

Deploy web application with Lambda, ApiGateway and DynamoDB | AWS Serverless Services

Create IAM Role.

PermissionsTrust relationshipsTagsLast AccessedRevoke sessions

Permissions policies (2) Info

↺

Simulate ↗

Remove

Add permissions ▼



You can attach up to 10 managed policies.

Filter by Type

🔍 Search

All types ▼

< 1 > ⚙️

<input type="checkbox"/>	Policy name ↗	Type	Attached entities
<input type="checkbox"/>	 AmazonDynamoDBFullAccess	AWS managed	1
<input type="checkbox"/>	 AWSLambdaBasicExecution...	AWS managed	1


IAM > Roles > Lambda-role

Lambda-role Info

Delete

Allows Lambda functions to call AWS services on your behalf.

Summary Edit

Creation date	ARN
October 18, 2024, 16:45 (UTC+05:30)	 arn:aws:iam::637423206813:role/Lambda-role
Last activity	Maximum session duration
-	1 hour

Create lambda Function:

[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](#)

▼ 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

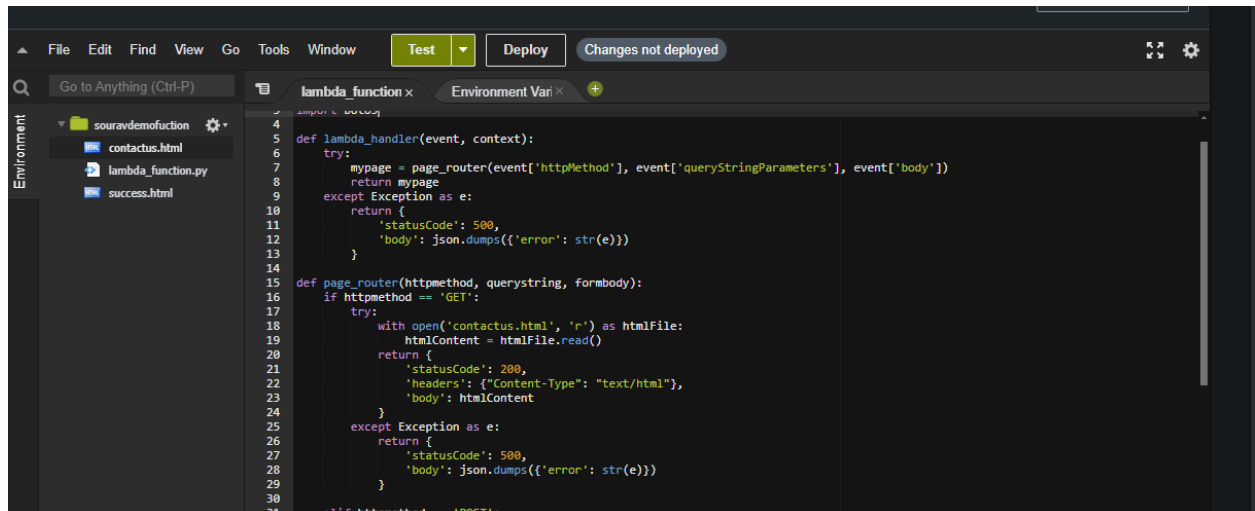
☒ Use an existing role

☐ 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 Lambda-role role](#) [↗](#) on the IAM console.



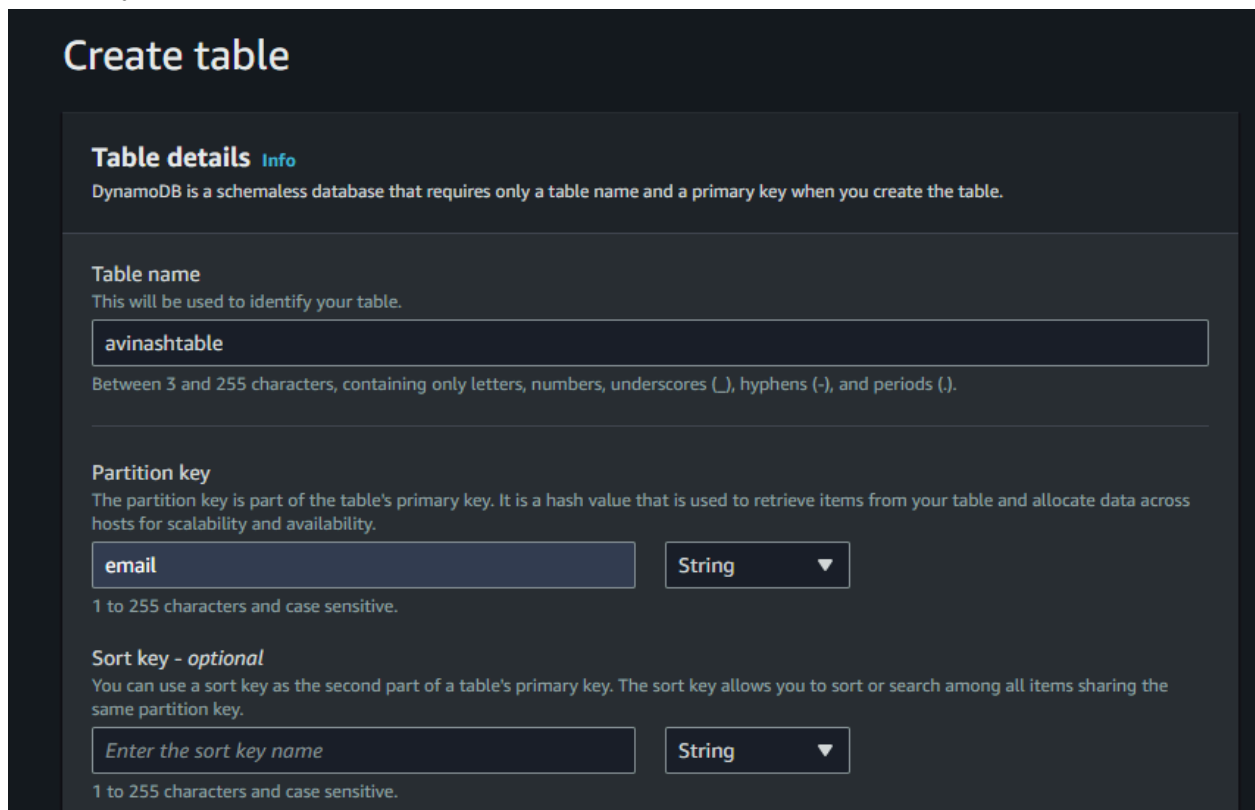
The screenshot shows an IDE with a file explorer on the left containing 'souravdemofunction', 'contactus.html', 'lambda_function.py', and 'success.html'. The main editor displays the 'lambda_function.py' file with the following Python code:

```
4
5 def lambda_handler(event, context):
6     try:
7         mypage = page_router(event['httpMethod'], event['queryStringParameters'], event['body'])
8         return mypage
9     except Exception as e:
10        return {
11            'statusCode': 500,
12            'body': json.dumps({'error': str(e)})
13        }
14
15 def page_router(httpmethod, querystring, formbody):
16     if httpmethod == 'GET':
17         try:
18             with open('contactus.html', 'r') as htmlFile:
19                 htmlContent = htmlFile.read()
20             return {
21                 'statusCode': 200,
22                 'headers': {'Content-Type': 'text/html'},
23                 'body': htmlContent
24             }
25         except Exception as e:
26             return {
27                 'statusCode': 500,
28                 'body': json.dumps({'error': str(e)})
29             }
30
31
```

For code follow the following GitHub Link:

https://github.com/souravchogale/AWS_Project_Deploy-web-application

Create DynamoDB:



The screenshot shows the 'Create table' page in the AWS Management Console. It includes a 'Table details' section with an 'Info' icon and a description of DynamoDB. The 'Table name' field is filled with 'avinastable'. The 'Partition key' is 'email' with a 'String' data type. The 'Sort key - optional' field is empty, with a placeholder 'Enter the sort key name' and a 'String' data type.

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.

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.

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.

1 to 255 characters and case sensitive.

Create API Gateway:

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

Import

Build

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

sourav-api-gateway

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

Create both get and post method:

API Gateway > APIs > Resources - sourav-api-gateway (9k0xjyvt2)

Resources

API actions ▼ Deploy API

Create resource

/

GET

POST

/ - POST - Method execution

Update documentation Delete

ARN
arn:aws:execute-api:us-east-1:637423206813:9k0xjyvt2/*/*POST/

Resource ID
7ksg92c0e2

Client

Method request

Integration request

Method response

Integration response
Proxy integration.....

Lambda integration

API Gateway > APIs > sourav-api-gateway (9k0xjyvt2) > Stages

Stages

Stage actions ▼ Create stage

dev

Stage details Info Edit

Stage name
dev

Web ACL
-

Burst Info
-

Default method-level caching
Inactive

Rate Info
-

Cache cluster Info
Inactive

Client certificate
-

Invoke URL
https://9k0xjyvt2.execute-api.us-east-1.amazonaws.com/dev

Copy invoke URL and paste it to the browser.

Output:

Welcome to my AWS Project Task

Contact Us

First Name:

Last Name:

Email ID:

Message:

2nd message

Submit

Welcome to my AWS Project Task

Contact Us

First Name:

Last Name:

Email ID:

Message:

Thanks for trying this Project. you can verify data in DynamoDB Table.

Tables (1)

Any tag key

Any tag value

Find tables

< 1 > ⚙

• avinashtable ☆

avinashtable

Autopreview

View table details

▶ Scan or query items

Expand to query or scan items.

✓ Completed. Read capacity units consumed: 0.5

Items returned (2)



Actions ▼

Create item

< 1 > ⚙

<input type="checkbox"/>	email (String) ▼	fname ▼	lname ▼	message ▼
<input type="checkbox"/>	narutouzumaki6136...	naruto	uzumaki	i+am+naruto...
<input type="checkbox"/>	souravchogale1997%...	sourav	chogale	My+aws+serv...