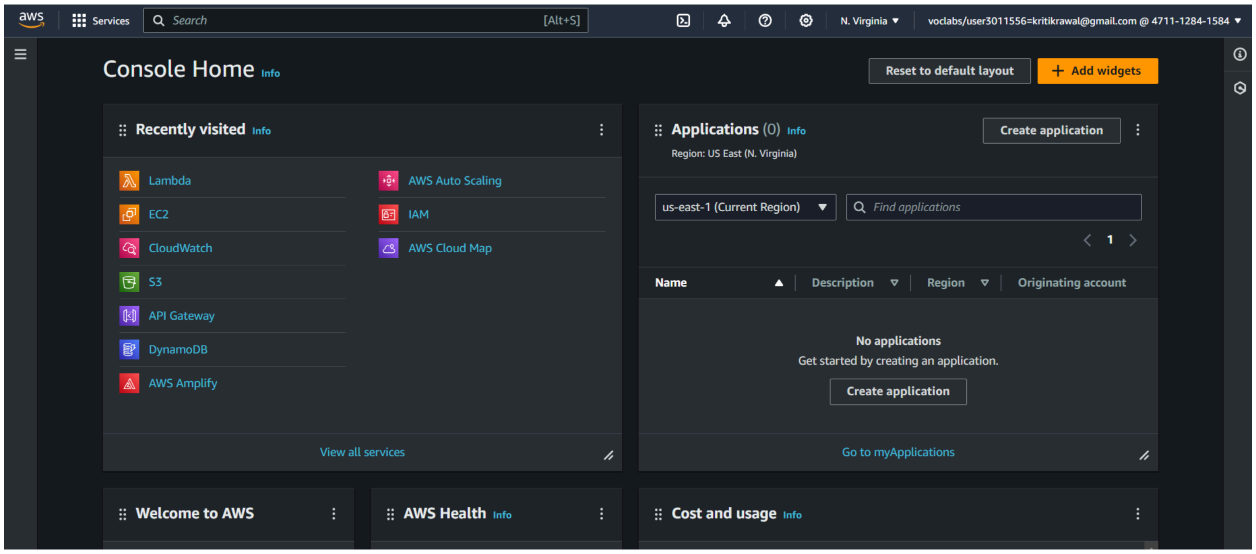
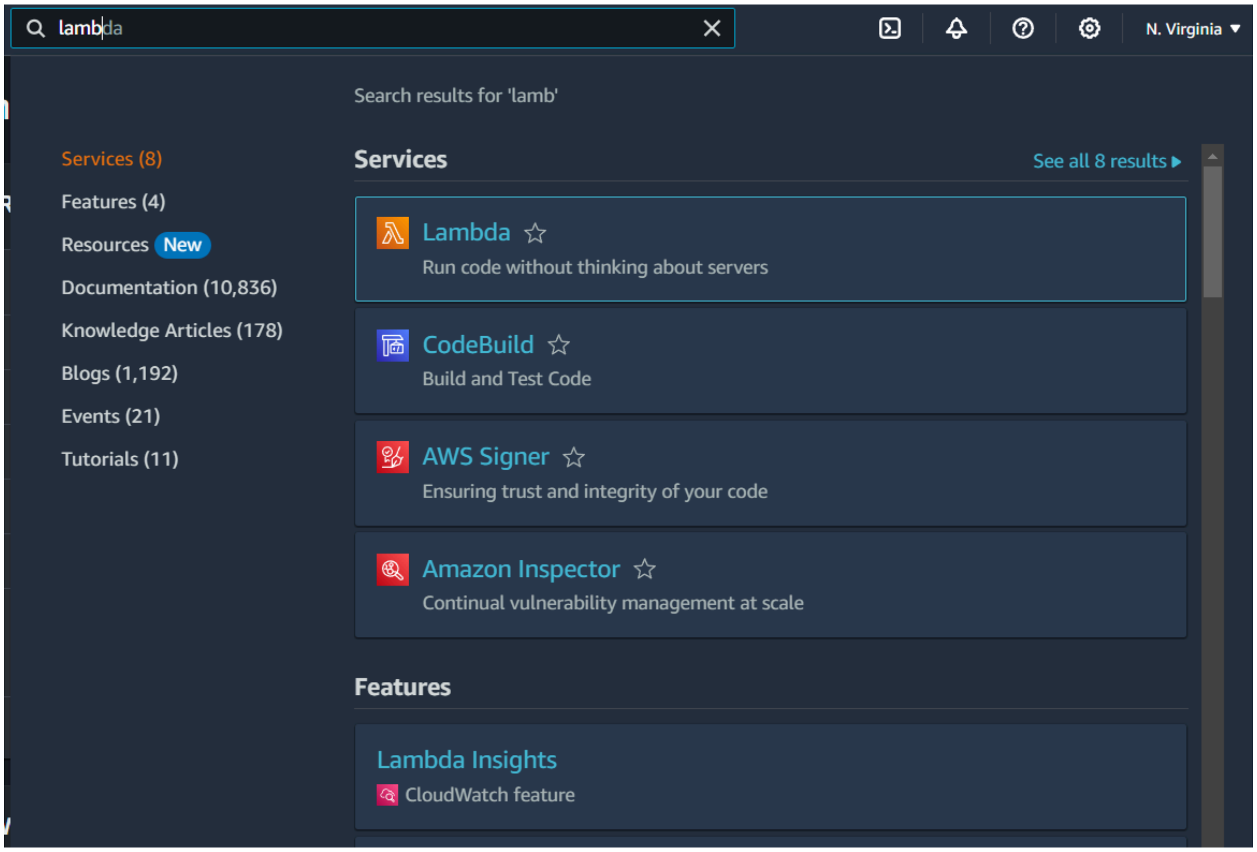
TASK 2:

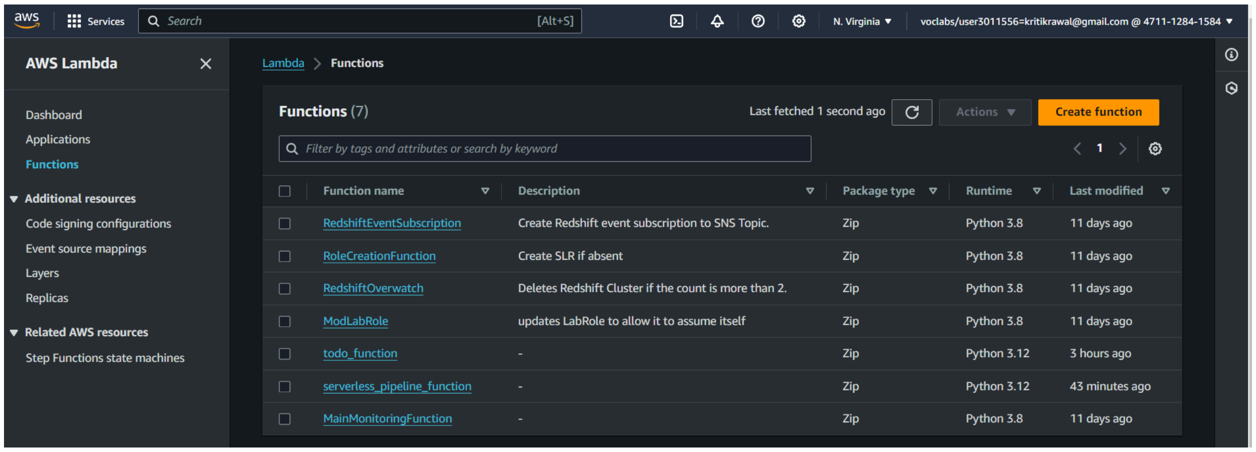
1. Navigate to https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1



2. Search Lambda



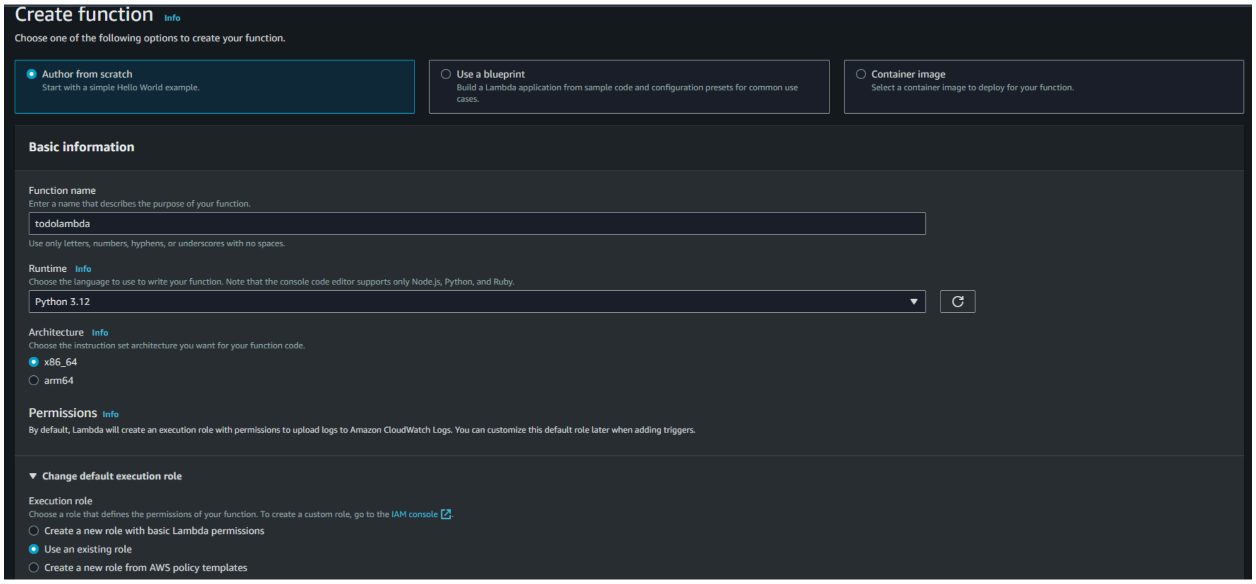
3. Click on the create function.



Functions (7)

Function name	Description	Package type	Runtime	Last modified
RedshiftEventSubscription	Create Redshift event subscription to SNS Topic.	Zip	Python 3.8	11 days ago
RoleCreationFunction	Create SLR if absent	Zip	Python 3.8	11 days ago
RedshiftOverwatch	Deletes Redshift Cluster if the count is more than 2.	Zip	Python 3.8	11 days ago
ModLabRole	updates LabRole to allow it to assume itself	Zip	Python 3.8	11 days ago
todo_function	-	Zip	Python 3.12	3 hours ago
serverless_pipeline_function	-	Zip	Python 3.12	43 minutes ago
MainMonitoringFunction	-	Zip	Python 3.8	11 days ago

4. Specify the function name, select Python 3.12 in runtime and select Use an existing role in Change default execution role.



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.
todolambda
Use only letters, numbers, hyphens, or underscores with no spaces.

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 3.12

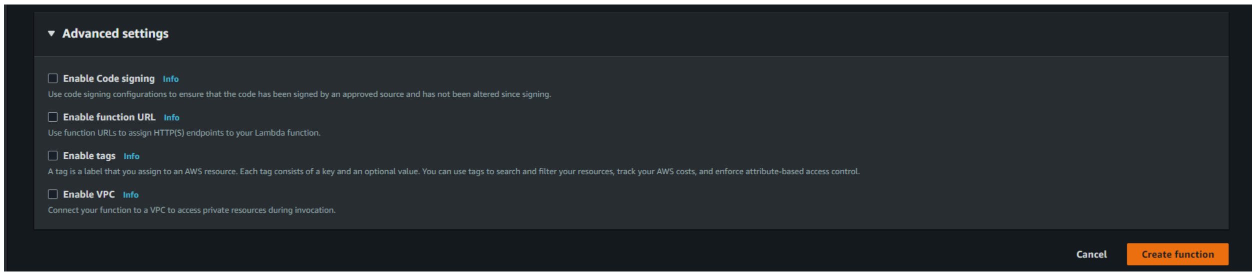
Architecture Info
Choose the instruction set architecture you want for your function code.
(●) x86_64
( ) arm64

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

5. Create the function.



▼ **Advanced settings**

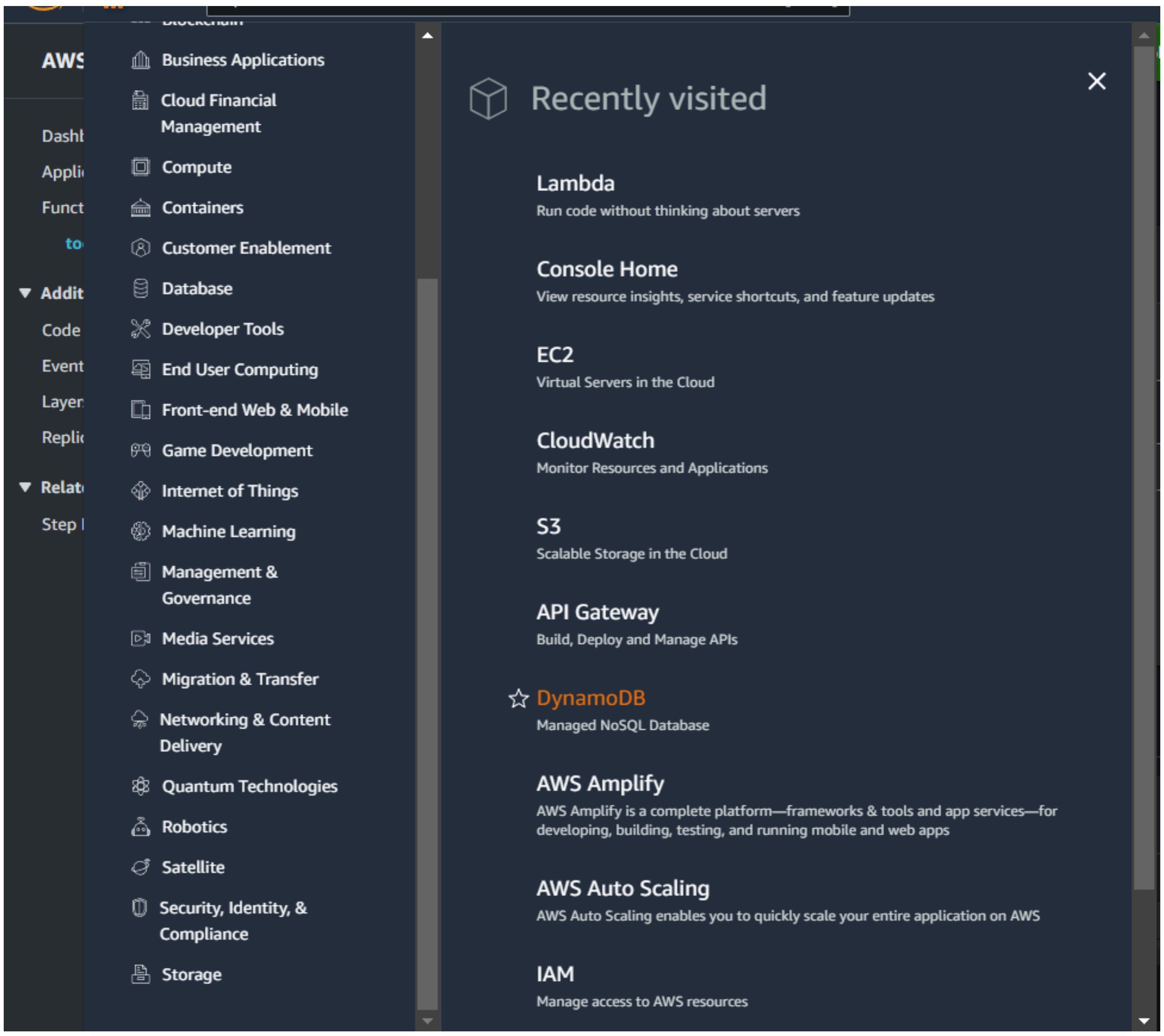
☐ Enable Code signing Info
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.

☐ Enable function URL Info
Use function URLs to assign HTTP(S) endpoints to your Lambda function.

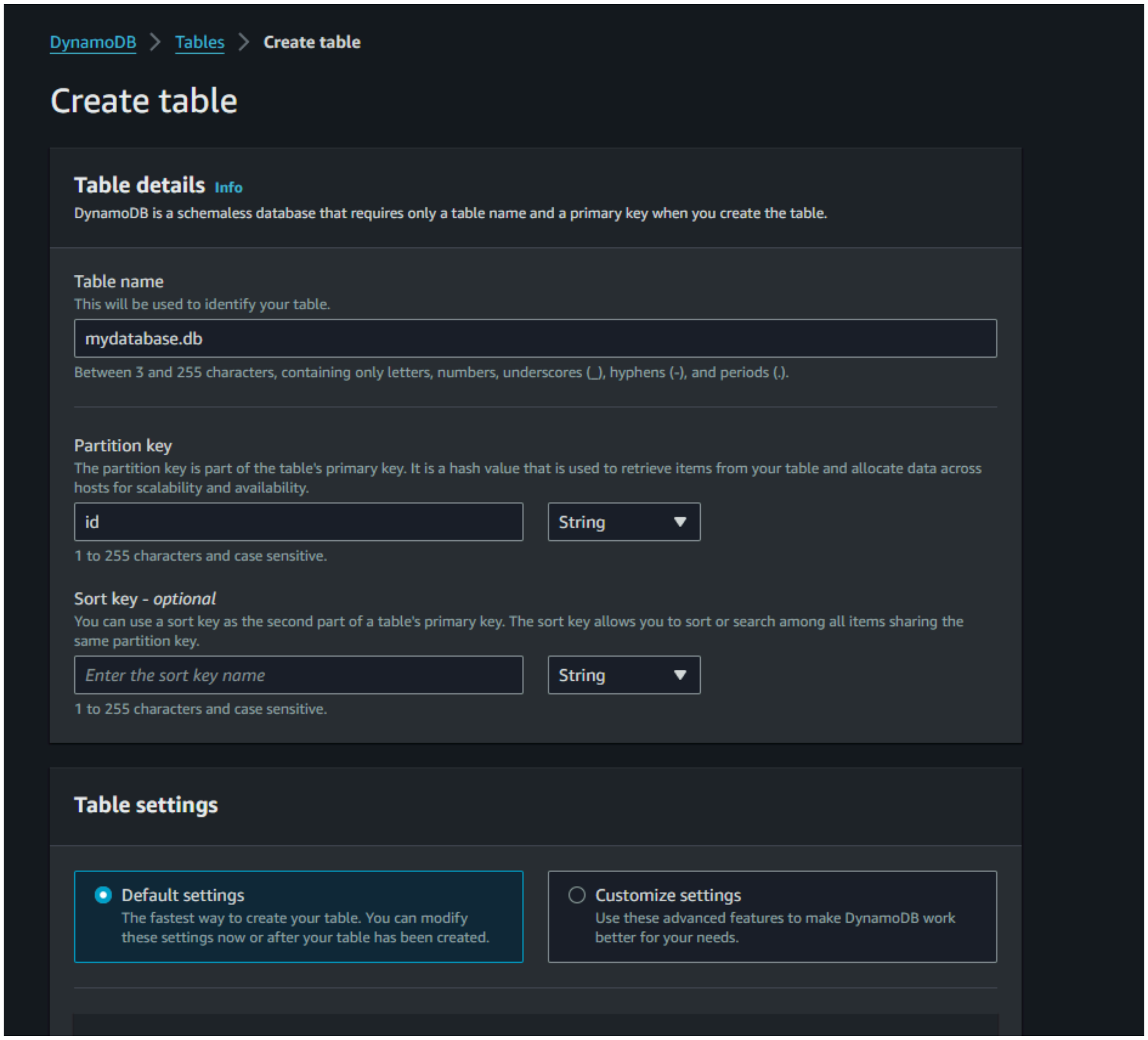
☐ Enable tags Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources, track your AWS costs, and enforce attribute-based access control.

☐ Enable VPC Info
Connect your function to a VPC to access private resources during invocation.

6. Select DynamoDB to create it.


AWS
Dashb
Appli
Funct
to
Addit
Code
Event
Layer
Replic
Relat
Step
Business Applications
Cloud Financial Management
Compute
Containers
Customer Enablement
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
Media Services
Migration & Transfer
Networking & Content Delivery
Quantum Technologies
Robotics
Satellite
Security, Identity, & Compliance
Storage
Recently visited
Lambda
Run code without thinking about servers
Console Home
View resource insights, service shortcuts, and feature updates
EC2
Virtual Servers in the Cloud
CloudWatch
Monitor Resources and Applications
S3
Scalable Storage in the Cloud
API Gateway
Build, Deploy and Manage APIs
DynamoDB
Managed NoSQL Database
AWS Amplify
AWS Amplify is a complete platform—frameworks & tools and app services—for developing, building, testing, and running mobile and web apps
AWS Auto Scaling
AWS Auto Scaling enables you to quickly scale your entire application on AWS
IAM
Manage access to AWS resources

7. Create the table.



DynamoDB > Tables > Create table

Create table

Table details Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

mydatabase.db

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

id — String

1 to 255 characters and case sensitive.

Sort key - *optional*

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

Enter the sort key name — String

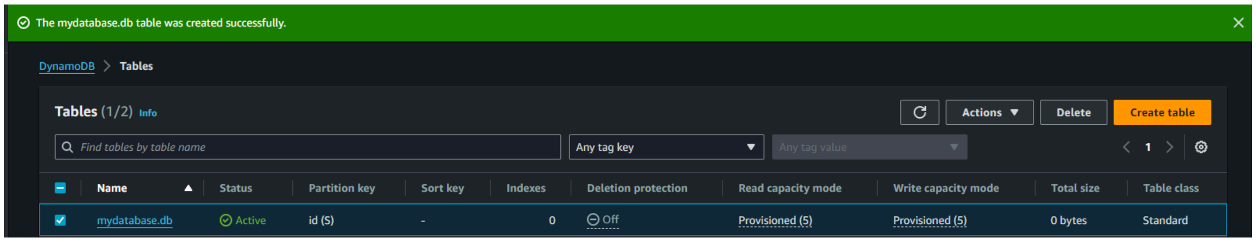
1 to 255 characters and case sensitive.

Table settings

[✓ Default settings] The fastest way to create your table. You can modify these settings now or after your table has been created.

Customize settings: Use these advanced features to make DynamoDB work better for your needs.

8. Check the table.



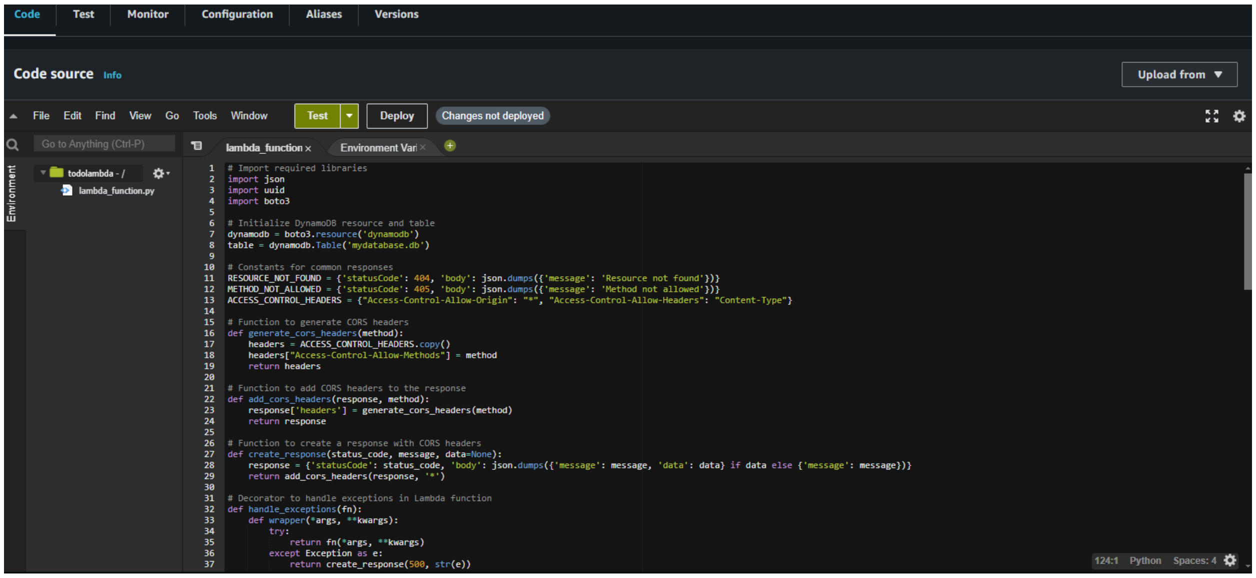
The mydatabase.db table was created successfully.

DynamoDB > Tables

Tables (1/2) Info

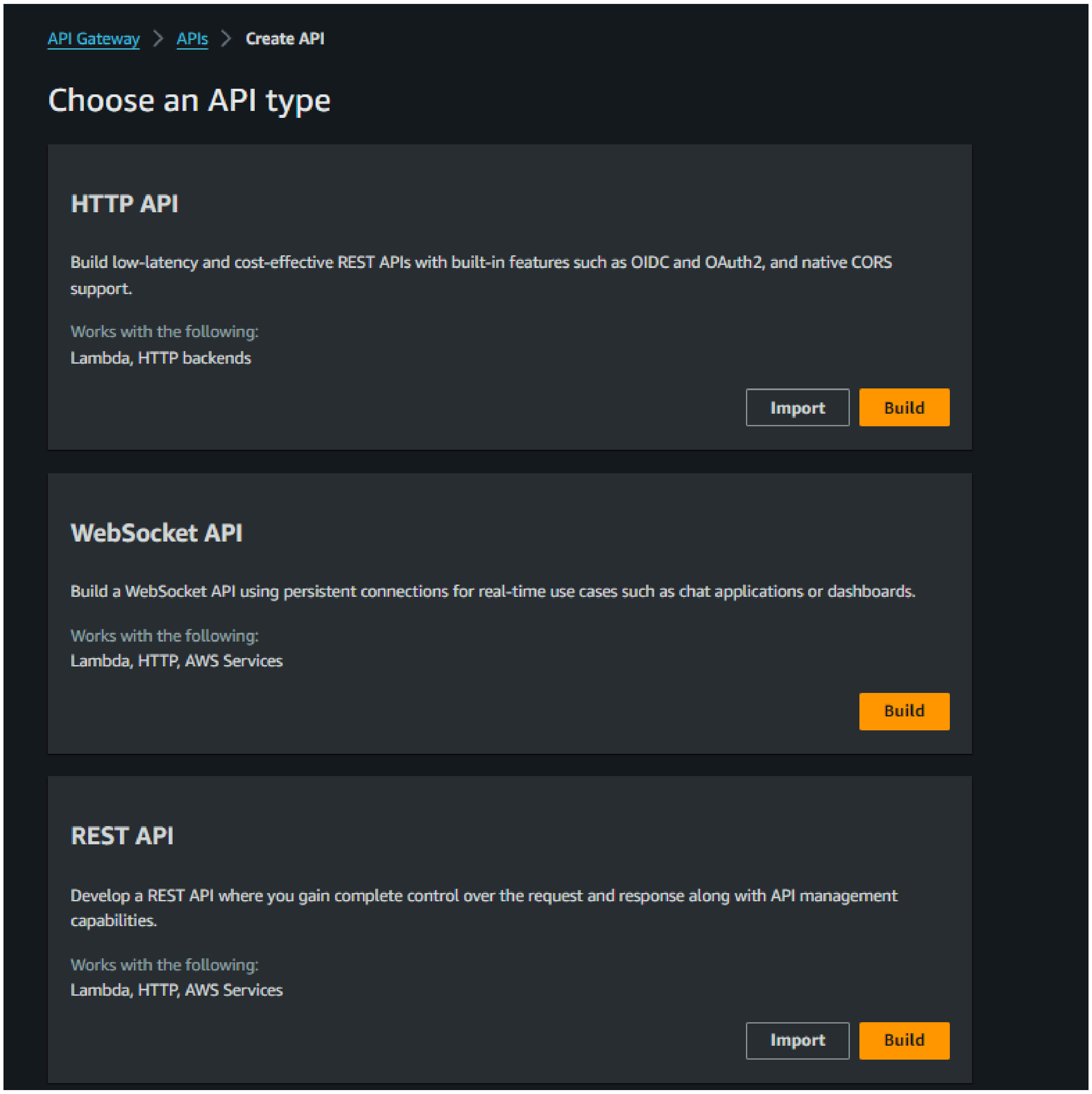
	Name	Status	Partition key	Sort key	Indexes	Deletion protection	Read capacity mode	Write capacity mode	Total size	Table class
✓	mydatabase.db	Active	id (S)	-	0	Off	Provisioned (5)	Provisioned (5)	0 bytes	Standard

9. Write the code in the code source.



```python
# Import required libraries
import json
import uuid
import boto3

# Initialize DynamoDB resource and table
dynamodb = boto3.resource('dynamodb')
table = dynamodb.Table('mydatabase.db')

# Constants for common responses
RESOURCE_NOT_FOUND = {'statusCode': 404, 'body': json.dumps({'message': 'Resource not found'})}
METHOD_NOT_ALLOWED = {'statusCode': 405, 'body': json.dumps({'message': 'Method not allowed'})}
ACCESS_CONTROL_HEADERS = {"Access-Control-Allow-Origin": "*", "Access-Control-Allow-Headers": "Content-Type"}

# Function to generate CORS headers
def generate_cors_headers(method):
    headers = ACCESS_CONTROL_HEADERS.copy()
    headers["Access-Control-Allow-Methods"] = method
    return headers

# Function to add CORS headers to the response
def add_cors_headers(response, method):
    response['headers'] = generate_cors_headers(method)
    return response

# Function to create a response with CORS headers
def create_response(status_code, message, data=None):
    response = {'statusCode': status_code, 'body': json.dumps({'message': message, 'data': data} if data else {'message': message})}
    return add_cors_headers(response, '*')

# Decorator to handle exceptions in Lambda function
def handle_exceptions(fn):
    def wrapper(*args, **kwargs):
        try:
            return fn(*args, **kwargs)
        except Exception as e:
            return create_response(500, str(e))
```

10. Go to choose an API gateway and choose REST API.


API Gateway > APIs > Create API

Choose an API type

HTTP API

Build low-latency and cost-effective REST APIs with built-in features such as OIDC and OAuth2, and native CORS support.

Works with the following:
Lambda, HTTP backends

WebSocket API

Build a WebSocket API using persistent connections for real-time use cases such as chat applications or dashboards.

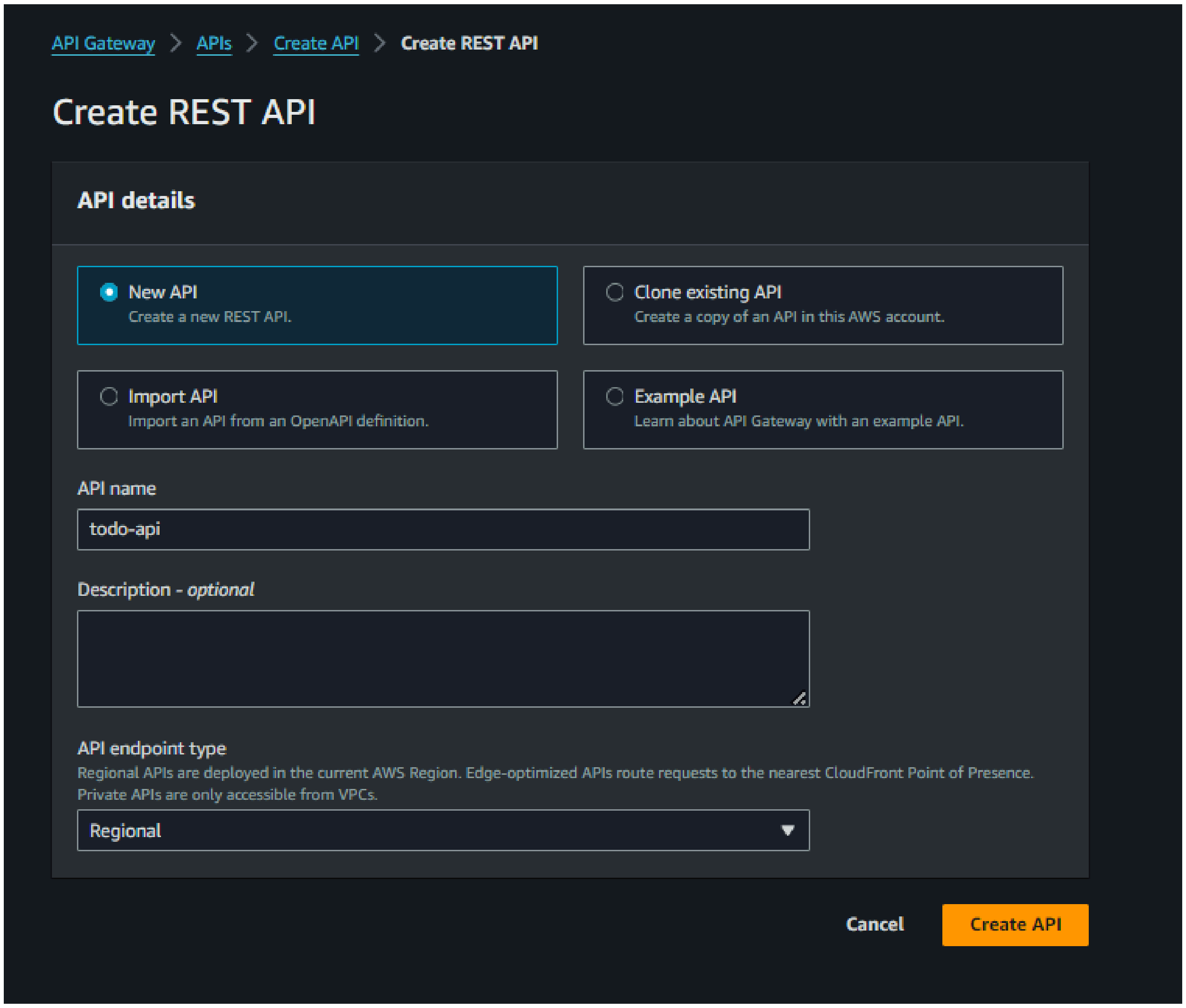
Works with the following:
Lambda, HTTP, AWS Services

REST API

Develop a REST API where you gain complete control over the request and response along with API management capabilities.

Works with the following:
Lambda, HTTP, AWS Services

11. Create the API



API Gateway > APIs > Create API > Create REST API

Create REST API

API details

[✓ New API — Create a new REST API.]
Clone existing API — Create a copy of an API in this AWS account.
Import API — Import an API from an OpenAPI definition.
Example API — Learn about API Gateway with an example API.

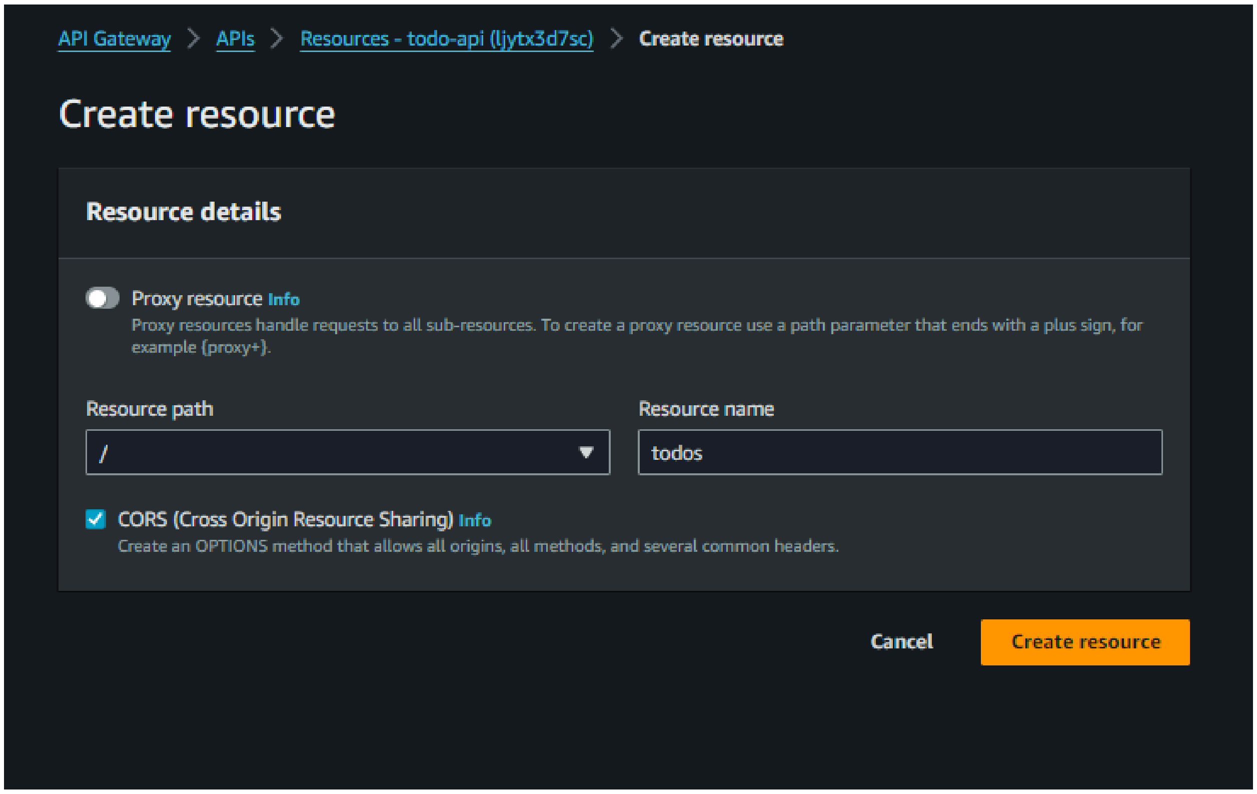
API name: todo-api

Description - *optional*

API endpoint type
Regional APIs are deployed in the current AWS Region. Edge-optimized APIs route requests to the nearest CloudFront Point of Presence. Private APIs are only accessible from VPCs.

Regional

12. Choose on create resource and create the resource.



API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create resource

Create resource

Resource details

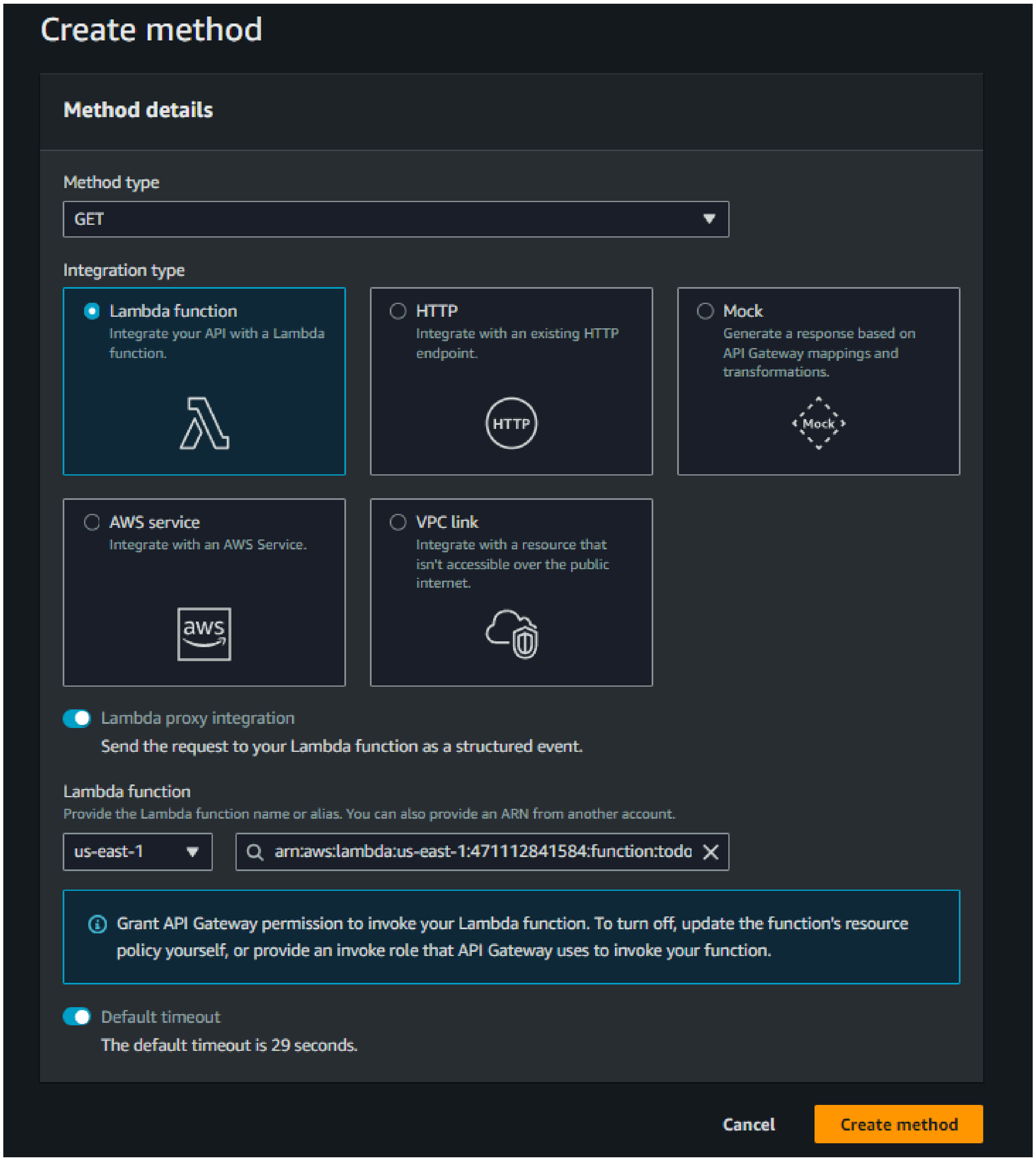
Proxy resource Info
Proxy resources handle requests to all sub-resources. To create a proxy resource use a path parameter that ends with a plus sign, for example {proxy+}.

Resource path: /

Resource name: todos

[✓] CORS (Cross Origin Resource Sharing) Info
Create an OPTIONS method that allows all origins, all methods, and several common headers.

13. Go to create method and choose GET in method type and also choose the todo function.


Create method

Method details

Method type

GET

Integration type

- [✓ Lambda function] Integrate your API with a Lambda function.
- HTTP: Integrate with an existing HTTP endpoint.
- Mock: Generate a response based on API Gateway mappings and transformations.
- AWS service: Integrate with an AWS Service.
- VPC link: Integrate with a resource that isn't accessible over the public internet.

Lambda proxy integration: on
Send the request to your Lambda function as a structured event.

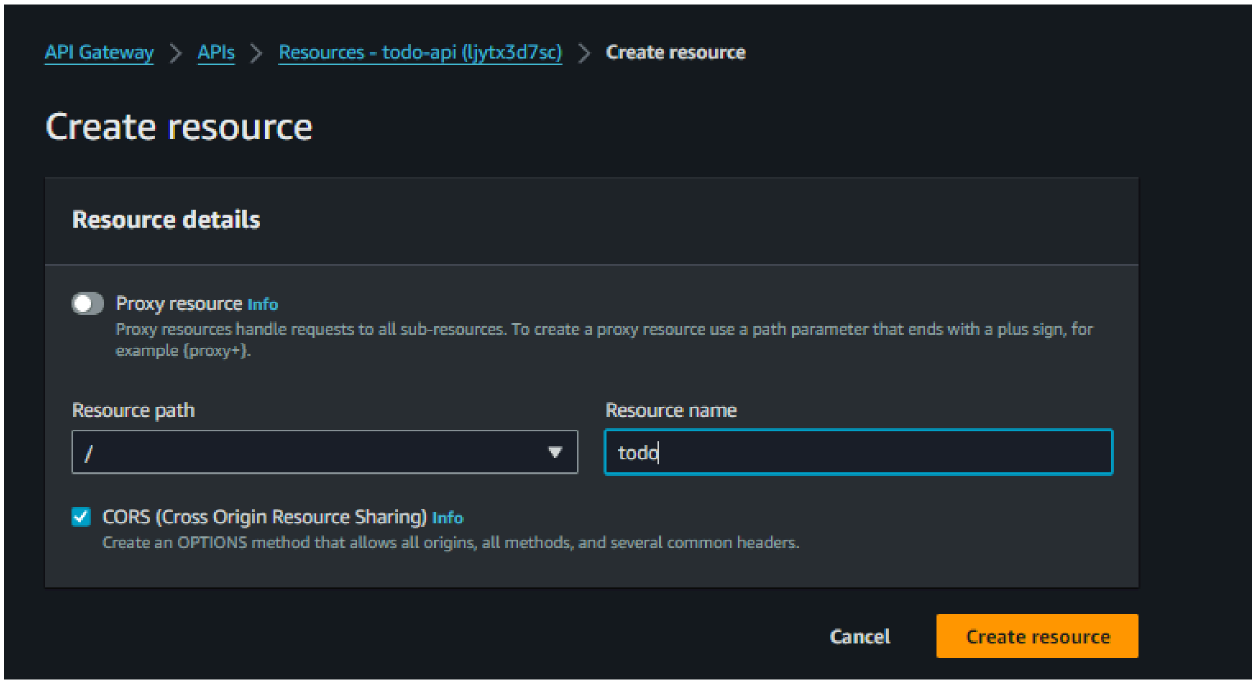
Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.

us-east-1 | arn:aws:lambda:us-east-1:471112841584:function:todo

Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.

Default timeout: on
The default timeout is 29 seconds.

14. Go to create resource and enter the resource name.



API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create resource

Create resource

Resource details

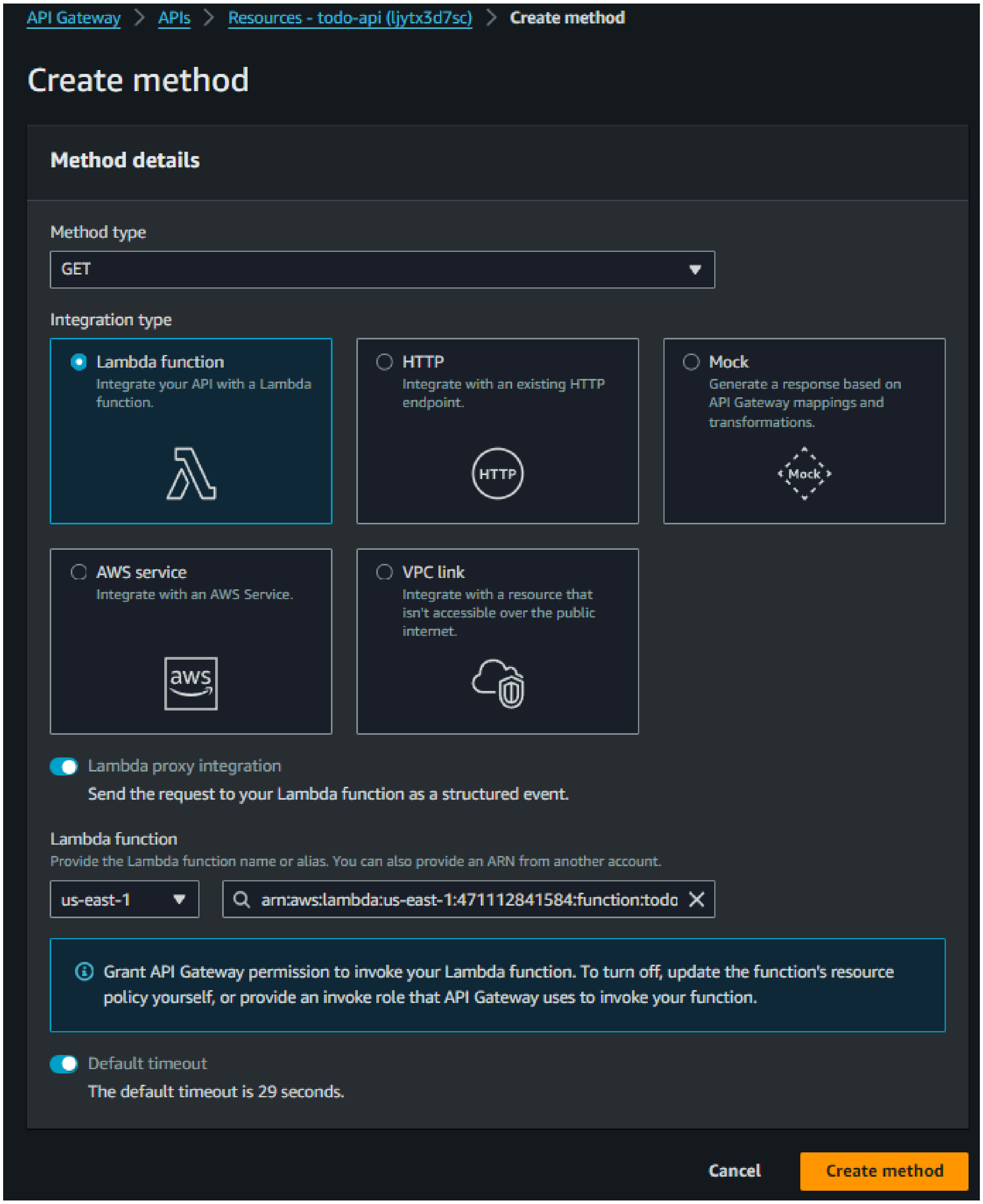
Proxy resource Info
Proxy resources handle requests to all sub-resources. To create a proxy resource use a path parameter that ends with a plus sign, for example (proxy+).

Resource path: /

Resource name: todo

[✓] CORS (Cross Origin Resource Sharing) Info
Create an OPTIONS method that allows all origins, all methods, and several common headers.

15. Create GET in todo.


API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create method

Create method

Method details

Method type

GET

Integration type

[✓ Lambda function] Integrate your API with a Lambda function.

HTTP Integrate with an existing HTTP endpoint.

Mock Generate a response based on API Gateway mappings and transformations.

AWS service Integrate with an AWS Service.

VPC link Integrate with a resource that isn't accessible over the public internet.

[✓ Lambda proxy integration]
Send the request to your Lambda function as a structured event.

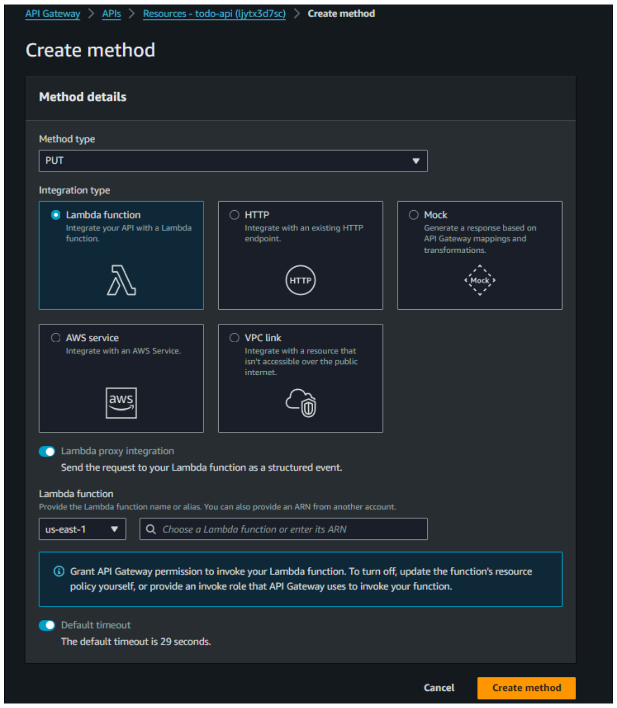
Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.

us-east-1 | arn:aws:lambda:us-east-1:471112841584:function:todo

Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.

[✓ Default timeout]
The default timeout is 29 seconds.

16. In the similar way create PUT, POST and DELETE.


API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create method

Create method

Method details

Method type: PUT

Integration type

- [✓ Lambda function] Integrate your API with a Lambda function.
- HTTP: Integrate with an existing HTTP endpoint.
- Mock: Generate a response based on API Gateway mappings and transformations.
- AWS service: Integrate with an AWS Service.
- VPC link: Integrate with a resource that isn't accessible over the public internet.

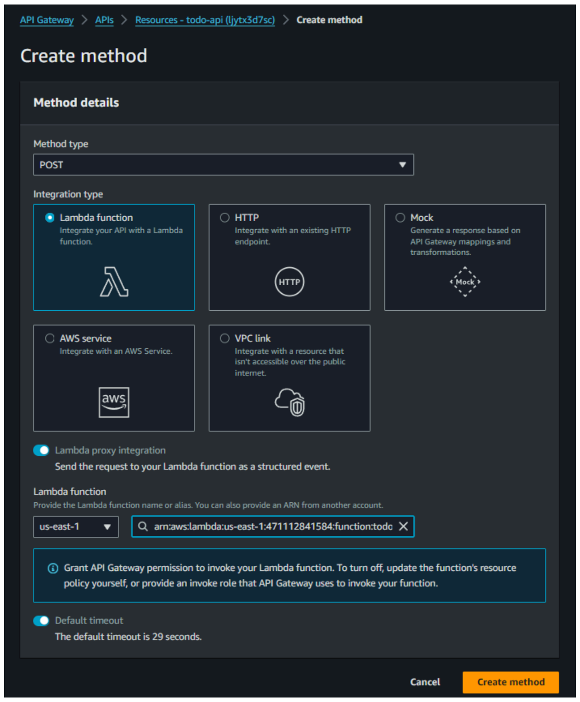
Lambda proxy integration
Send the request to your Lambda function as a structured event.

Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.
us-east-1 | Choose a Lambda function or enter its ARN

Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.

Default timeout
The default timeout is 29 seconds.

Cancel | Create method


API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create method

Create method

Method details

Method type: POST

Integration type

- [✓ Lambda function] Integrate your API with a Lambda function.
- HTTP: Integrate with an existing HTTP endpoint.
- Mock: Generate a response based on API Gateway mappings and transformations.
- AWS service: Integrate with an AWS Service.
- VPC link: Integrate with a resource that isn't accessible over the public internet.

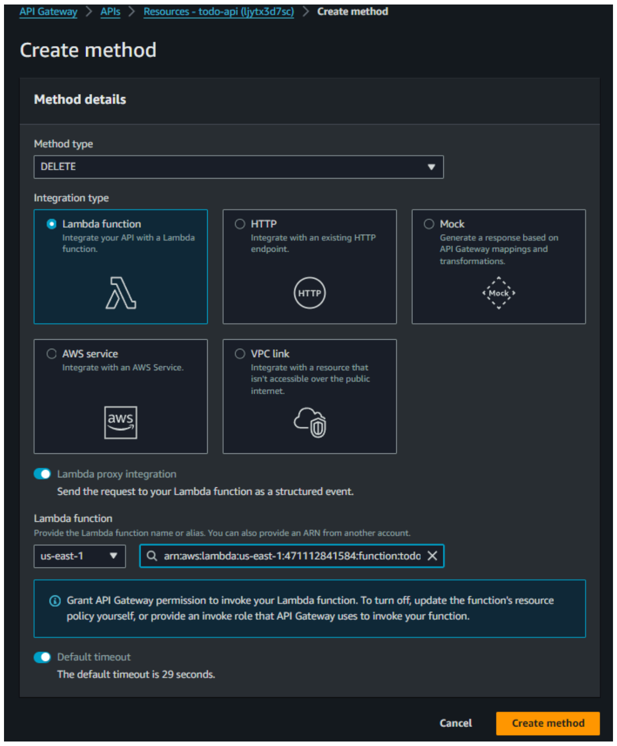
Lambda proxy integration
Send the request to your Lambda function as a structured event.

Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.
us-east-1 | arn:aws:lambda:us-east-1:471112841584:function:todo

Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.

Default timeout
The default timeout is 29 seconds.

Cancel | Create method


API Gateway > APIs > Resources - todo-api (ljytx3d7sc) > Create method

Create method

Method details

Method type: DELETE

Integration type

- [✓ Lambda function] Integrate your API with a Lambda function.
- HTTP: Integrate with an existing HTTP endpoint.
- Mock: Generate a response based on API Gateway mappings and transformations.
- AWS service: Integrate with an AWS Service.
- VPC link: Integrate with a resource that isn't accessible over the public internet.

Lambda proxy integration
Send the request to your Lambda function as a structured event.

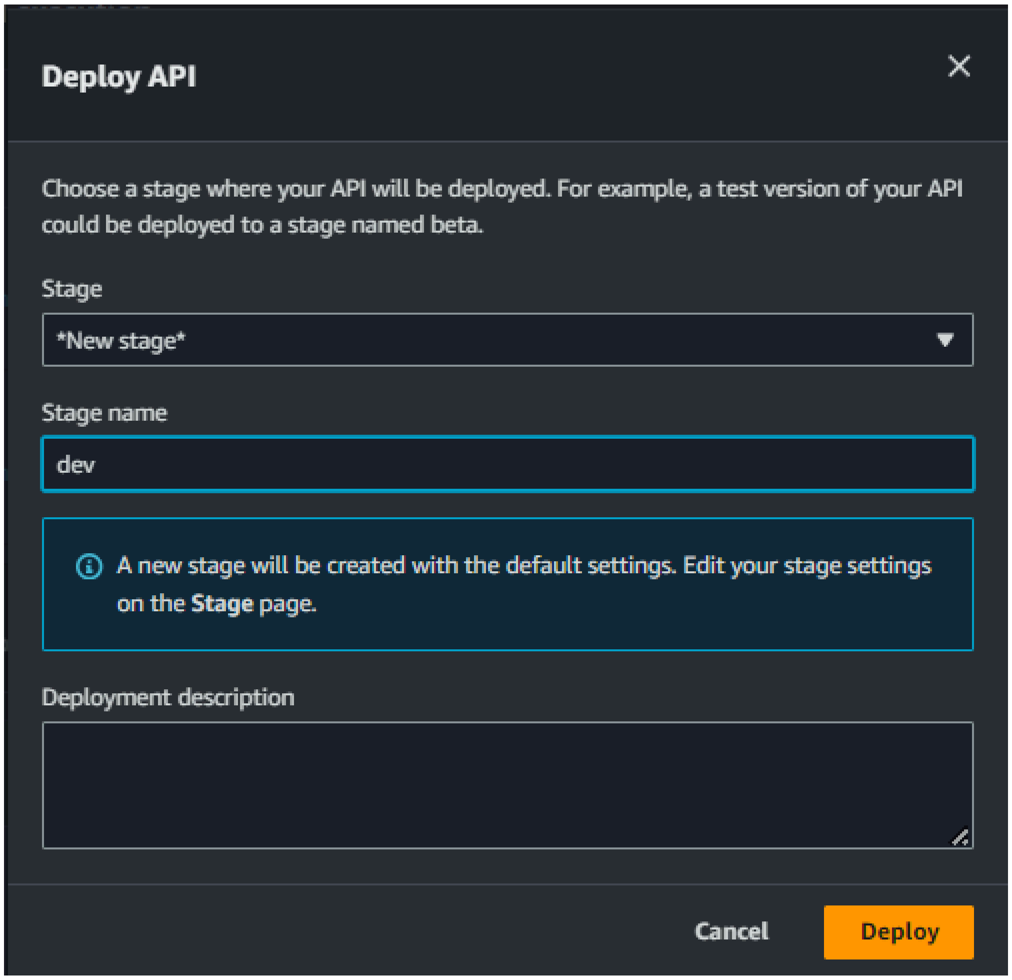
Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.
us-east-1 | arn:aws:lambda:us-east-1:471112841584:function:todo

Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.

Default timeout
The default timeout is 29 seconds.

Cancel | Create method

17. Deploy the API



Deploy API

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

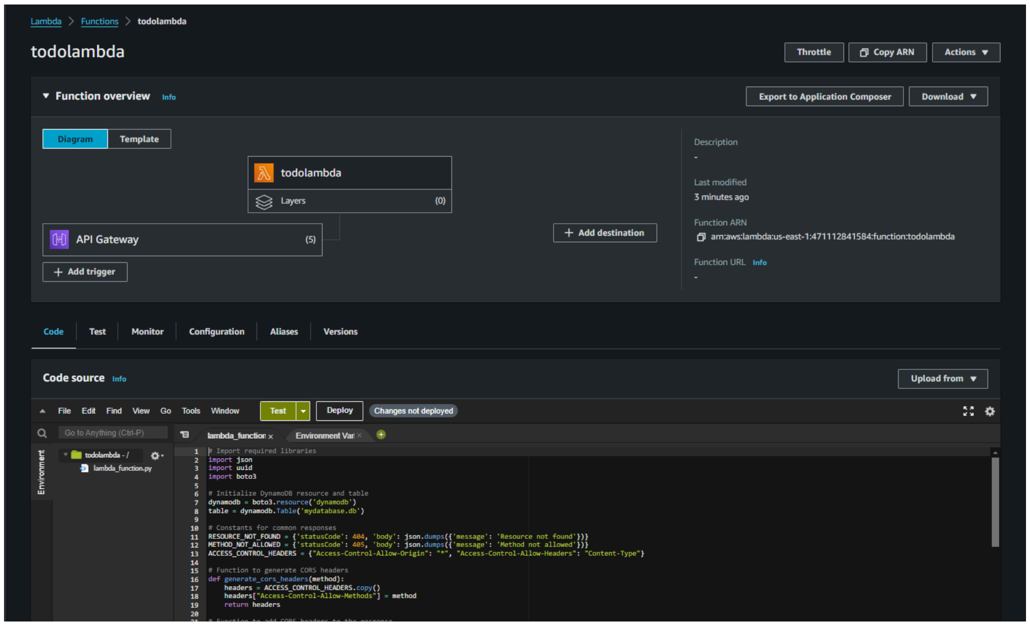
Stage: *New stage*

Stage name: dev

A new stage will be created with the default settings. Edit your stage settings on the **Stage** page.

Deployment description:

18. Go to the lambda function.



todolambda

Description: -

Last modified: 3 minutes ago

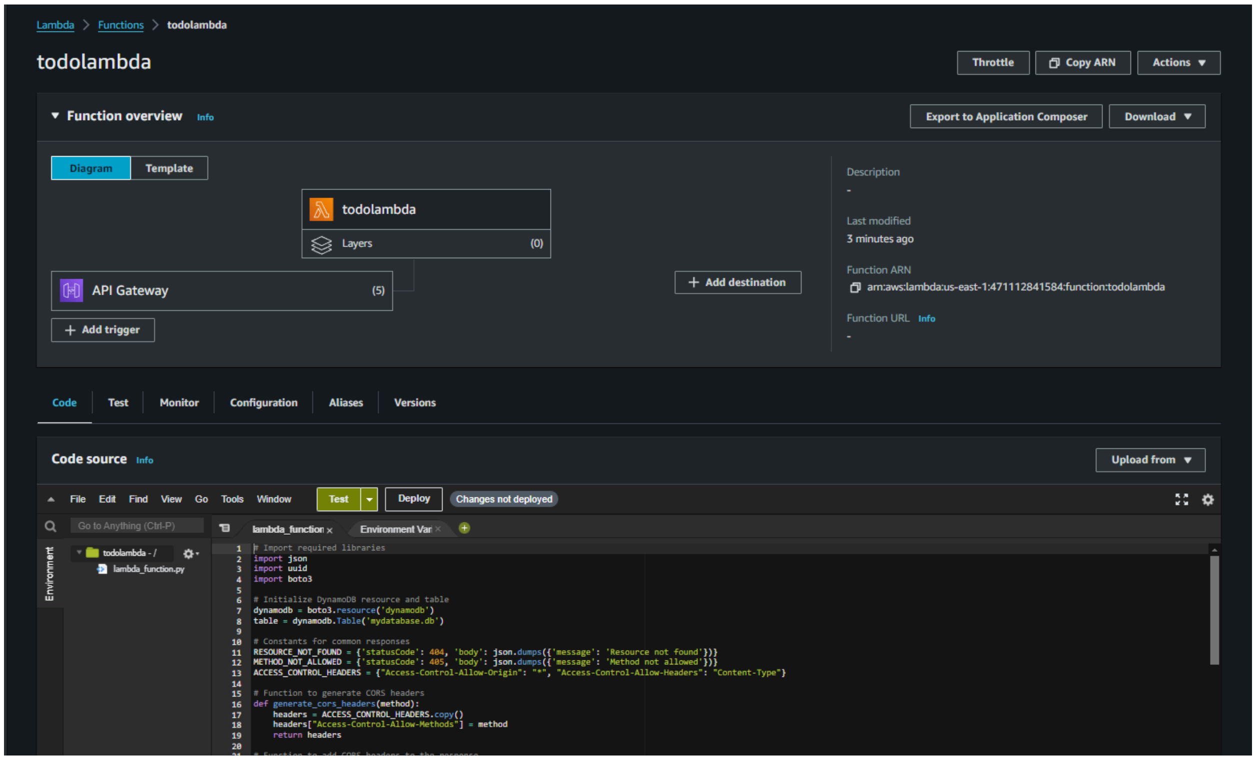
Function ARN: arn:aws:lambda:us-east-1:471112841584:function:todolambda

Function URL: -

Code source

```
# Import required libraries
import json
import uuid
import boto3

# Initialize DynamoDB resource and table
dynamodb = boto3.resource('dynamodb')
table = dynamodb.Table('mydatabase.db')

# Constants for common responses
RESOURCE_NOT_FOUND = {'statusCode': 404, 'body': json.dumps({'message': 'Resource not found'})}
METHOD_NOT_ALLOWED = {'statusCode': 405, 'body': json.dumps({'message': 'Method not allowed'})}
ACCESS_CONTROL_HEADERS = {"Access-Control-Allow-Origin": "*", "Access-Control-Allow-Headers": "Content-Type"}

# Function to generate CORS headers
def generate_cors_headers(method):
    headers = ACCESS_CONTROL_HEADERS.copy()
    headers["Access-Control-Allow-Methods"] = method
    return headers

```

19. Click on deploy



Description
-

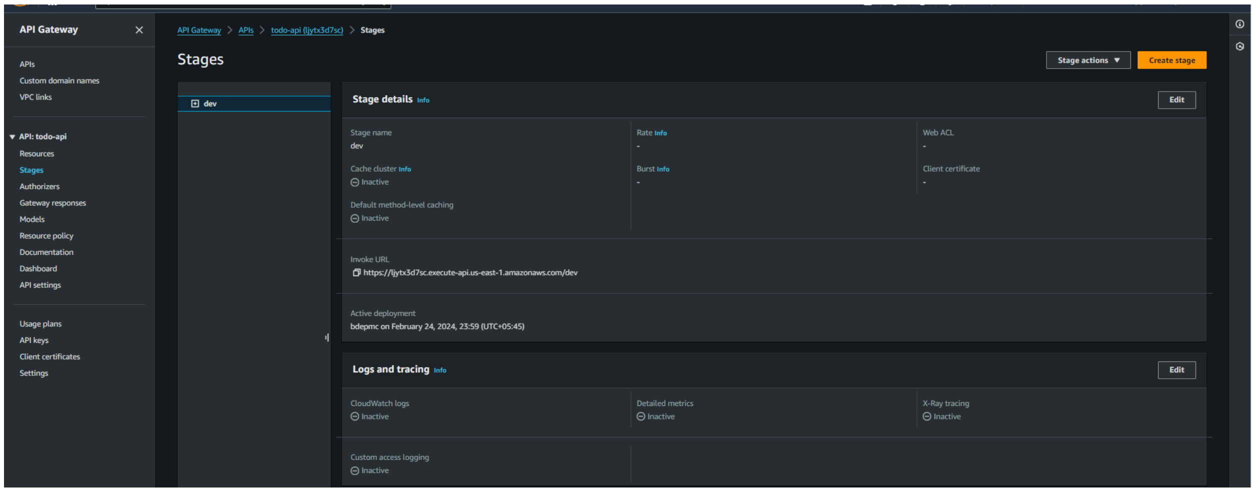
Last modified
3 minutes ago

Function ARN
arn:aws:lambda:us-east-1:471112841584:function:todolambda

Function URL Info
-

```
# Import required libraries
import json
import uuid
import boto3

# Initialize DynamoDB resource and table
dynamodb = boto3.resource('dynamodb')
table = dynamodb.Table('mydatabase.db')

# Constants for common responses
RESOURCE_NOT_FOUND = {'statusCode': 404, 'body': json.dumps({'message': 'Resource not found'})}
METHOD_NOT_ALLOWED = {'statusCode': 405, 'body': json.dumps({'message': 'Method not allowed'})}
ACCESS_CONTROL_HEADERS = {"Access-Control-Allow-Origin": "*", "Access-Control-Allow-Headers": "Content-Type"}

# Function to generate CORS headers
def generate_cors_headers(method):
    headers = ACCESS_CONTROL_HEADERS.copy()
    headers["Access-Control-Allow-Methods"] = method
    return headers

```

20. Go to API gateway and select stages.



Stage name
dev

Rate Info
-

Web ACL
-

Cache cluster Info
Inactive

Burst Info
-

Client certificate
-

Default method-level caching
Inactive

Invoke URL
https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev

Active deployment
bdepmc on February 24, 2024, 23:59 (UTC+05:45)

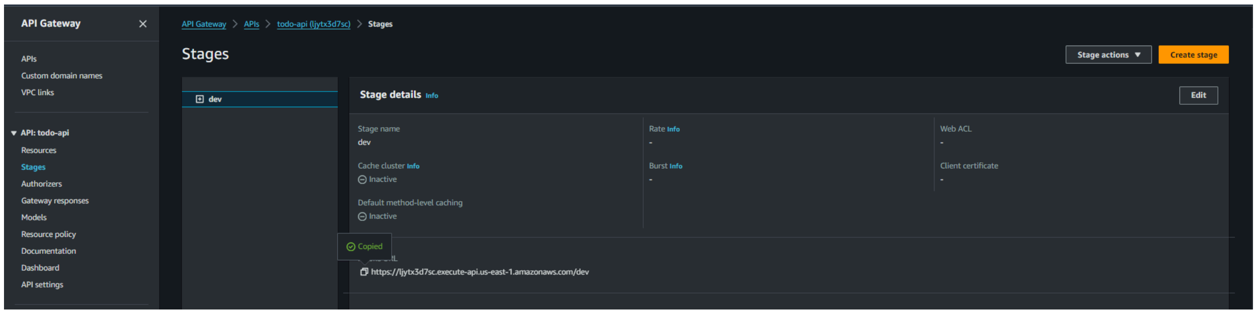
CloudWatch logs
Inactive

Detailed metrics
Inactive

X-Ray tracing
Inactive

Custom access logging
Inactive

21. Copy the invoke URL.

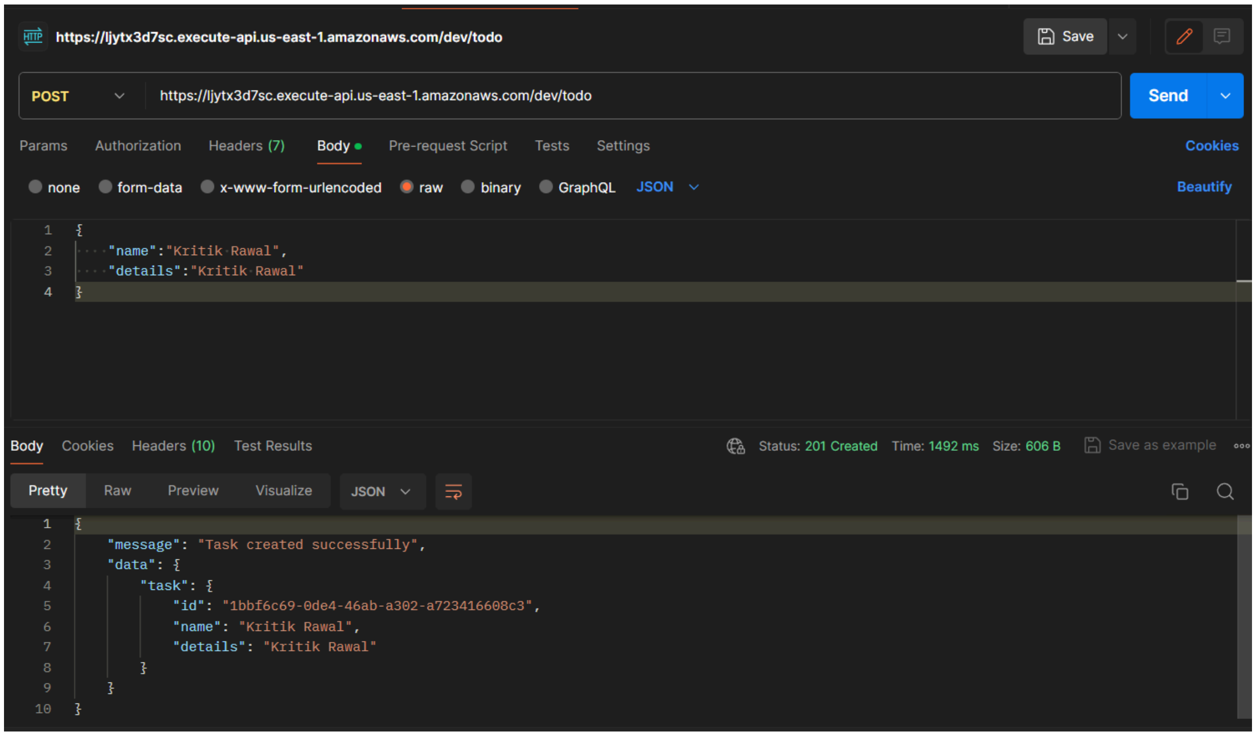


Stage details Info

Stage name: dev
Rate Info: -
Web ACL: -
Cache cluster Info: Inactive
Burst Info: -
Client certificate: -
Default method-level caching: Inactive

https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev

22. Go to postman and check.



https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo

POST https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo

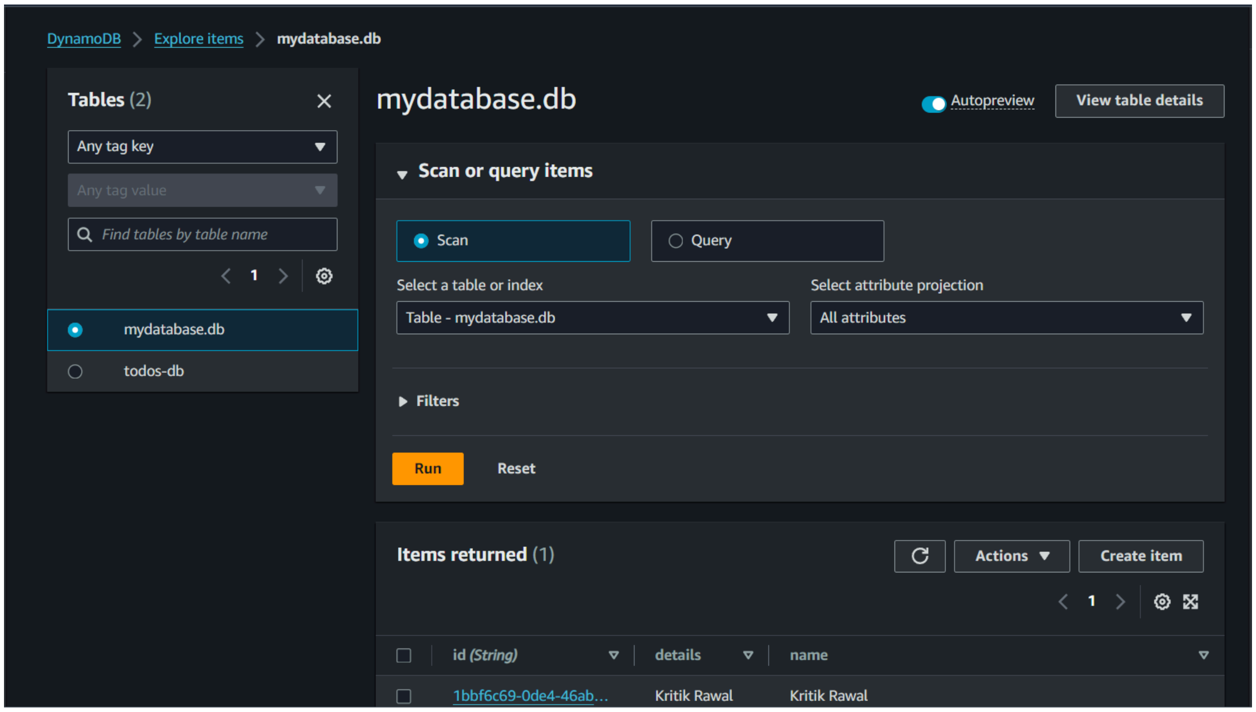
Send

```
{
    "name":"Kritik Rawal",
    "details":"Kritik Rawal"
}
```

Status: 201 Created Time: 1492 ms Size: 606 B

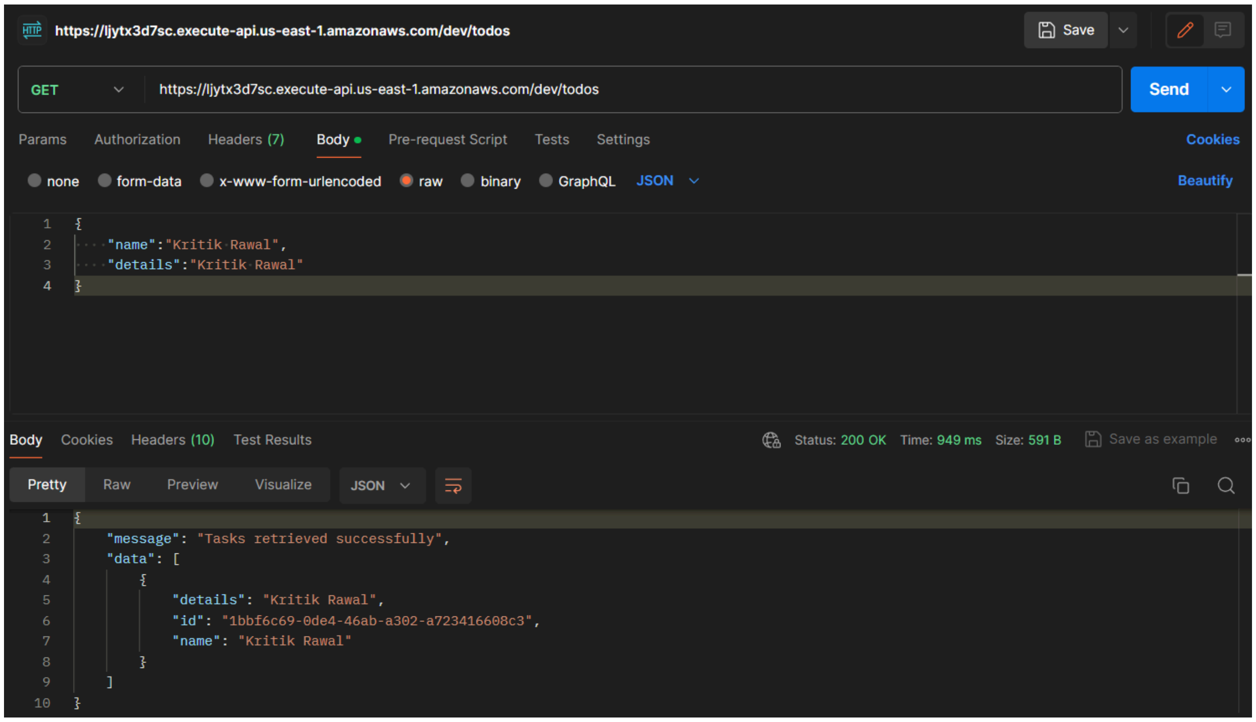
```
{
    "message": "Task created successfully",
    "data": {
        "task": {
            "id": "1bbf6c69-0de4-46ab-a302-a723416608c3",
            "name": "Kritik Rawal",
            "details": "Kritik Rawal"
        }
    }
}
```

23. In the dynamodb check the items returned.



id (String)	details	name
1bbf6c69-0de4-46ab...	Kritik Rawal	Kritik Rawal

24. Send a GET request.



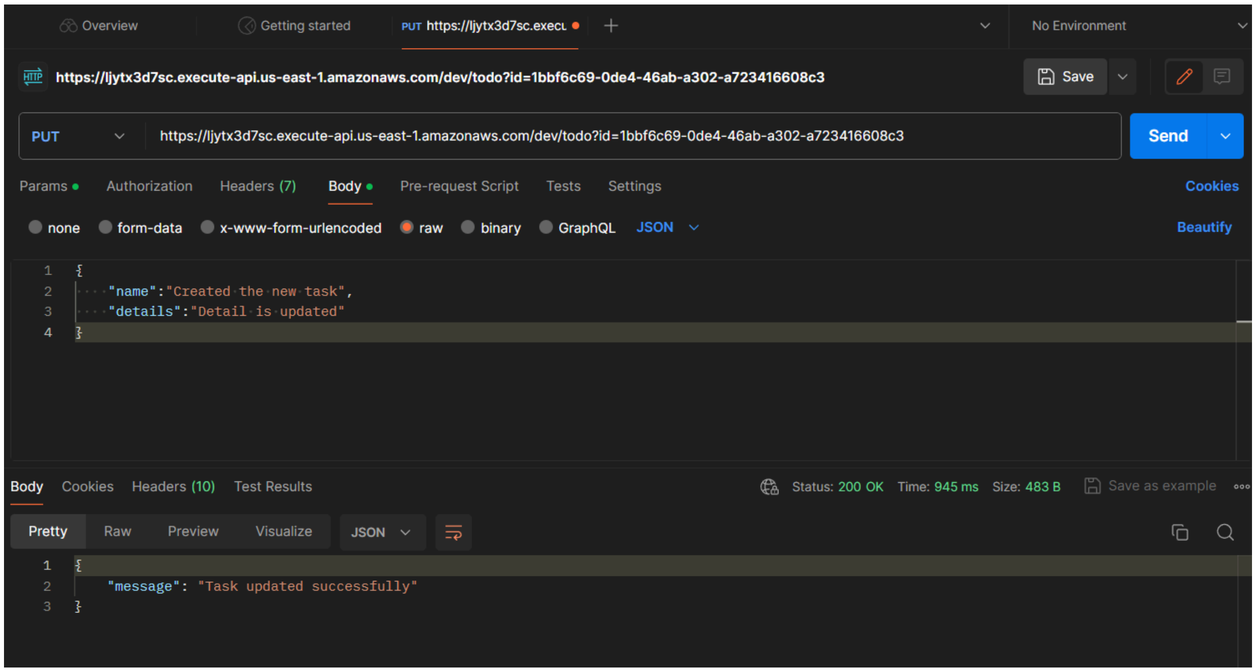
```
GET https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todos
```

```
{
    "name":"Kritik Rawal",
    "details":"Kritik Rawal"
}
```

Status: 200 OK Time: 949 ms Size: 591 B

```
{
    "message": "Tasks retrieved successfully",
    "data": [
        {
            "details": "Kritik Rawal",
            "id": "1bbf6c69-0de4-46ab-a302-a723416608c3",
            "name": "Kritik Rawal"
        }
    ]
}
```

25. Send a POST request.



https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo?id=1bbf6c69-0de4-46ab-a302-a723416608c3

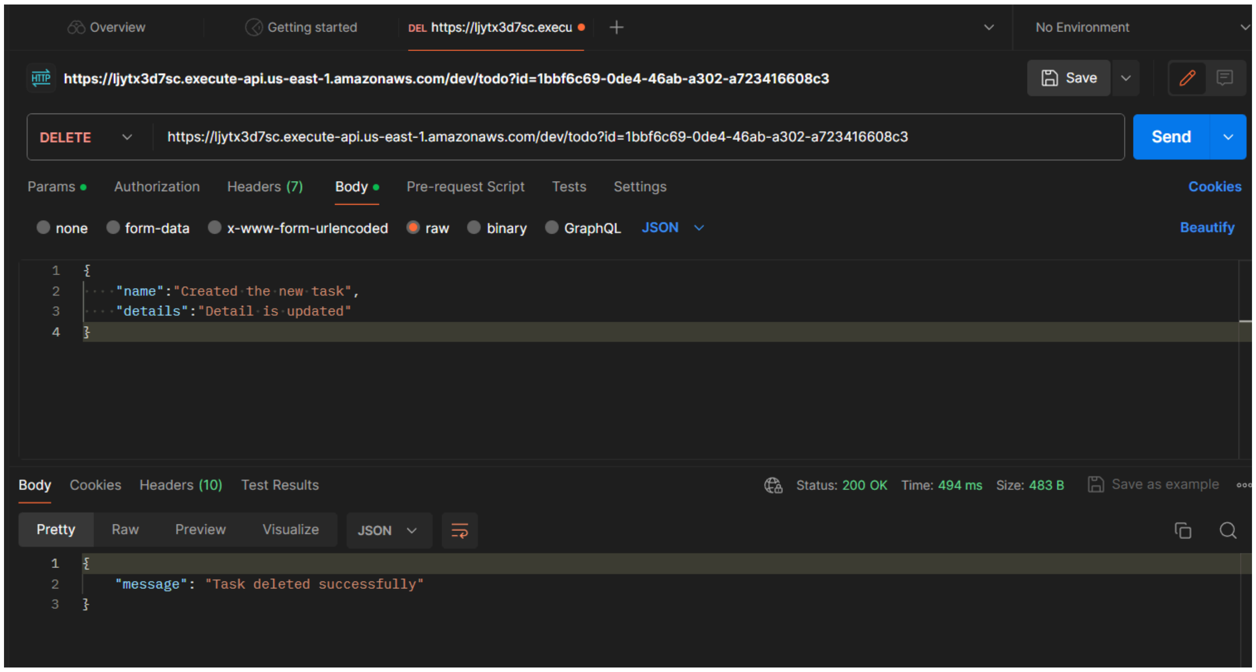
PUT https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo?id=1bbf6c69-0de4-46ab-a302-a723416608c3

```
{
    "name":"Created the new task",
    "details":"Detail is updated"
}
```

Status: 200 OK Time: 945 ms Size: 483 B

```
{
    "message": "Task updated successfully"
}
```

26. DELETE the request.



https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo?id=1bbf6c69-0de4-46ab-a302-a723416608c3

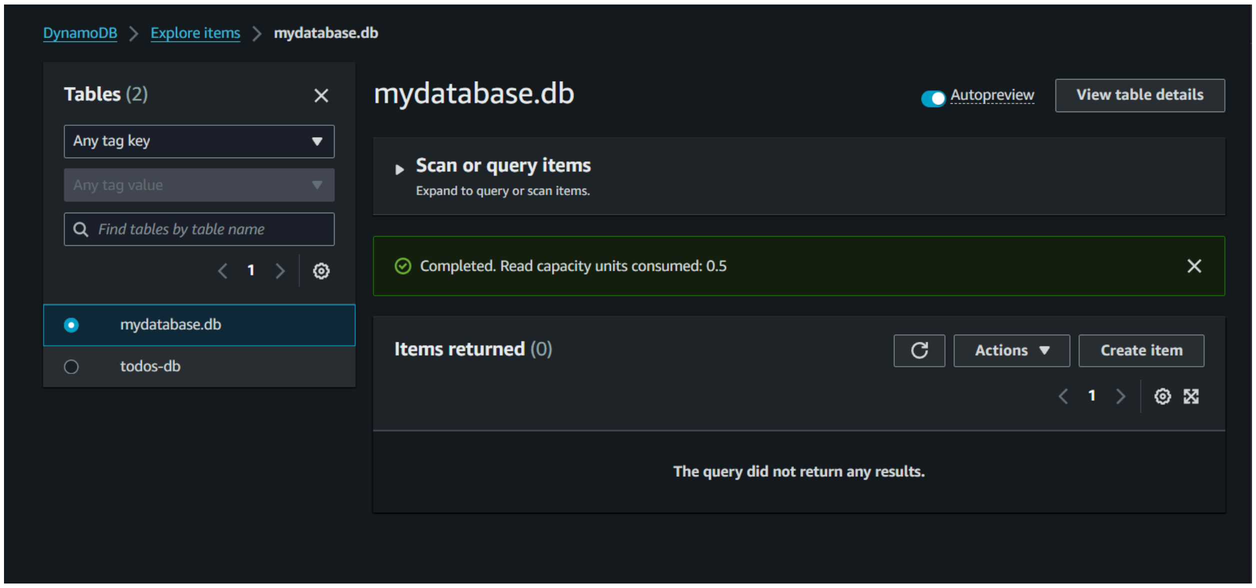
DELETE https://ljytx3d7sc.execute-api.us-east-1.amazonaws.com/dev/todo?id=1bbf6c69-0de4-46ab-a302-a723416608c3

```
{
    "name":"Created the new task",
    "details":"Detail is updated"
}
```

Status: 200 OK Time: 494 ms Size: 483 B

```
{
    "message": "Task deleted successfully"
}
```

27. Refresh the page.



DynamoDB > Explore items > mydatabase.db

Tables (2)

Any tag key

Any tag value

Find tables by table name

● mydatabase.db

○ todos-db

mydatabase.db

Autopreview

View table details

Scan or query items

Expand to query or scan items.

Completed. Read capacity units consumed: 0.5

Items returned (0)

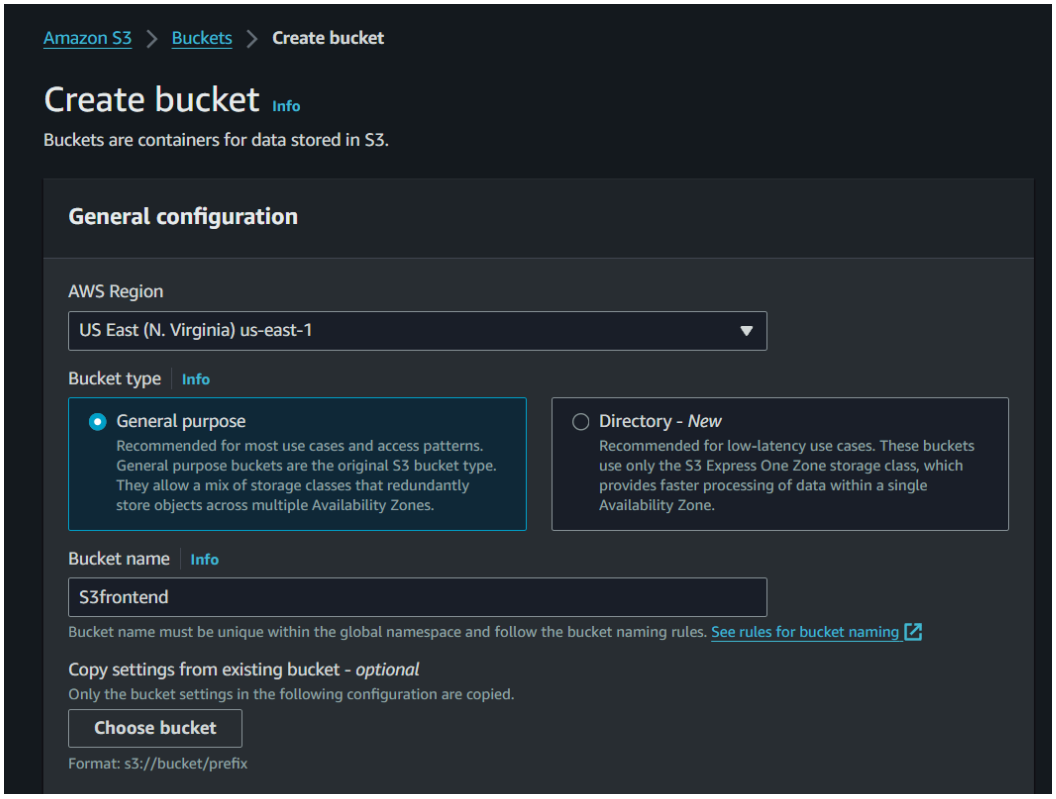
Actions

Create item

The query did not return any results.

TASK 1:

1. Create a new S3 bucket.


Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

US East (N. Virginia) us-east-1

Bucket type Info

[✓ General purpose] Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

Directory - *New* Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info

S3frontend

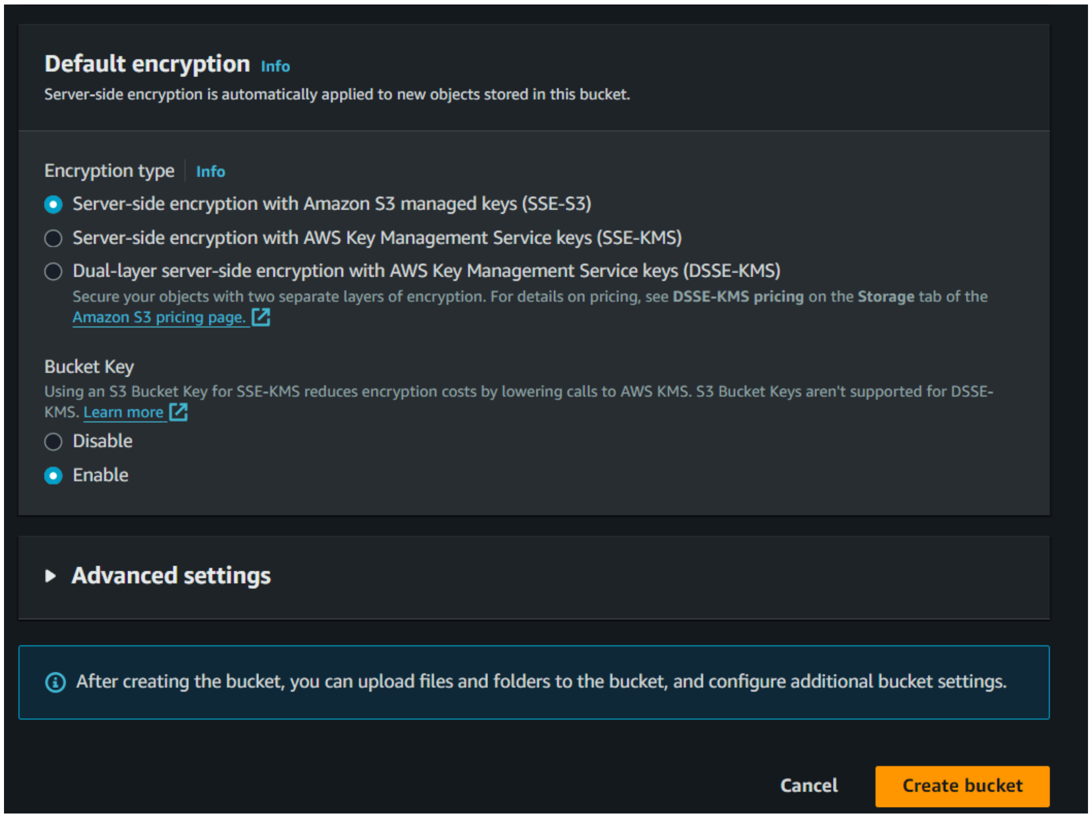
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix


Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type Info

[✓ Server-side encryption with Amazon S3 managed keys (SSE-S3)]

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

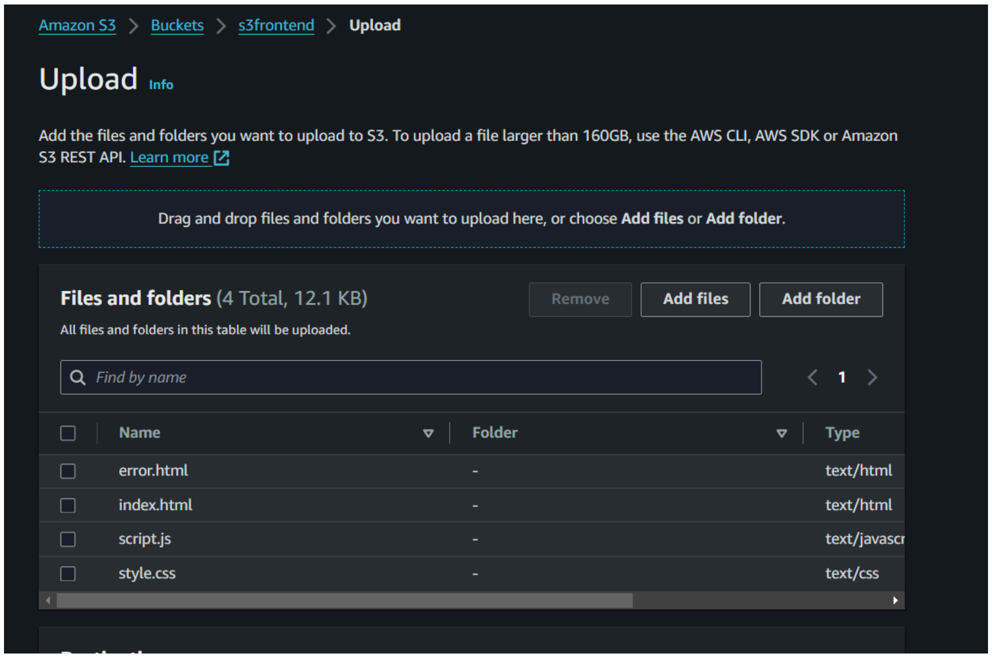
Disable [✓ Enable]

Advanced settings

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel Create bucket

2. Upload the files inside the bucket.



Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. Learn more

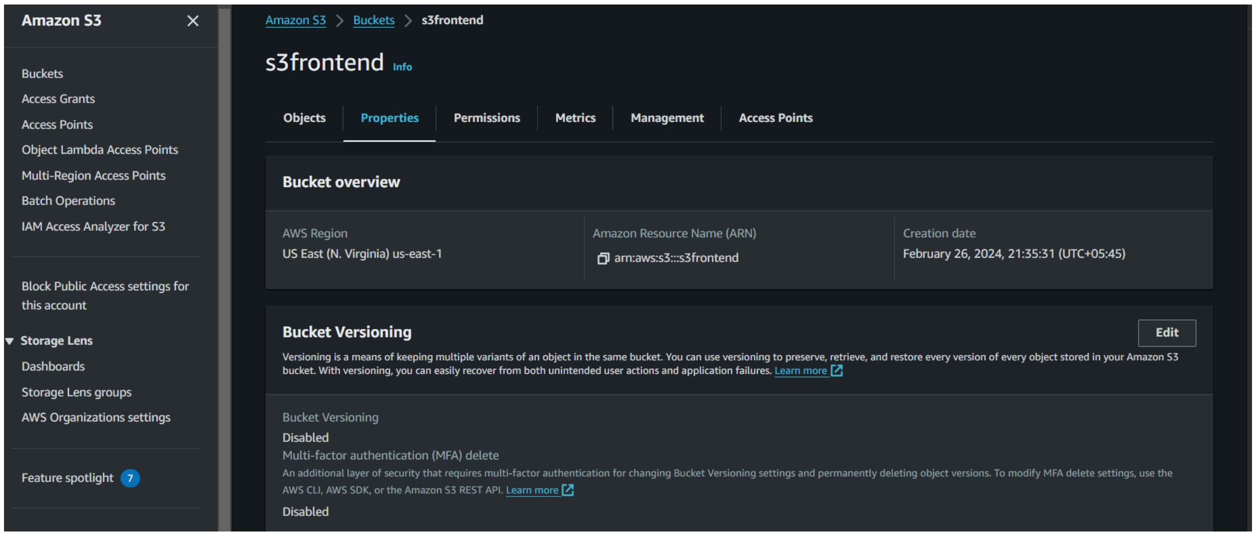
Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (4 Total, 12.1 KB)

All files and folders in this table will be uploaded.

Name	Folder	Type
error.html	-	text/html
index.html	-	text/html
script.js	-	text/javascr
style.css	-	text/css

3. Go to properties



Bucket overview

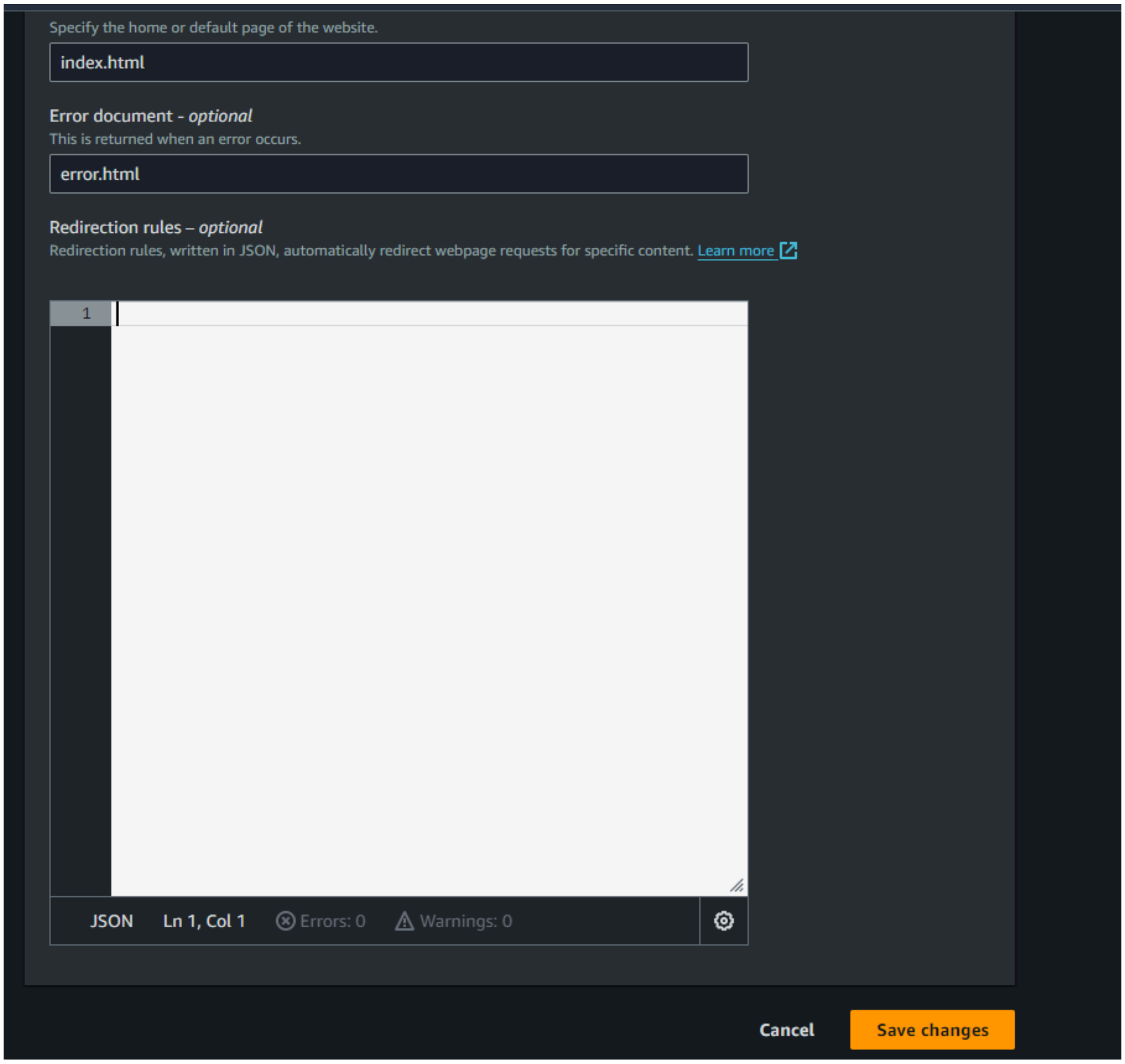
AWS Region	Amazon Resource Name (ARN)	Creation date
US East (N. Virginia) us-east-1	arn:aws:s3:::s3frontend	February 26, 2024, 21:35:31 (UTC+05:45)

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. Learn more

Bucket Versioning
Disabled
Multi-factor authentication (MFA) delete
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. Learn more
Disabled

4. Add the file name.



Specify the home or default page of the website.

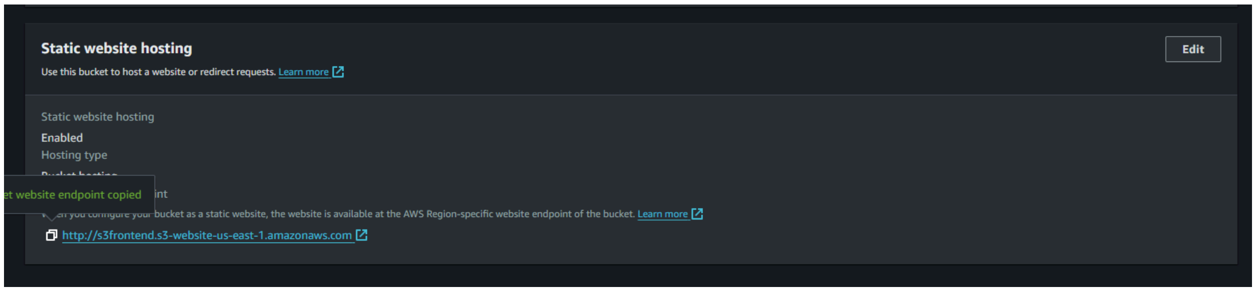
index.html

Error document - *optional*
This is returned when an error occurs.

error.html

Redirection rules – *optional*
Redirection rules, written in JSON, automatically redirect webpage requests for specific content. Learn more

5. Copy the name of URL.



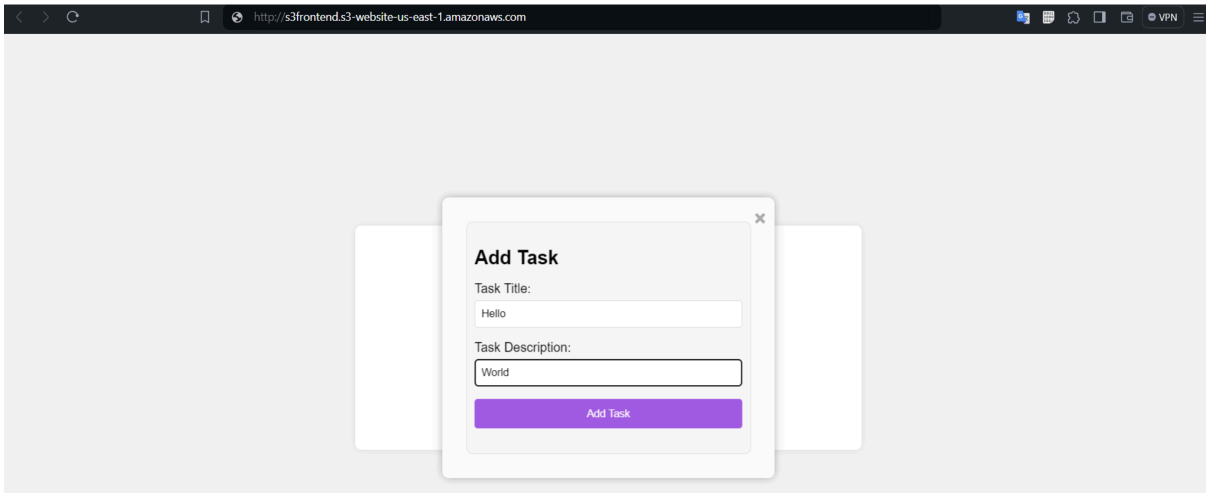
Static website hosting
Use this bucket to host a website or redirect requests. Learn more

Static website hosting
Enabled
Hosting type
Bucket hosting
…int
…en you configure your bucket as a static website, the website is available at the AWS Region-specific website endpoint of the bucket. Learn more

http://s3frontend.s3-website-us-east-1.amazonaws.com

6. Perform the CRUD operations.

A. Adding the task

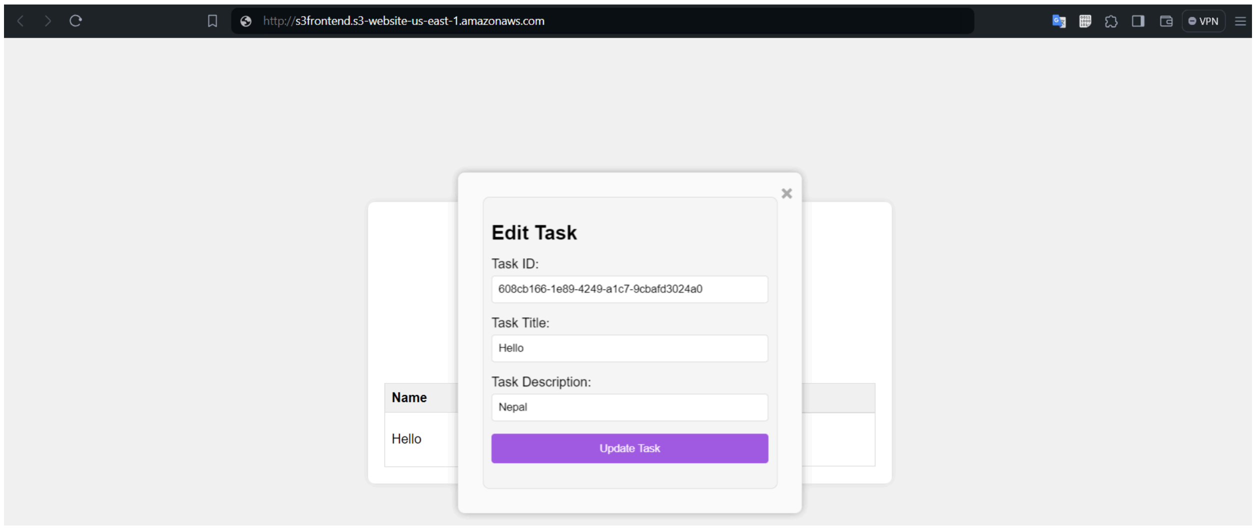

http://s3frontend.s3-website-us-east-1.amazonaws.com

Add Task

Task Title: Hello

Task Description: World

B. Editing the task


http://s3frontend.s3-website-us-east-1.amazonaws.com

Edit Task

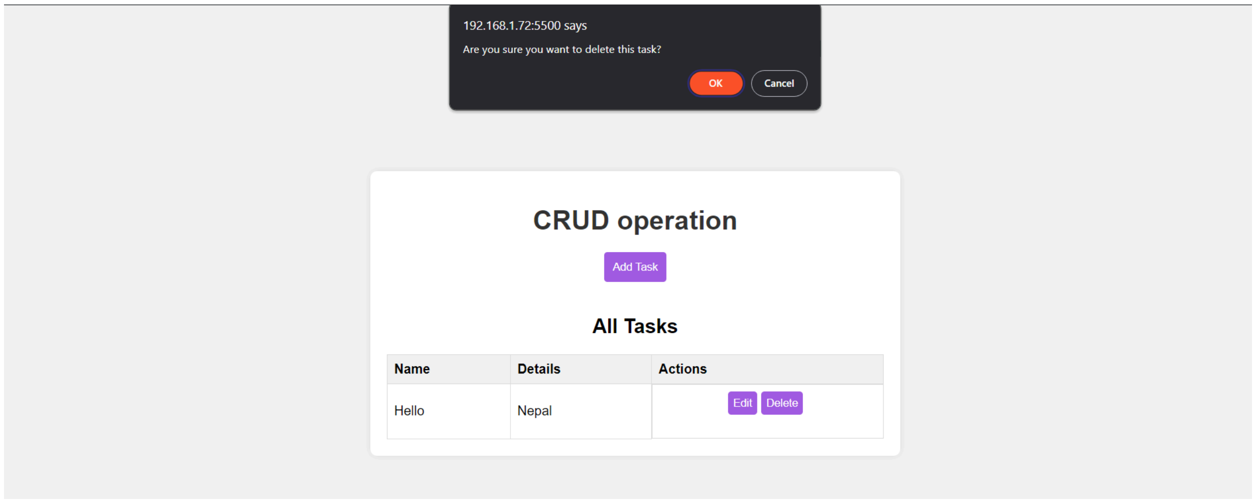
Task ID: 608cb166-1e89-4249-a1c7-9cbafd3024a0

Task Title: Hello

Task Description: Nepal

Name
Hello

C. Deleting the task.



192.168.1.72:5500 says

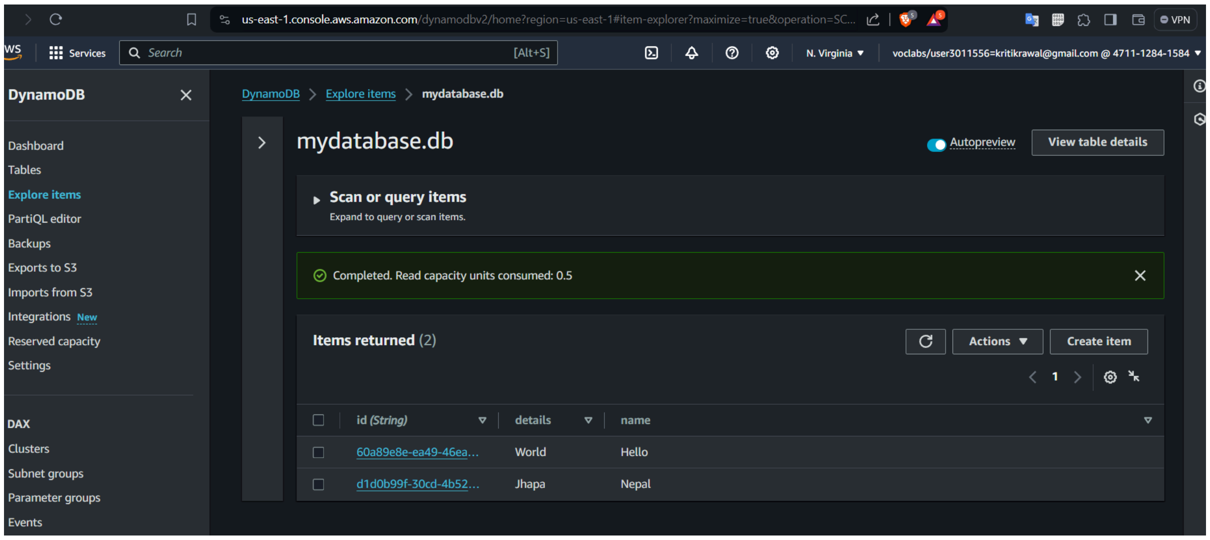
Are you sure you want to delete this task?

CRUD operation

All Tasks

Name	Details	Actions
Hello	Nepal	Edit Delete

7. Check the details in the DynamoDB



us-east-1.console.aws.amazon.com/dynamodbv2/home?region=us-east-1#item-explorer?maximize=true&operation=SC...

mydatabase.db

Scan or query items

Expand to query or scan items.

Completed. Read capacity units consumed: 0.5

Items returned (2)

id (String)	details	name
60a89e8e-ea49-46ea...	World	Hello
d1d0b99f-30cd-4b52...	Jhapa	Nepal