

@rapkyt

SERVERS FOR SERVERS

APLICACIÓN EJEMPLO - NEWSLETTER

28

ldn endpoint POST /newsletter para registrar.

```
service: serverless-effortless
provider:
  name: aws
  runtime: python3.6
  region: ${opt:region, 'us-west-2'}
  stage: ${opt:stage, self:custom.defaultStage}
  memorySize: 512
  logRetentionInDays: 30
  environment:
    DYNAMODB_TABLE: ${self:custom.dynamodb_resource}
  iamRoleStatements:
    - Effect: Allow
      Action:
        - dynamodb:*
      Resource: ${self:custom.dynamodb_resource}
custom:
  dynamodb_table: ${self:service}-${opt:stage, self:provider.stage}
```




APLICACIÓN DE EJEMPLO - NEWSLETTER

Un endpoint POST /newsletter para registrarse.

```
service: serverless-effortless
provider:
  name: aws
  runtime: python3.6
  region: ${opt:region, 'us-west-2'}
  stage: ${opt:stage, self:custom.defaultStage}
  memorySize: 512
  logRetentionInDays: 30
  environment:
    DYNAMODB_TABLE: ${self:custom.dynamodb_resource}
  iamRoleStatements:
    - Effect: Allow
      Action:
        - dynamodb:*
      Resource: ${self:custom.dynamodb_resource}
custom:
  dynamodb_table: ${self:service}-${opt:stage, self:provider.stage}
```

APLICACIÓN DE EJEMPLO – NEWSLETTER

Un endpoint POST /newsletter para registrarse.

Resources:

NewsletterDynamoDbTable:

Type: 'AWS::DynamoDB::Table'

DeletionPolicy: Retain

Properties:

AttributeDefinitions:

-

AttributeName: email

AttributeType: S

KeySchema:

-

AttributeName: email

KeyType: HASH

ProvisionedThroughput:

ReadCapacityUnits: 1

WriteCapacityUnits: 1

TableName: \${self:custom.dynamodb_resource}