

## Consolidando o seu conhecimento

Chegou a hora de você seguir todos os passos realizados por mim durante esta aula. Caso já tenha feito, excelente. Se ainda não, segue um passo a passo.

No serviço IAM crie um novo *policy* e *role* para acessar DynamoDB:

- Sobre a *policy*:
  - Defina o serviço DynamoDB
  - Use a permissão `Read` apenas e adicione o ARN da tabela
  - Use o nome `PhotoCollection-DynamoDB_Acesso_a_tabela`
- Sobre o *role*:
  - No novo *role* escolha o serviço *API Gateway*
  - Use o nome `PhotoCollection-ROLE-DynamoDB_Acesso_a_tabela`
  - Ainda no *role* vincule a *policy* com *role*

De volta a API Gateway, na nossa API `PhotoCollection`, vamos criar um novos recursos `/photo` com sub-recursos `/id`, `/assunto` e `/consulta`.

- Na *PhotoCollection* crie um recurso `/photo`
  - Adicione um sub-recurso `/photo/{id}`
  - Use o método GET (*Resource Path*: `/photo/{id}`)
  - A integração deve ser AWS Service -> DynamoDB
  - Use Método `POST` na action `Scan`
  - Também coloque o nome da *role* criada
  - No *Integration Request* procure *Mapping Templates*
    - Escolha a segunda opção (a recomendada)
    - Adicione um template com o *Content-Type* e digite no campo `application/json`
    - Clique no template e adicione o JSON abaixo:

```
{
  "TableName" : "PhotoCollection",
  "FilterExpression" : "id = :v1",
  "ExpressionAttributeValues" : {
    ":v1" : { "S" : "$input.params('id') " }
  }
}
```

- Salve e volta a visão do recurso e clique no *Teste*
  - Teste com uma ID cadastrado no banco

Sub-recurso `/assunto`

- Adicione mais um sub-recurso `/assunto`
  - Use o método GET (*Resource Path*: `/photo/assunto`)

- Use as mesmas configurações:
- AWS Service -> DynamoDB, Method POST com action *Scan*, cole a role
- Crie um novo JSON template:
- Clique no template e adicione o JSON abaixo:

```
{
  "TableName" : "PhotoCollection",
  "ProjectionExpression" : "id, descricao, colecao",
  "FilterExpression" : "assunto = :v1",
  "ExpressionAttributeValues" : {
    ":v1" : { "S" : "$input.params('nome')" }
  }
}
```

- Salve e volta a visão do recurso e clique no *Teste*
  - Teste com *Query String*: nome=batman

#### Sub-recurso /consulta

- Use o método GET (*Resource Path*: /photo/consulta )
  - Configurações: AWS Service -> DynamoDB, Method POST com action *Scan*, cole a role
  - Template: ` { "TableName": "PhotoCollection", "FilterExpression": "assunto = :v1 AND colecao = v2", "ExpressionAttributeValues": { ":v1": { "S": "\$input.params('assunto')"}, ":v2": { "S": "\$input.params('colecao')"} } } `
- Salve e volta a visão do recurso e clique no *Teste*
- Teste com *Query String*: assunto=batman&colecao=dc

#### Mock Service /photo

- No *Integration Response* apague o template já existente e adicione um novo template HTML com o conteúdo da página HTML abaixo desse vídeo
  - Adicione um header da resposta: Content-Type e 'text/html'

Por favor faça um novo deploy e teste os recursos.