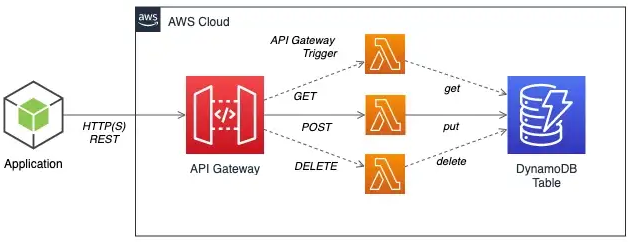
**DEPLOY APPLICATION IN THE AWS CLOUD**

When an user makes a request via the RESTful API exposed by the API Gateway, an event is fired, which invokes a [Lambda](https://github.com/garystafford/serverless-sqs-dynamo-demo/blob/master/lambda_apigtw_to_dynamodb/app.js) function using the [API Gateway Event Source for Lambda](https://docs.aws.amazon.com/lambda/latest/dg/with-on-demand-https.html) functionality. The event contains details about the HTTP request that is received. The event triggers the lambda function corresponding with the HTTP request method. All data are store in a DynamoDB service.



**INSTRUCTIONS**

1. Install serverless framework (https://www.serverless.com/)

**$ npm install -g serverless**

2. Add AWS credentials (only first time)

**$ serverless config credentials --provider aws --key [key] --secret [secret]**

3. We check that the content of the **serverless.yml** file is correct:

service: fresco-recipes

provider:

name: aws

runtime: python3.7

region: eu-west-1

functions:

service:

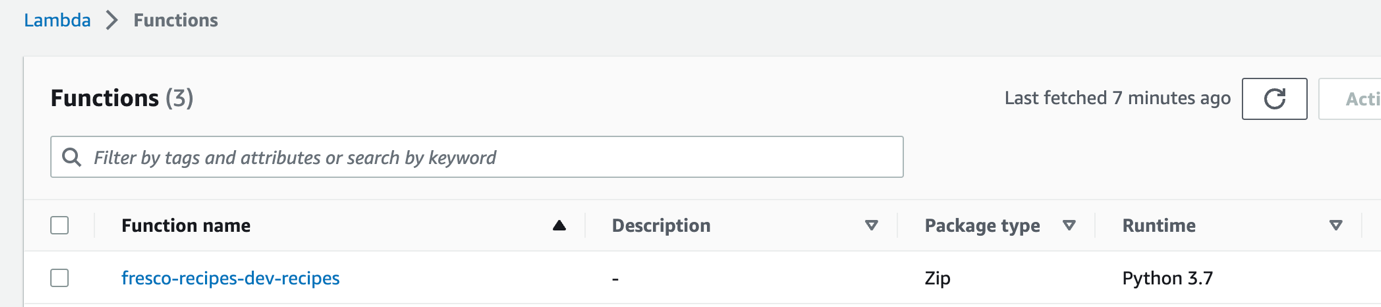
handler: lambda\_function.lambda\_handler

4. We deploy the application

**$ sls deploy**

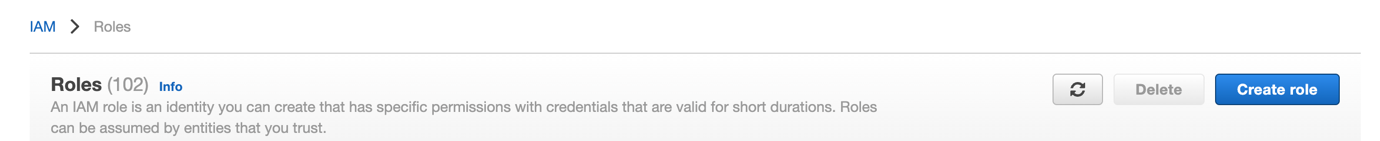
And check that our lambda function has been created in AWS Cloud.

Lambda > Functions



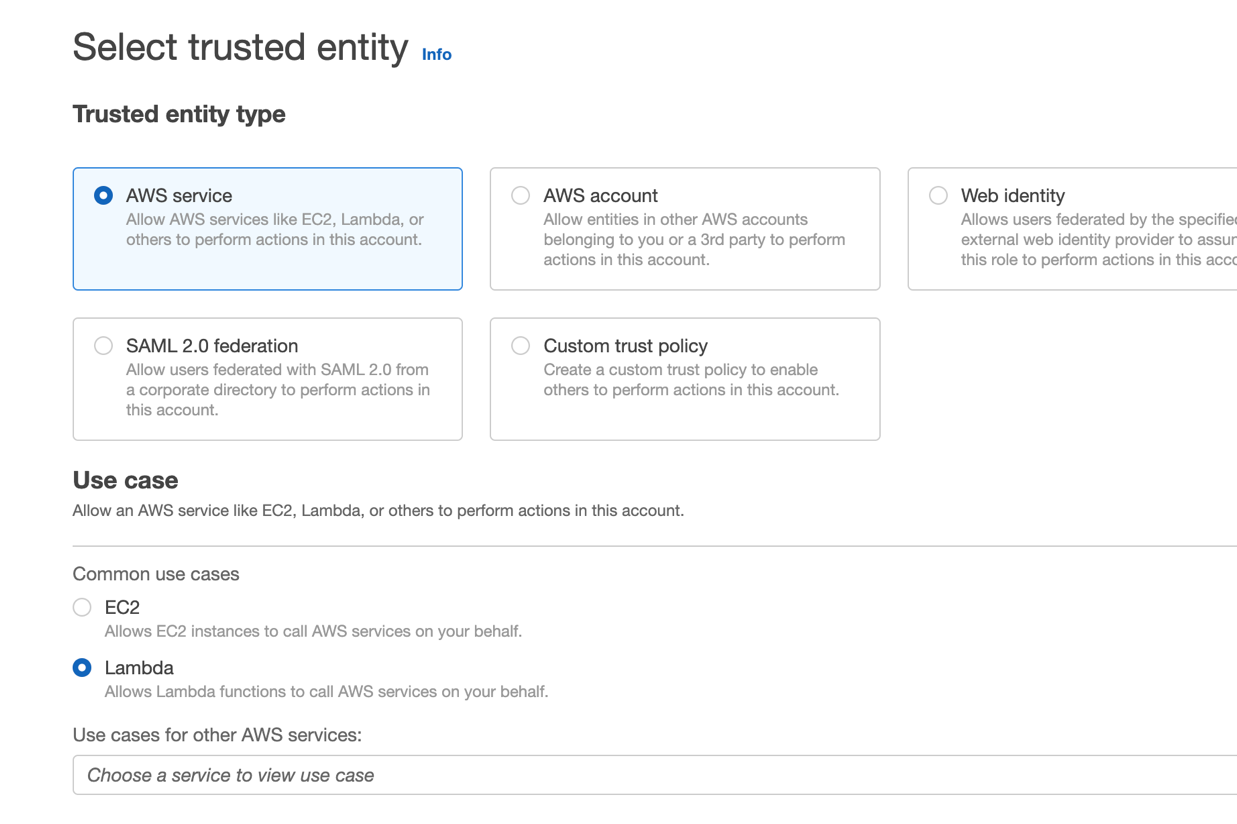
5. Create a IAM (Identity and Access Management) role:

IAM > Roles



We click on “Create role”

Select AWS Service, Use Case Lambda and Next



Select the following policies:

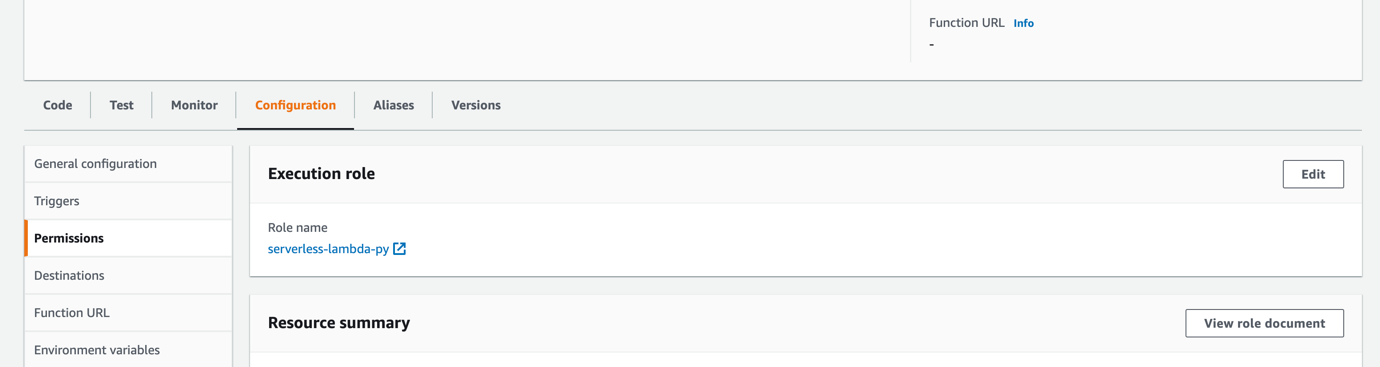
CloudWatchFullAccess

AmazonDynamoDBFullAccess

And give a role name: **serverless-lambda-py**

6. We associate the role that we have created to our lambda

Configuration > Execution role > Edit



And select **serverless-lambda-py.**

7. We need to assign a layer with the required libraries to our lambda function.

For that, we run the following script

**$ ./get\_layer\_packages.sh**

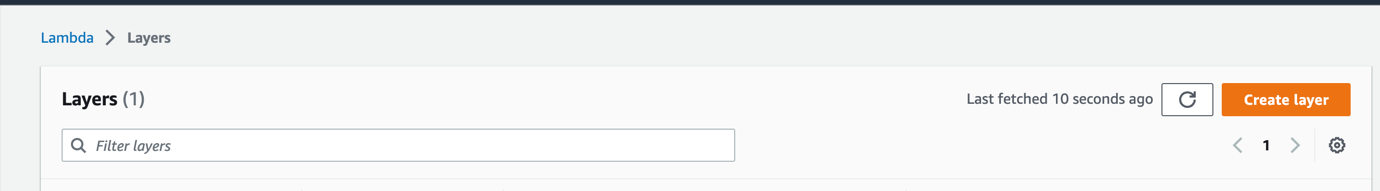
This script downloads the libraries described in the file requirements.txt in the folder ‘python’.

We zip this folder:

**$ zip -r fresco\_layer.zip python/**

Now:

Lambda > Layer > Create Layer

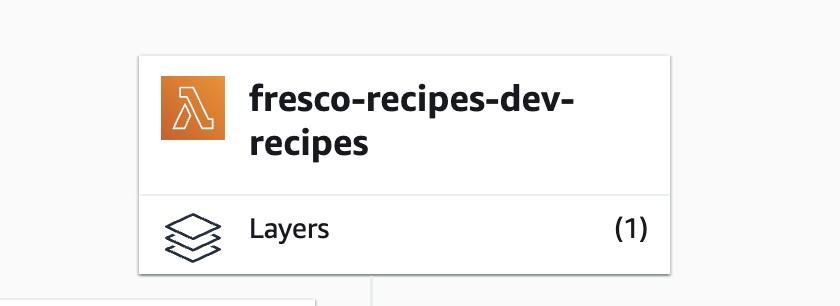


And upload the zip file. Select runtime python 3.7 and “Create layer”.

We copy the arn generated:

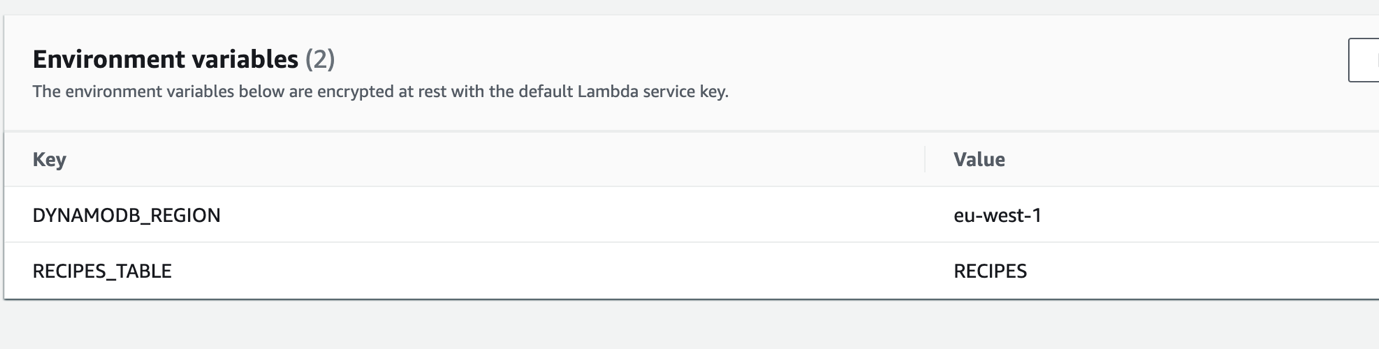
arn:aws:lambda:eu-west-1:980508671241:layer:fresco\_layer:4

Now we attach this file to our lambda function. For that, in the function overview, we click on “layers”:



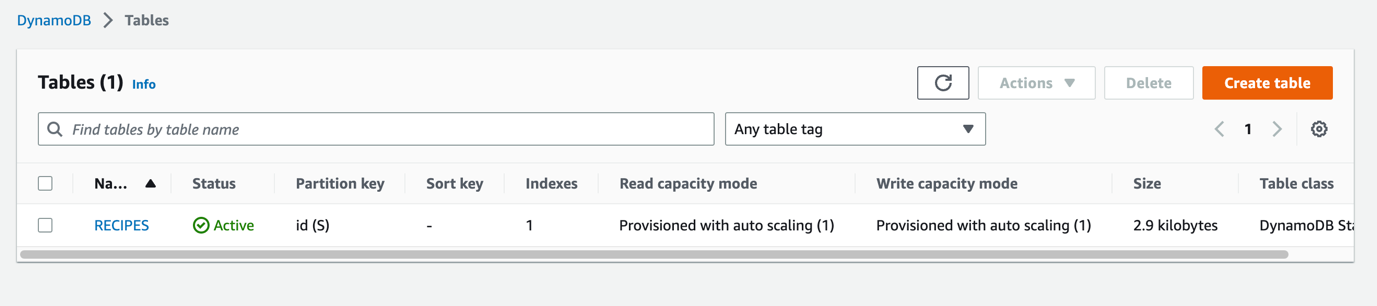
We click on “add a layer” and coy the arn text we copied before.

8. We fill environment variables.



9. We create table RECIPES in DynamoDB.

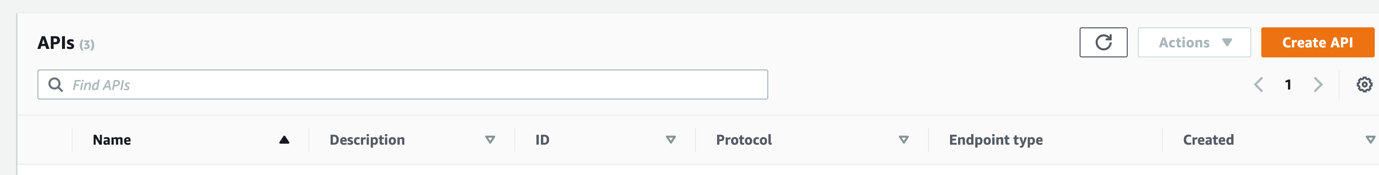
For that, we click in Create Table:



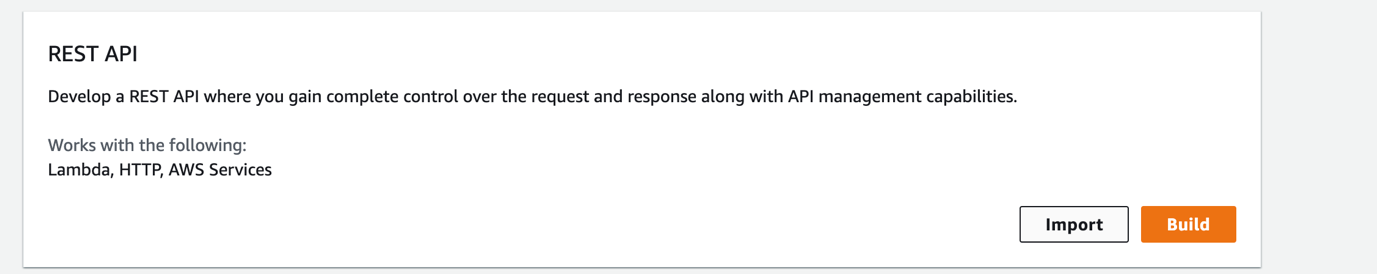
We name it “RECIPES” and fill “partition key” with “id”.

10. We create the API.

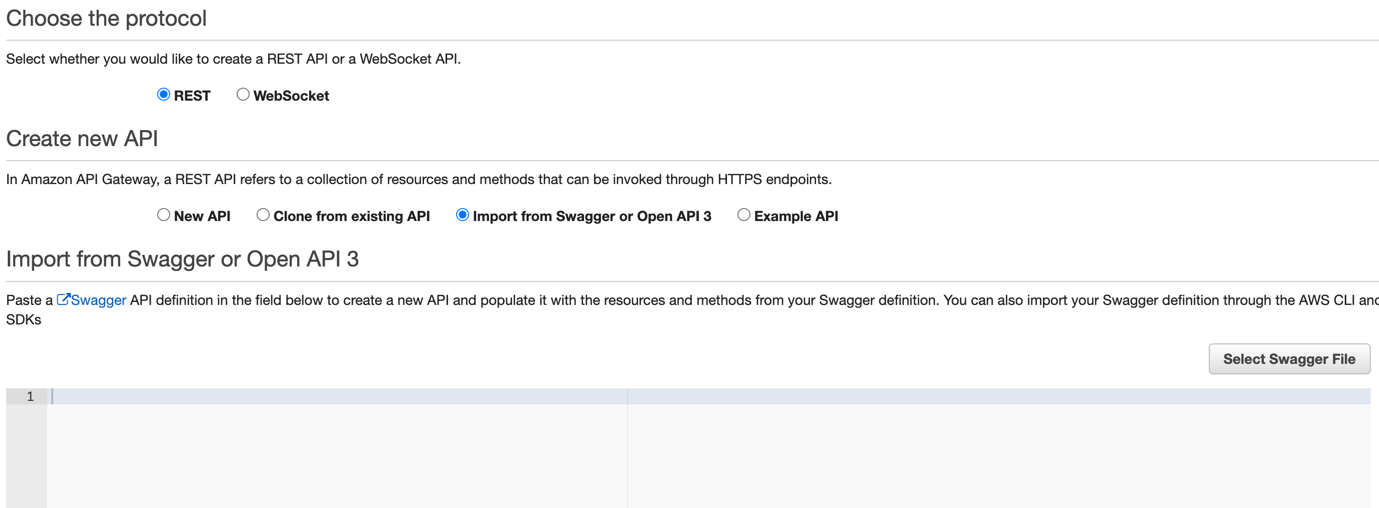
We go to API Gateway service and click on “Create API”



In REST API option, we click on “Import”



We select “Select Swagger file” and click on “import”.



11. We copy the invoke url of our API:

