

API CRUD 45 MINUTOS:

DYNAMO DB:

http-crud-tutorial-items

Ações

Explorar itens da tabela

Visão geral

Índices

Monitorar

Tabelas globais

Backups

Exportações e streams

Permissões - novo

Configurações adicionais

Proteja sua tabela do DynamoDB contra gravações e exclusões acidentais

Editar PITR

×

Quando você ativa a recuperação a um ponto anterior no tempo (PITR), o DynamoDB faz backup dos dados da tabela automaticamente para que você possa restaurar a qualquer segundo nos últimos 35 dias. Cobranças adicionais se aplicam. [Saiba mais](#)

Informações gerais

Informações

Chave de partição id (String)	Chave de classificação -	Modo de capacidade Provisionada	Status da tabela Ativo
Alarmes Nenhum alarme ativo	Recuperação em um ponto anterior no tempo (PITR) Desativado	Política baseada em recursos Não ativo	

Informações adicionais

Tabelas (6)

Qualquer chave de etiqueta

Qualquer valor de etiqueta

Localizar tabelas por nome de tabelas

http-crud-tutorial-items

serverlesspresso-config-table

serverlesspresso-counting-table

serverlesspresso-metrics-table

serverlesspresso-order-table

serverlesspresso-validator-table

http-crud-tutorial-items

Visualização automática

Visualizar detalhes da tabela

Verificar ou consultar itens

Expandir para consultar ou verificar itens.

Concluído. Unidades de capacidade de leitura consumidas: 0.5

Itens retornados (1)

id (String)

1

LAMBDA:

Visão geral da função

Informações

Exportar para o Application Composer

Fazer download

Diagrama

Modelo

http-crud-tutorial-function

Layers (0)

API Gateway

(2)

Adicionar destino

Adicionar gatilho

Descrição

-

Última modificação

há 3 dias

ARN da função

arn:aws:lambda:us-east-1:992382375057:function:http-crud-tutorial-function

URL da função

Informações

Origem do códigoInformações

Fazer upload de

FileEditFindViewGoToolsWindowTestDeploy

Go to Anything (Ctrl-P)

index.js xEnvironment Vari x

Environment

http-crud-tutorial-findex.js

```
1 |
2 | const AWS = require("aws-sdk");
3 |
4 | const dynamo = new AWS.DynamoDB.DocumentClient();
5 |
6 | exports.handler = async (event, context) => {
7 |   let body;
8 |   let statusCode = 200;
9 |   const headers = {
10 |     "Content-Type": "application/json"
11 |   };
12 |
13 |   try {
14 |     switch (event.routeKey) {
15 |       case "DELETE /items/{id}":
16 |         await dynamo
17 |           .delete({
18 |             TableName: "http-crud-tutorial-items",
19 |             Key: {
20 |               id: event.pathParameters.id
21 |             }
22 |           })
23 |           .promise();
24 |         body = `Deleted item ${event.pathParameters.id}`;
25 |         break;
26 |       case "GET /items/{id}":
27 |         body = await dynamo
28 |           .get({
29 |             TableName: "http-crud-tutorial-items",
30 |             Key: {
31 |               id: event.pathParameters.id
32 |             }
33 |           })
34 |           .promise();
35 |         break;
36 |       case "GET /items":
37 |         body = await dynamo.scan({ TableName: "http-crud-tutorial-items" }).promise();
```

1:1 JavaScript Spaces: 2

API GATEWAY:

Rotas para http-crud-tutorial-api

Criar

Pesquisar

▼ /items

PUT

GET

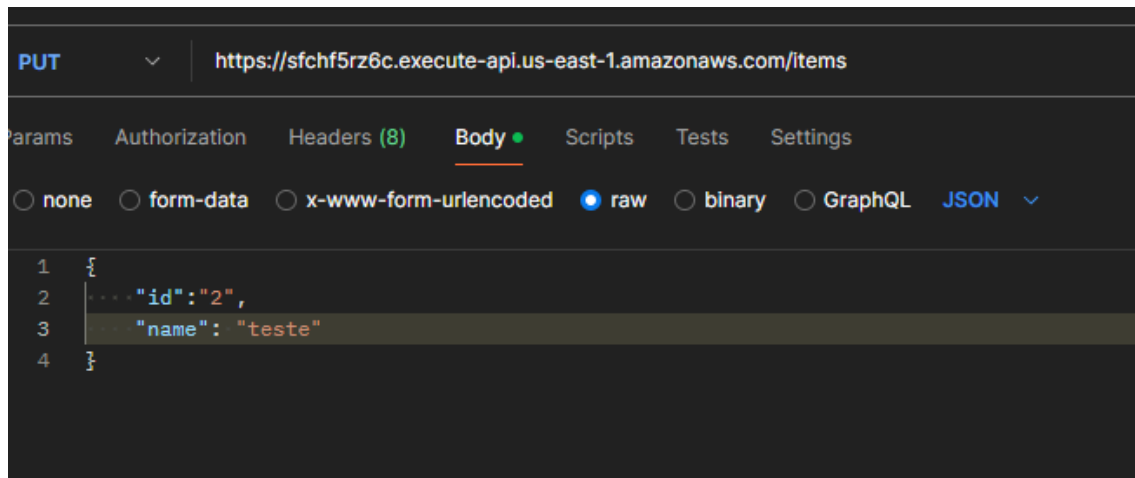
▼ /{id}

GET

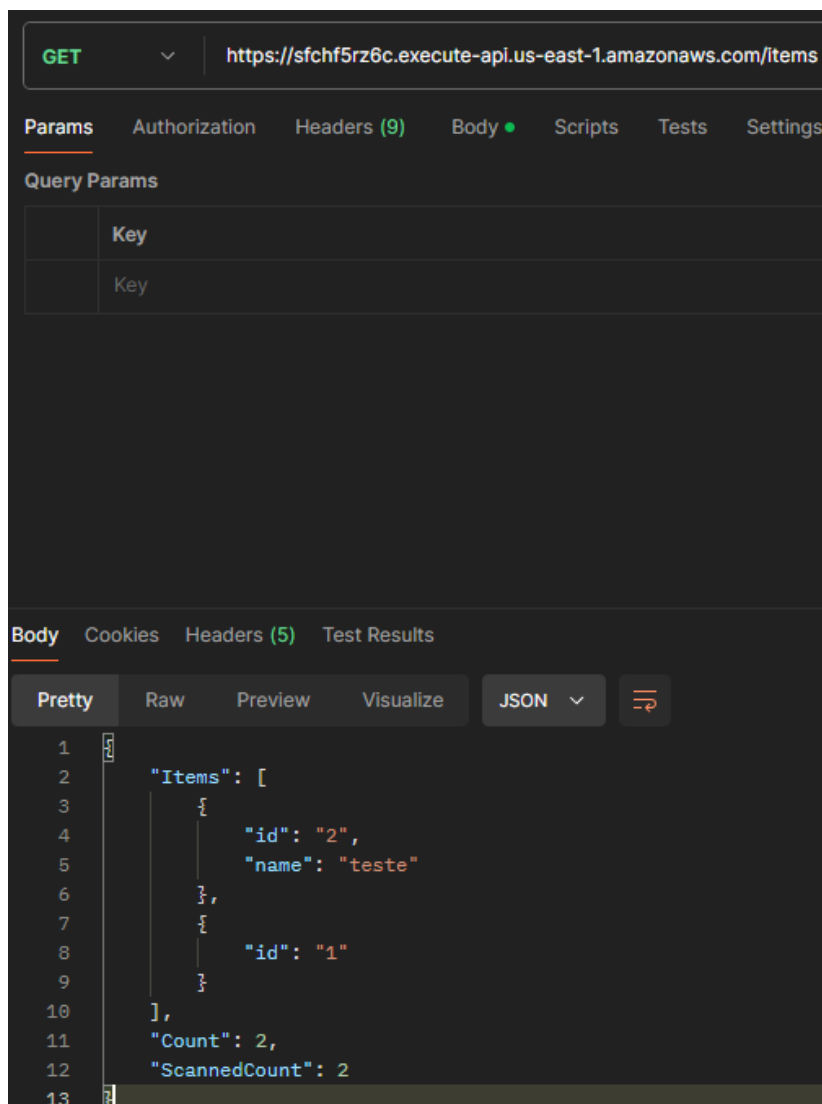
DELETE

MÉTODOS:

PUT:



GET:



DELETE:

