

Lambda with S3 in AWS

Function name	Description	Package type	Runtime	Last modified
MainMonitoringFunction	-	Zip	Python 3.9	1 month ago
RoleCreationFunction	Create SLR if absent	Zip	Python 3.9	1 month ago
ModLabRole	updates LabRole to allow it to assume itself	Zip	Python 3.9	1 month ago
RedshiftOverwatch	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.9	1 month ago
RedshiftEventSubscription	Create Redshift event subscription to SNS Topic.	Zip	Python 3.9	1 month ago

Amazon S3
Store and retrieve any amount of data from anywhere

Amazon S3 is an object storage service that offers industry-leading scalability, data availability, security, and performance.

Create a bucket

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

Create bucket

Pricing

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Estimate your monthly bill using the [AWS Simple Monthly Calculator](#)

[View pricing details](#)

Resources

[User guide](#)
[API reference](#)

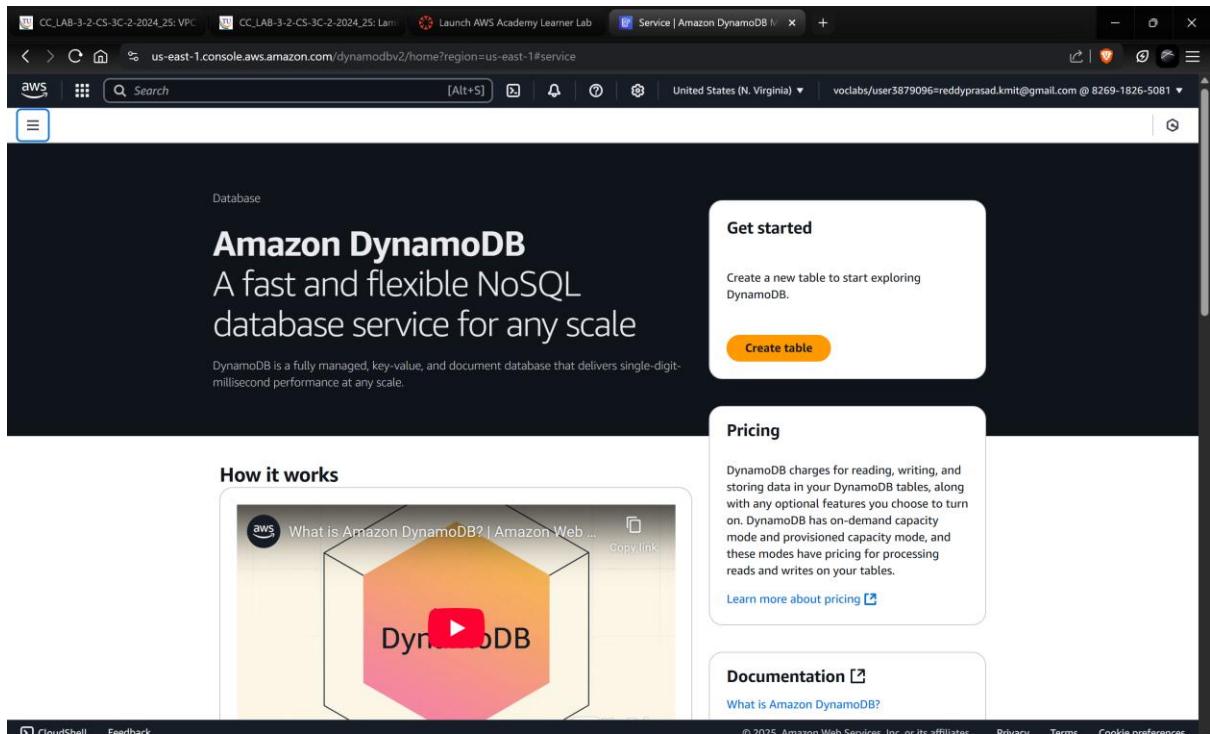
How it works

Introducing Amazon S3 | Amazon Web Services

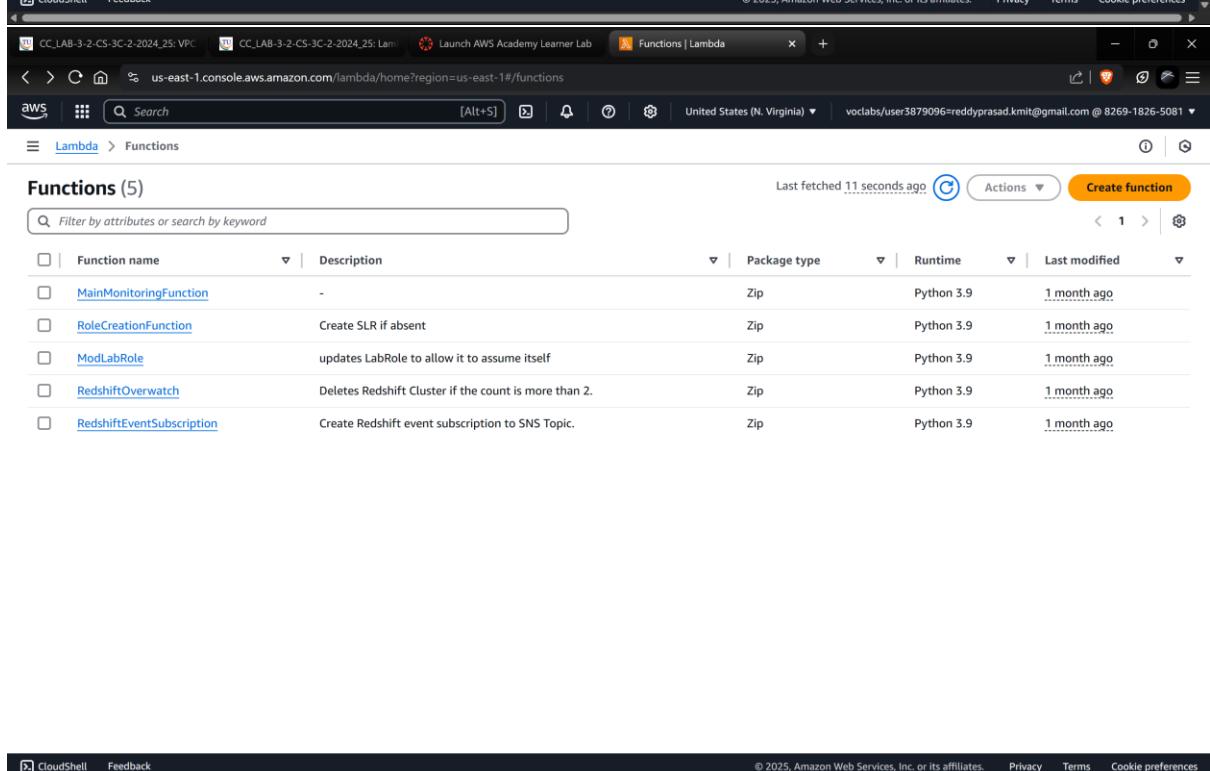
Copy link

Amazon S3

Lambda with S3 in AWS



The screenshot shows the Amazon DynamoDB console homepage. It features a large central image of a hexagonal cube with a play button icon, labeled "DynamoDB". To the left, there's a section titled "How it works" with a video thumbnail. On the right, there are three main sections: "Get started" (with a "Create table" button), "Pricing" (describing usage fees), and "Documentation" (linking to the "What is Amazon DynamoDB?" page). The top navigation bar includes tabs for "VPC", "Lambda", "Launch AWS Academy Learner Lab", and "Service | Amazon DynamoDB". The bottom footer contains links for "CloudShell", "Feedback", "Privacy", "Terms", and "Cookie preferences".



The screenshot shows the AWS Lambda console under the "Functions" tab. It displays a table of five functions, each with a checkbox, function name, description, package type, runtime, and last modified date. The functions listed are:

Function name	Description	Package type	Runtime	Last modified
MainMonitoringFunction	-	Zip	Python 3.9	1 month ago
RoleCreationFunction	Create SLR if absent	Zip	Python 3.9	1 month ago
ModLabRole	updates LabRole to allow it to assume itself	Zip	Python 3.9	1 month ago
RedshiftOverwatch	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.9	1 month ago
RedshiftEventSubscription	Create Redshift event subscription to SNS Topic.	Zip	Python 3.9	1 month ago

The top navigation bar includes tabs for "VPC", "Lambda", "Launch AWS Academy Learner Lab", and "Service | Amazon DynamoDB". The bottom footer contains links for "CloudShell", "Feedback", "Privacy", "Terms", and "Cookie preferences".

Lambda with S3 in AWS

CC_LAB-3-2-CS-3C-2-2024_25: VPC CC_LAB-3-2-CS-3C-2-2024_25: Lam Launch AWS Academy Learner Lab Create function | Functions | Lab +

aws Search [Alt+S] United States (N. Virginia) voclabs/user3879096=reddyprasad.kmit@gmail.com @ 8269-1826-5081

Lambda > Functions > Create function

Create function Info

Choose one of the following options to create your function.

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.

Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

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

Architecture Info
Choose the instruction set architecture you want for your function code.
 x86_64
 arm64

Permissions Info
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

Additional Configurations

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

CC_LAB-3-2-CS-3C-2-2024_25: VPC CC_LAB-3-2-CS-3C-2-2024_25: Lam Launch AWS Academy Learner Lab Create function | Functions | Lab +

aws Search [Alt+S] United States (N. Virginia) voclabs/user3879096=reddyprasad.kmit@gmail.com @ 8269-1826-5081

Lambda > Functions > Create function

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

Architecture Info
Choose the instruction set architecture you want for your function code.
 x86_64
 arm64

Permissions Info
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

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

Existing role
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.
 
View the LabRole role [on the IAM console](#).

Additional Configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.

Create function

Lambda with S3 in AWS

The screenshot shows the 'Create function' wizard in the AWS Lambda console. The page title is 'Create function'. It includes fields for selecting the language (Python 3.9), architecture (x86_64), and execution role (LabRole). A section for 'Additional Configurations' is also visible.

The screenshot shows the 'Create table' wizard in the AWS DynamoDB console. It includes fields for 'Table name' (my_table) and 'Partition key' (unique). A 'Sort key - optional' field is also present. The 'Table settings' section offers 'Default settings' or 'Customize settings' options. The 'Default table settings' section is also visible at the bottom.

Lambda with S3 in AWS

The screenshot shows the 'Create table' page in the AWS DynamoDB console. The table name is 'my_table'. The 'Default table settings' section lists various configuration options:

Setting	Value	Editable after creation
Table class	DynamoDB Standard	Yes
Capacity mode	On-demand	Yes
Maximum read capacity units	-	Yes
Maximum write capacity units	-	Yes
Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Deletion protection	Off	Yes
Resource-based policy	Not active	Yes

Tags
Tags are pairs of keys and optional values, that you can assign to AWS resources. You can use tags to control access to your resources or track your AWS spending.
No tags are associated with the resource.

Add new tag
You can add 50 more tags.

Tables
The my_table table was created successfully.

DynamoDB

- Dashboard
- Tables**
- Explore items
- PartiQL editor
- Backups
- Exports to S3
- Imports from S3
- Integrations
- Reserved capacity
- Settings

DAX

- Clusters
- Subnet groups
- Parameter groups
- Events

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

Lambda with S3 in AWS

The screenshot shows the 'Create bucket' page in the AWS Management Console. The bucket name is 'myawsbucket'. Under 'Bucket type', 'General purpose' is selected. The 'Object Ownership' section shows 'ACLs disabled (recommended)'. The status bar at the bottom right indicates 'voclabs/user3879096=reddyprasad.kmit@gmail.com @ 8269-1826-5081'.

The screenshot shows the 'Create bucket' page in the AWS Management Console. The bucket name is 'kmittankbund'. Under 'Bucket type', 'General purpose' is selected. The 'Object Ownership' section shows 'ACLs disabled (recommended)'. The status bar at the bottom right indicates 'voclabs/user3879096=reddyprasad.kmit@gmail.com @ 8269-1826-5081'.

Lambda with S3 in AWS

The screenshot shows the AWS Lambda function configuration interface. At the top, there are tabs for 'Overview', 'Code', 'Triggers', 'Logs', and 'Environment'. Below the tabs, the 'Triggers' section is active, showing a list of triggers: 'Amazon S3' (selected), 'AWS Lambda', 'CloudWatch Metrics', and 'CloudWatch Logs'. For the 'Amazon S3' trigger, it is set to 'Create objects in my bucket' and 'us-east-1'. The 'IAM Role' dropdown is set to 'Lambda role'. The 'Handler' field contains 'lambda_function.lambda_handler'. The 'Timeout' is set to 300 seconds. The 'Memory' is set to 128 MB. The 'Environment Variables' section is empty. The 'Logs' tab shows log entries for the Lambda function.

Lambda with S3 in AWS

The screenshot shows the 'Add trigger' configuration page for AWS Lambda. At the top, there are tabs for 'Trigger configuration' and 'Info'. A dropdown menu labeled 'Select a source' is open, showing the search term 's3'. Below the dropdown, a list of triggers is displayed, with 'aws s3' selected. On the right side of the page are 'Cancel' and 'Add' buttons.

This screenshot is identical to the one above, but the 'aws s3' trigger has been selected from the dropdown menu. The list below now shows 'aws s3' as the active trigger. The 'Add' button is visible on the right.



Lambda with S3 in AWS

The screenshot shows the 'Add trigger' configuration page for an AWS Lambda function. The top navigation bar includes tabs for 'CloudShell', 'Feedback', and links to 'Launch AWS Academy Learn', 'Add triggers | Lambda', 'List tables | Amazon DynamoDB', and 'S3 buckets | S3 | us-east-1'. The main content area is titled 'Trigger configuration' with an 'Info' link. It shows a selected trigger type 'aws asynchronous storage' under 'Bucket'. A search bar for 'Bucket' contains 's3/kmittankbund' with a region dropdown set to 'us-east-1'. Under 'Event types', 'All object create events' is selected. There are optional fields for 'Prefix - optional' (e.g., 'images/') and 'Suffix - optional' (e.g., '.jpg'). A note about recursive invocation is present, along with a checkbox for acknowledging the use of the same S3 bucket for both input and output. The bottom section contains a note about Lambda adding necessary permissions for AWS S3 to invoke the function, followed by 'Cancel' and 'Add' buttons.

Lambda with S3 in AWS

The screenshot shows the AWS Lambda console interface. A Lambda function named "tankbund" is displayed. It has one trigger, "S3", which is associated with the "my_table" table in Amazon DynamoDB. The "Configuration" tab is selected, showing general configuration details like memory (128 MB) and ephemeral storage (512 MB). A success message indicates that the trigger was successfully added.

Function overview

Configuration

General configuration

Description	Memory	Ephemeral storage
-	128 MB	512 MB

DynamoDB

Tables

Name	Status	Partition key	Sort key	Indexes	Replication Regions	Deletion protection	Favorite	Re
my_table	Active	unique (\$)	-	0	0	Off	☆	Or

Lambda with S3 in AWS

The screenshot shows the AWS DynamoDB console. On the left, the navigation menu includes 'Tables' under 'DynamoDB'. The main area displays 'my_table' details. Key settings shown include:

- General information:** Partition key (unique String), Sort key (-), Capacity mode (On-demand), Item count (0), Table status (Active), Table size (0 bytes).
- Point-in-time recovery (PITR):** Enabled (Off).
- Resource-based policy:** Not active.
- Amazon Resource Name (ARN):** arn:aws:dynamodb:us-east-1:826918265081:table/my_table

The screenshot shows the 'Scan or query items' interface for 'my_table'. The 'Scan' button is selected. The 'Items returned (0)' section indicates no items found.

Scan or query items:

- Scan (selected)
- Query

Select a table or index: Table - my_table | Select attribute projection: All attributes

Items returned (0):

No items
No items to display.

Lambda with S3 in AWS

The screenshot shows the AWS Lambda Functions console. A Lambda function named "tankbund" is selected. The "Function overview" tab is active, displaying a diagram where an "S3" trigger is connected to the "tankbund" function. The "Configuration" tab is also visible, showing the general configuration settings like memory and ephemeral storage.

Function overview

General configuration

Code source

```
lambda_function.py
1 import boto3
2 from uuid import uuid4
3 def lambda_handler(event, context):
4     s3 = boto3.client("s3")
5     dynamodb = boto3.resource('dynamodb')
6     for record in event['Records']:
7         bucket_name = record['s3']['bucket']['name']
8         object_key = record['s3']['object']['key']
9         size = record['s3']['object'].get('size', -1)
10        event_name = record['eventName']
11        event_time = record['eventTime']
12        dynamoTable = dynamodb.Table('newtable')
13        dynamoTable.put_item(
14            Item={'unique': str(uuid4()), 'Bucket': bucket_name, 'Object': object_key, 'Size': size, 'Event': event_name, 'Time': event_time})
```

Lambda with S3 in AWS

The screenshot shows the AWS Lambda function editor for a function named "tankbund". The "Code source" tab is selected. The code in "lambda_function.py" is as follows:

```
import boto3
from uuid import uuid4
def lambda_handler(event, context):
    s3 = boto3.client("s3")
    dynamodb = boto3.resource('dynamodb')
    for record in event['Records']:
        bucket_name = record['s3']['bucket']['name']
        object_key = record['s3']['object']['key']
        size = record['s3']['object'].get('size', -1)
        event_name = record['eventName']
        event_time = record['eventTime']
        dynamoTable = dynamodb.Table('my_table')
        dynamoTable.put_item(
            Item={'unique': str(uuid4()), 'Bucket': bucket_name, 'Object': object_key, 'Size': size, 'Event': event_name, 'Time': event_time})
```

The "DEXPLORER" sidebar shows a folder "TANKBUND" containing "lambda_function.py". Below it, a "DEPLOY [UNDEPLOYED CHANGES]" section indicates "You have undeployed changes." with "Deploy (Ctrl+Shift+U)" and "Test (Ctrl+Shift+I)" buttons.

A success message at the bottom of the screen says "Successfully updated the function tankbund."

Lambda with S3 in AWS

The screenshot shows the AWS Lambda console for the 'tankbund' function. The function is triggered by an S3 event. The status bar indicates an error: 'Executing function: failed (logs [2])'. The 'Test' tab is selected. A red box highlights the error message. The 'Event name' field contains 'lambdatest'. The 'Objects' section of the S3 bucket 'kmittankbund' is shown, displaying 'No objects'.

Function Overview

- Last modified: 2 minutes ago
- Function ARN: arn:aws:lambda:us-east-1:826918265081:function:tankbund
- Function URL: Info

Test

Executing function: failed (logs [2])

▶ Details

Diagnose with Amazon Q

Test event Info

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

lambdatest

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

CloudWatch Logs Live Tail Save Test

CloudShell Feedback

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

The screenshot shows the AWS S3 console for the 'kmittankbund' bucket. The 'Objects' tab is selected, showing 0 objects. The 'Actions' dropdown menu is open, showing options like Copy S3 URI, Copy URL, Download, Open, Delete, Create folder, and Upload. A search bar at the top allows finding objects by prefix. The table below lists columns for Name, Type, Last modified, Size, and Storage class.

kmittankbund Info

Objects | Metadata | Properties | Permissions | Metrics | Management | Access Points

Objects (0)

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](#)

Find objects by prefix: Show versions

Name	Type	Last modified	Size	Storage class
No objects				
You don't have any objects in this bucket.				

Upload

CloudShell Feedback

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

Lambda with S3 in AWS

The screenshot shows the AWS Lambda with S3 upload interface. The top navigation bar includes tabs for CC_LAB-3-2-CS-3C-2-2024_2, CC_LAB-3-2-CS-3C-2-2024_2, Launch AWS Academy Learn, tankbund | Functions | Lambda, Items | Amazon DynamoDB, Upload objects - S3 buck, and a user profile. The main content area shows the upload process:

- Upload Info:** A large text input field with placeholder "Drag and drop files and folders you want to upload here, or choose Add files or Add folder." Below it is a table showing one file: "script.js" (60.0 B, text/javascript).
- Destination Info:** Shows the destination as "s3://kmittankbund". It includes sections for "Destination details" (Bucket settings), "Permissions" (Grant public access and access to other AWS accounts), and "Properties" (Specify storage class, encryption settings, tags, and more).
- Upload status:** A green success message: "Upload succeeded" with a link to "For more information, see the Files and folders table." Below it is a summary table:

Summary	
Destination	Succeeded
s3://kmittankbund	1 file, 60.0 B (100.00%)
	Failed 0 files, 0 B (0%)
- Files and folders:** A table showing the uploaded file: "script.js" (60.0 B, text/javascript, Status: Succeeded).

Lambda with S3 in AWS

The screenshot shows the AWS DynamoDB console interface. On the left, the navigation menu includes 'Dashboard', 'Tables', 'Explore items', 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Under 'DAX', there are 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. The main area displays the 'my_table' table with one item returned. The item details are as follows:

Attribute name	Value	Type	Action
unique - Partition key	ce0e930c-a3e5-4bcd-a3ed-7263df23c4a0	String	
Bucket	kmittankbund	String	Remove
Event	ObjectCreated:Put	String	Remove
EventTime	2025-04-04T05:51:13.554Z	String	Remove
Object	script.js	String	Remove
Size	60	Number	Remove

At the bottom right of the item details, there are buttons for 'Cancel', 'Save', and 'Save and close'.

Lambda with S3 in AWS

The screenshot shows the AWS S3 console interface. The top navigation bar includes tabs for CC_LAB-3-2-CS-3C-2-2024_2, CC_LAB-3-2-CS-3C-2-2024_2, Launch AWS Academy Learn..., tankbund | Functions | Lambda, Items | Amazon DynamoDB, kmittankbund - S3 buck..., and voclabs/user3879096=reddyprasad.kmit@gmail.com @ 8269-1826-5081. The main page displays the 'Objects' section for the 'kmittankbund' bucket. A single object, 'script.js', is listed. The object details show it is a JavaScript file (js) last modified on April 4, 2025, at 11:21:14 (UTC+05:30), with a size of 60.0 B and a storage class of Standard. Action buttons include Copy S3 URI, Copy URL, Download, Open, Delete, Actions (with a dropdown menu), Create folder, and Upload.

The screenshot shows the AWS S3 console 'Upload' page for the 'kmittankbund' bucket. The top navigation bar is identical to the previous screenshot. The main area is titled 'Upload' and contains a large dashed blue box for dragging and dropping files or choosing them via 'Add files' or 'Add folder'. Below this is a table titled 'Files and folders (1 total, 294.0 B)' showing one file, 'index.html', which is a text/html file of size 294.0 B. There are 'Remove', 'Add files', and 'Add folder' buttons. The 'Destination' section shows the destination as 's3://kmittankbund'. The 'Destination details' section notes that bucket settings impact new objects stored there. The 'Permissions' and 'Properties' sections are also visible. The bottom of the page includes standard AWS footer links: CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

Lambda with S3 in AWS

The screenshot shows the AWS Lambda console interface. At the top, there's a summary of upload status: 1 file succeeded (index.html) and 0 files failed. Below this, the 'Files and folders' tab is selected, showing a table with one item: index.html (text/html, 294.0 B, Succeeded). In the main area, the 'Function overview' section for the 'tankbund' function is displayed. It shows the function name, ARN, and a code editor with 'lambda_function.py' containing the following code:

```
import boto3
from uuid import uuid4
```

Below the code editor, there are tabs for Code, Test, Monitor, Configuration, Aliases, and Versions. The URL for the function is also visible at the bottom of the page.

Lambda with S3 in AWS

The screenshot shows the AWS Lambda console interface. At the top, there are several tabs: CC_LAB-3-2-CS-3C-2-2024_2, CC_LAB-3-2-CS-3C-2-2024_2, Launch AWS Academy Learn, tankbund | Functions, Items | Amazon DynamoDB, kmittankbund - S3 bucket, and a search bar. The URL in the address bar is us-east-1.console.aws.amazon.com/lambda/home?region=us-east-1#/functions/tankbund?subtab=triggers&tab=configure.

The main area shows the 'Configuration' tab selected. On the left, a sidebar lists various configuration options: General configuration, Triggers, Permissions, Destinations, Function URL, Environment variables, Tags, VPC, RDS databases, Monitoring and operations tools, and Concurrency and recursion detection. The 'Triggers' section is expanded, showing one trigger named 'S3: kmittankbund'. The ARN of the trigger is listed as arn:aws:s3:::kmittankbund. There are buttons for 'Edit', 'Delete', and 'Add trigger'.

Below the triggers, there is a 'Function ARN' field containing arn:aws:lambda:us-east-1:826918265081:function:tankbund. A 'Function URL' field is also present.

The second screenshot shows the 'Edit trigger' configuration page. The title is 'Edit trigger' under the 'Trigger configuration' section. The 'Bucket' field is populated with 'arn:aws:s3:::kmittankbund'. The 'Event types' section includes options for 'All object create events', 'Permanently deleted', and 'Delete marker created'. Below these, there are fields for 'Prefix - optional' (e.g. images/) and 'Suffix - optional' (e.g. jpg). A note at the bottom states: 'Lambda will add the necessary permissions for AWS S3 to invoke your Lambda function from this trigger. Learn more about the Lambda permissions model.'

Lambda with S3 in AWS

Objects (2)

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](#)

Name	Type	Last modified	Size	Storage class
index.html	html	April 4, 2025, 11:23:02 (UTC+05:30)	294.0 B	Standard
script.js	js	April 4, 2025, 11:21:14 (UTC+05:30)	60.0 B	Standard

Successfully deleted objects
View details below.

Delete objects: status

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

Name	Type	Last modified	Size	Error

Failed to delete (0)

Name	Type	Last modified	Size	Error



Lambda with S3 in AWS

The screenshot shows the AWS DynamoDB console interface. On the left, there's a navigation sidebar with sections like 'Dashboard', 'Tables', 'Explore items', 'PartiQL editor', 'Backups', 'Exports to S3', 'Imports from S3', 'Integrations', 'Reserved capacity', and 'Settings'. Below that is a 'DAX' section with 'Clusters', 'Subnet groups', 'Parameter groups', and 'Events'. The main area is titled 'Scan or query items' and shows a table named 'my_table' selected. It includes options for 'Scan' (selected) or 'Query', 'Select a table or index' (set to 'Table - my_table'), and 'Select attribute projection' (set to 'All attributes'). A 'Filters' section has a 'Run' button and a 'Reset' button. A green message bar at the bottom says 'Completed. Read capacity units consumed: 2'. Below this is a table titled 'Items returned (1/3)' with three rows. The columns are 'unique (String)', 'Bucket', 'Event', 'EventTime', and 'Object'. The first two rows have empty checkboxes. The third row has a checked checkbox and the value '91ccefc0-0955-4cf2-...' under 'unique (String)'. The 'Bucket' column shows 'kmittankbu...', 'Event' shows 'ObjectCreat...', 'EventTime' shows '2025-04-0...', and 'Object' shows 'script.js' and 'index.html' respectively. There are 'Actions' and 'Create item' buttons above the table, and a footer with links to CloudShell, Feedback, Privacy, Terms, and Cookie preferences.

unique (String)	Bucket	Event	EventTime	Object
	kmittankbu...	ObjectCreat...	2025-04-0...	script.js
	kmittankbu...	ObjectCreat...	2025-04-0...	index.html
<input checked="" type="checkbox"/> 91ccefc0-0955-4cf2-...	kmittankbu...	ObjectRem...	2025-04-0...	script.js