
inventoryTable = dynamodb.Table('Inventory');
# This handler is run every time the Lambda function is triggered
def lambda_handler(event, context):
    # Show the incoming event in the debug log
    print("Event received by Lambda function: " + json.dumps(event, indent=2))
    # Get the bucket and object key from the Event
    bucket = event['Records'][0]['s3']['bucket']['name']
    key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'])
    localFilename = '/tmp/inventory.txt'
    # Download the file from S3 to the local filesystem
    try:
        s3.meta.client.download_file(bucket, key, localFilename)
    except Exception as e:
        print(e)
        print('Error getting object {} from bucket {}. Make sure they exist and your bucket is in the same
region as this function.'.format(key, bucket))
        raise e
    # Read the Inventory CSV file
    with open(localFilename) as csvfile:
        reader = csv.DictReader(csvfile, delimiter=',')
        # Read each row in the file
        rowCount = 0
        for row in reader:
            rowCount += 1
            # Show the row in the debug log
            print(row['store'], row['item'], row['count'])
            try:
                # Insert Store, Item and Count into the Inventory table
                inventoryTable.put_item(
                    Item={
                        'Store': row['store'],
                        'Item': row['item'],
                        'Count': int(row['count'])})
            except Exception as e:
                print(e)
                print("Unable to insert data into DynamoDB table".format(e))
        # Finished!
        return "%d counts inserted" % rowCount

```

Examine the code. It performs the following steps:

- Download the file from Amazon S3 that triggered the event
- Loop through each line in the file

- Insert the data into the DynamoDB *Inventory* table

Screenshot 1: S3 Bucket Creation

The screenshot shows a guided lab interface and the AWS Management Console. In the lab interface, step 14 asks to choose a bucket name, and step 15 specifies entering 'inventory-<number>'. The AWS console shows the 'Create bucket' wizard with the bucket name 'inventory-16339' and the AWS Region set to 'US East (N. Virginia) us-east-1'. The 'General configuration' tab is selected.

Screenshot 2: S3 Bucket Properties

The screenshot shows the AWS Management Console with the 'Properties' tab selected for the 'inventory-16339' bucket. It displays basic information like the AWS Region (US East (N. Virginia) us-east-1), Amazon Resource Name (ARN) (arn:aws:s3:::inventory-16339), and Creation date (November 16, 2022 (UTC+06:30)). The 'Bucket overview' section shows that Bucket Versioning is disabled.

Screenshot 3: Lambda Function Configuration

The screenshot shows the AWS Management Console with the 'Lambda' service selected. A Lambda function named 'Load-Inventory' is being configured. The 'Event source mapping' section is visible, showing a mapping from the 'inventory-16339' S3 bucket to the Lambda function. The 'Edit' button for the mapping is highlighted.

The image shows three screenshots of a guided lab titled "Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda".

Screenshot 1: The lab interface shows a progress bar at 100% completion. The task list includes:

- 17. Choose the name of your *inventory-* bucket.
- 18. Choose the **Properties** tab.
- 19. Scroll down to **Event notifications**.

Instructions state: You will configure an event to trigger when an object is created in the S3 bucket.

Screenshot 2: The "Event notifications" configuration page for the "inventory-16339" bucket. It shows a table with one row:

Name	Event types
	No event notifications

Buttons include "Edit", "Delete", and "Create event notification". A note says: Choose **Create event notification** to be notified when a specific event occurs.

Screenshot 3: The "Create event notification" configuration page. It has fields for:

- Event name:** Load-Inventory
- Prefix - optional:** images/
- Suffix - optional:** .jpg

Event types: A section for selecting events to trigger notifications. It includes:

- Object creation:**
 - All object create events s3:ObjectCreated:*
 - Put s3:ObjectCreated:Put
 - Post s3:ObjectCreated:Post
 - Copy

Full Album New Jeans (뉴진스) +

ture.com/courses/29564/assignments/249244?module_item_id=2466886

IS Academy Clou... AWS Educate AWS Rampup shebuild schedule VGW

gnments

ed Lab - Implementing a Serverless Architecture with AWS Lambda

Module 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

No Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab End Lab 2:50

EN_US

19. Scroll down to **Event notifications**.
You will configure an event to trigger when an object is created in the S3 bucket.

20. Click **Create event notification** then configure these settings:

- Name: Load-Inventory
- Event types: All object create events
- Destination: Lambda Function
- Lambda function: Load-Inventory
- Choose **Save changes**

When an object is created in the bucket, this configuration tells Amazon S3 to invoke the Load-Inventory Lambda function that you created earlier.

Load-Inventory - Lambda AWS Management Console inventory-16339 - S3 bucket

AWS Services Search [Alt+S] Global v vclabs/user2211147=Nwe_Nw...

Destination

Before Amazon S3 can publish messages to a destination, you must grant the Amazon S3 principal the necessary permissions to call the relevant API to publish messages to an SNS topic, an SQS queue, or a Lambda function. Learn more ↗

Destination Choose a destination to publish the event. Learn more ↗

Lambda function Run a Lambda function script based on S3 events.

SNS topic Fanout messages to systems for parallel processing or directly to people.

SQS queue Send notifications to an SQS queue to be read by a server.

Specify Lambda function Choose from your Lambda functions Enter Lambda function ARN

Lambda function Load-Inventory

Cancel Save changes

Type here to search

Open This PC Desktop

Organize New folder

OneDrive - Person This PC Desktop

Internet Download Accelerator Internet Download Manager inventory-berlin inventory-calcutta inventory-karachi inventory-pusan inventory-shanghai

Documents Downloads Music Pictures Videos Local Disk (C:) Data (D:)

inventory-springfield job jp1 jp2 jp3 Module 13 - GL 2 Question Module 13 GL 2

File name: Open Cancel

Name Folder Type Size

No files or folders You have not chosen any files or folders to upload.

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy

82°F Clear ENG US 11/16/2022

Type here to search

Open This PC Desktop

Organize New folder

OneDrive - Person This PC Desktop

Internet Download Accelerator Internet Download Manager inventory-berlin inventory-calcutta inventory-karachi inventory-pusan inventory-shanghai

Documents Downloads Music Pictures Videos Local Disk (C:) Data (D:)

inventory-springfield job jp1 jp2 jp3 Module 13 - GL 2 Question Module 13 GL 2

File name: Open Cancel

Name Folder Type Size

No files or folders You have not chosen any files or folders to upload.

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy

82°F Clear ENG US 11/16/2022

Inventory System

<https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-200-ACACAD-20-EN/mod13-guided/w...>

Inventory Dashboard

Choose a store to view current inventory levels.

Store	Item	Count
All Stores	Echo Show	18
All Stores	Echo Plus	0
All Stores	Echo Look	10
All Stores	Echo Dot	12
All Stores	Echo (2nd Gen)	19
All Stores	Amazon Tap	15

Inventory

Item	Count
Echo Show	18
Echo Plus	0
Echo Look	10
Echo Dot	12
Echo (2nd Gen)	19
Amazon Tap	15

This page uses an Amazon Cognito identity to retrieve data directly from Amazon DynamoDB.

Inventory System

<https://aws-tc-largeobjects.s3-us-west-2.amazonaws.com/ILT-TF-200-ACACAD-20-EN/mod13-guided/w...>

Inventory Dashboard

Choose a store to view current inventory levels.

Store	Item	Count
All Stores	Echo Show	18
All Stores	Echo Plus	0
All Stores	Echo Look	10
All Stores	Echo Dot	12
All Stores	Echo (2nd Gen)	19
All Stores	Amazon Tap	15

Inventory

Item	Count
Echo Show	18
Echo Plus	0
Echo Look	10
Echo Dot	12
Echo (2nd Gen)	19
Amazon Tap	15

This page uses an Amazon Cognito identity to retrieve data directly from Amazon DynamoDB.

DynamoDB

Tables

Name	Status	Partition key	Sort key	Indexes	Read capacity
Inventory	Active	Store (S)	Item (S)	0	Provisioned

Items returned (6)

Store	Item	Count
Berlin	Amazon Tap	15
Berlin	Echo (2nd Gen)	19
Berlin	Echo Dot	12
Berlin	Echo Look	10
Berlin	Echo Plus	0
Berlin	Echo Show	18

Simple Notification Service

The image shows a dual-monitor setup. The left monitor displays a lab assignment titled "File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda". It includes a table for due date, points, and submission status, and a list of 38 steps. Step 38 is highlighted with a red box and says "38. In the lower half of the page, choose **Create subscription**". The right monitor displays the "Amazon SNS > Topics > Create topic" page. It shows two options: "FIFO (first-in, first-out)" and "Standard". The "Standard" option is selected. Below it, there is a "Name" field containing "NoStock" and a note about character restrictions. At the bottom of the page, there are "Feedback", "Privacy", "Terms", and "Cookie preferences" links, along with a copyright notice for 2022, Amazon Web Services, Inc. or its affiliates.

Create subscription

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

35. Choose **Next step**

36. Scroll to the bottom of the page.

37. Choose **Create topic**

To receive notifications, you must subscribe to the topic. You can subscribe to several methods, such as SMS and email.

38. In the lower half of the page, choose **Create subscription**

Amazon SNS

Type Standard

Subscriptions

Subscriptions (0)

Create subscription

No subscriptions found

Create subscription

Details

Topic ARN: arn:aws:sns:us-east-1:675435150030:NoStock

Protocol: Email

Endpoint: nwenwehtaywin@gmail.com

Info: After your subscription is created, you must confirm it.

Subscription to NoStock created successfully.
The ARN of the subscription is arn:aws:sns:us-east-1:675435150030>NoStock:d44320f1-549f-4aae-93cd-3d63085acad5.

Subscription: d44320f1-549f-4aae-93cd-3d63085acad5

Details	
ARN	Status
arn:aws:sns:us-east-1:675435150030>NoStock:d44320f1-549f-4aae-93cd-3d63085acad5	Pending confirmation
Endpoint	Protocol
nwenwehtaywin@gmail.com	EMAIL
Topic	Subscription Principal
NoStock	arn:aws:iam::675435150030:role/voclabs

Gmail

AWS Notification - Subscription Confirmation

AWS Notifications <no-reply@sns.amazonaws.com> to me 7:23 PM (0 minutes ago)

You have chosen to subscribe to the topic: arn:aws:sns:us-east-1:675435150030>NoStock

To confirm this subscription, click or visit the link below (If this was in error no action is necessary). [Confirm subscription](#)

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to [sns-opt-out](#)

Reply **Forward**

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

Function name: Check-Stock
Runtime: Python 3.7
 Expand ▾ Choose or create an execution role.
 Execution role: Use an existing role
 Existing role: Lambda-Check-Stock-Role
 Create function

This role was configured with permissions to send a notification.

41. Scroll down to the **Code source** section, and in the **Environment** tab:
42. In the code editor, delete all the code.
43. Copy the following code, and in the **Code Source** editor, paste it over the existing code.

```
# Stock Check Lambda Function
```

Basic information

Function name: Check-Stock
Runtime: Python 3.7
Architecture: Lambda

Permissions

By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

Change default execution role

Create a new role with basic Lambda permissions
 Use an existing role
 Create a new role from AWS policy templates

Existing role: Lambda-Check-Stock-Role

Advanced settings

The screenshot shows three windows illustrating the implementation of a Serverless Architecture with AWS Lambda.

Top Window: A browser window showing the AWS Lambda console. The URL is <https://us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/Check-Stock?name=Check-Stock>. The title bar says "Successfully updated the function Check-Stock." The code source tab is selected, displaying the following Python code:

```

1  store = dynamodb.Table('Inventory')
2  item = record['dynamodb']['NewImage']['Item'][S]
3  # Construct message to be sent
4  message = store + ' is out of stock of ' + item
5  print(message)
6  # Connect to SNS
7  sns = boto3.client('sns')
8  alertTopic = 'NoStock'
9  snsTopicArn = [t['TopicArn'] for t in sns.list_topics()['Topics']]
10 if t['TopicArn'].lower().endswith(':'+ alertTopic.lower())]:
11     sns.publish(
12         TopicArn=snsTopicArn,
13         Message=message,
14         Subject='Inventory Alert!',
15         MessageStructure='raw'
16     )
17     # Finished!
18     return 'Successfully processed {} records.'.format(len(event['Records']))
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34

```

Middle Window: A browser window showing the AWS Lambda function details page for "Check-Stock". The URL is <https://us-east-1.console.aws.amazon.com/lambda/functions/Check-Stock>. The title bar says "Check-Stock". The "Function overview" section shows:

- Name:** Check-Stock
- Last modified:** 1 minute ago
- Function ARN:** arn:aws:lambda:us-east-1:675435150030:function:Check-Stock
- Function URL:** [Info](#)

Bottom Window: A browser window showing the assignment submission page for "File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda". The URL is https://pcloud.kw.ac.kr/courses/29564/assignments/249244?module_item_id=249244. The "Code source" tab is selected, showing the same Python code as the top window. The "Test" tab is also visible.

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

If the inventory count is zero, send a message to the user.

You will now configure the function so it triggers when data is added to the DynamoDB.

44. Choose **Deploy** to save your code changes.

45. Scroll to the **Designer** section (which is at the top of the page).

46. Choose **+Add trigger** and then configure these settings:

- Select a trigger: **DynamoDB**
- DynamoDB Table: Inventory**
- Choose **Add**

You are now ready to test the system!

Task 6: Testing the System

You will now upload an inventory file to Amazon S3, which will trigger the function. This function will load data into DynamoDB, which will then trigger the Lambda function.

Trigger configuration

DynamoDB

DynamoDB table

Choose or enter the ARN of a DynamoDB table.

arn:aws:dynamodb:us-east-1:675435150030:table/Inventory

Activate trigger

Select to activate the trigger now. Keep unchecked to create the trigger in a deactivated state for testing (recommended).

Batch size

The number of records in each batch to send to the function.

100

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates.

Privacy Terms Cookie preferences

82°F Clear ENG US 7:27 PM 11/16/2022

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

45. Scroll to the **Designer** section (which is at the top of the page).

46. Choose **+Add trigger** and then configure these settings:

- Select a trigger: **DynamoDB**
- DynamoDB Table: Inventory**
- Choose **Add**

You are now ready to test the system!

Task 6: Testing the System

You will now upload an inventory file to Amazon S3, which will trigger the function. This function will load data into DynamoDB, which will then trigger the Lambda function.

Configuration

Triggers (1)

DynamoDB: Inventory

arn:aws:dynamodb:us-east-1:675435150030:table/Inventory/stream/2022-11-16T12:12:32.224

state: Enabled

Details

Feedback Looking for language selection? Find it in the new Unified Settings [?](#)

© 2022, Amazon Web Services, Inc. or its affiliates.

Privacy Terms Cookie preferences

82°F Clear ENG US 7:27 PM 11/16/2022

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

You will now upload an inventory file to Amazon S3, which will trigger a Lambda function. This function will load data into DynamoDB, which will then trigger another Lambda function. If the Lambda function detects an item with a quantity of zero, it will send a message to an Amazon SNS topic. Then, Amazon SNS will notify you through SMS or email.

47. On the Services menu, choose S3.

48. Choose the name of your *Inventory-* bucket.

49. Choose Upload and upload a different inventory file.

50. Return to the Inventory System Dashboard and refresh the page.

You should now be able to use the Store menu to view the inventory levels.

File 13 Guided Lab - Implementing a Serverless Architecture with AWS Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

47. On the Services menu, choose S3.

48. Choose the name of your *Inventory-* bucket.

49. Choose Upload and upload a different inventory file.

50. Return to the Inventory System Dashboard and refresh the page.

You should now be able to use the Store menu to view the inventory levels.

Also, you should receive a notification through SMS or email (each inventory file has one item that is out of stock).

If you did not receive a notification, wait a few minutes as the DynamoDB trigger can sometimes take a few minutes to execute.

51. Try to upload multiple inventory files at the same time. What happens?

Amazon S3

Buckets

Access Points Object Lambda Access Points Multi-Region Access Points Batch Operations Access analyzer for S3

Block Public Access settings for this account

Storage Lens Dashboards AWS Organizations settings

Feature spotlight

AWS Marketplace for S3

Amazon S3 > Buckets > inventory-16339

inventory-16339 info

Objects (1)

Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Actions Copy S3 URI Copy URL Download Open Delete

Create folder Upload

Find objects by prefix

Name	Type	Last modified	Size	Storage class
inventory-berlin.csv	csv	November 16, 2022, 19:17:24 (UTC+06:30)	146.0 B	Standard

Feedback Looking for language selection? Find it in the new Unified Settings © 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 82°F Clear ENG 7:28 PM US 11/16/2022

Load-Invent... AWS Man... Check-Stock... inventory... Items | Am... inventory... +

s3.console.aws.amazon.com/s3/buckets/inventory-16339?region=us-east-1&tab=objects

Global vclabs/user2211147=Nwe_Nwe_Htay_Win @ 6754-3515-0

Inventory Dashboard

Choose a store to view current inventory levels.

Store:	Store	Item	Count
All Stores	All Stores	Echo Show	32
All Stores	All Stores	Echo Plus	16
All Stores	All Stores	Echo Look	13
Berlin	Berlin	Echo Dot	19
Calcutta	Calcutta	Echo (2nd Gen)	19
All Stores	All Stores	Amazon Tap	30

Inventory

Item	Count
Echo Show	32
Echo Plus	16
Echo Look	13
Echo Dot	19
Echo (2nd Gen)	19
Amazon Tap	30

This page uses an Amazon Cognito identity to retrieve data directly from Amazon DynamoDB.

82°F Clear ENG 7:29 PM US 11/16/2022

Now, two store

Upload all remaining files

질스 노래모음 (가사포함) | ← | → | pCloud :: File Manager

com/courses/29564/assignments/249244?module_item_id=24

ny Clou... aws educate aws rampup shebuild sched

s Implementing a Serverless Architecture with AWS Lambda

ile 13 Guided Lab - Implementing a Se Lambda

Due Date Points 100 Submitting an external tool

Submit Details AWS Start Lab

If you did not receive a notification, wait a few minutes a DynamoDB trigger can sometimes take a few minutes to en

51. Try to upload multiple inventory files at the same time. Wh

Submitting your work

52. At the top of these instructions, choose **Submit** to record choose **Yes**.

53. If the results don't display after a couple of minutes, return choose **Grades**.

Tip: You can submit your work multiple times. After you ch

Choose a store to view current inventory levels.

Inventory Dashboard

Store:	Store	Item	Count
All Stores	All Stores	Echo Show	72
All Stores	All Stores	Echo Plus	59
All Stores	Berlin	Echo Look	81
All Stores	Calcutta	Echo Dot	84
All Stores	Karachi	Echo (2nd Gen)	96
All Stores	Pusan	Amazon Tap	40
All Stores	Shanghai		
All Stores	Springfield		

Inventory

Item	Count
Echo Show	~72
Echo Plus	~59
Echo Look	~81
Echo Dot	~84
Echo (2nd Gen)	~96
Amazon Tap	~40

This page uses an Amazon Cognito identity to retrieve data directly from Amazon DynamoDB.

Type here to search

82°F Clear 7:30 PM 11/16/2022