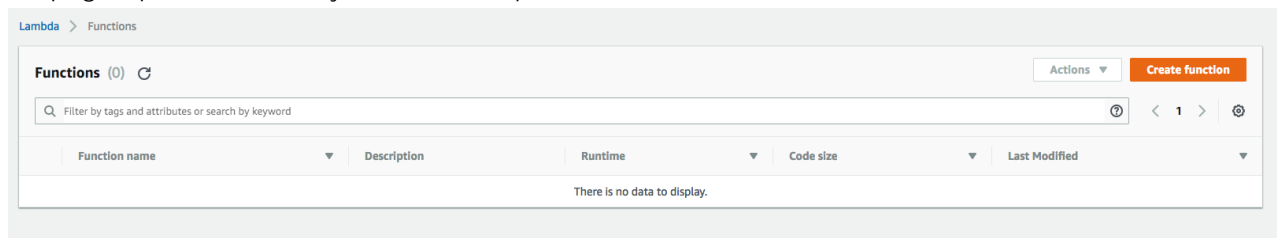


Aula 02.1 - Primeira api com lambda

Criando o lambda

1. Na pagina principal do serviço Lambda, clique em 'Create Function'



2. Preencha com as informações pedidas

Create function

Author from scratch (selected) Start with a simple "hello world" example.

Blueprints Choose a preconfigured template as a starting point for your Lambda function.

Serverless Application Repository Find and deploy serverless apps published by developers, companies, and partners on AWS.

Author from scratch info

Name: helloworld

Runtime: Python 3.6

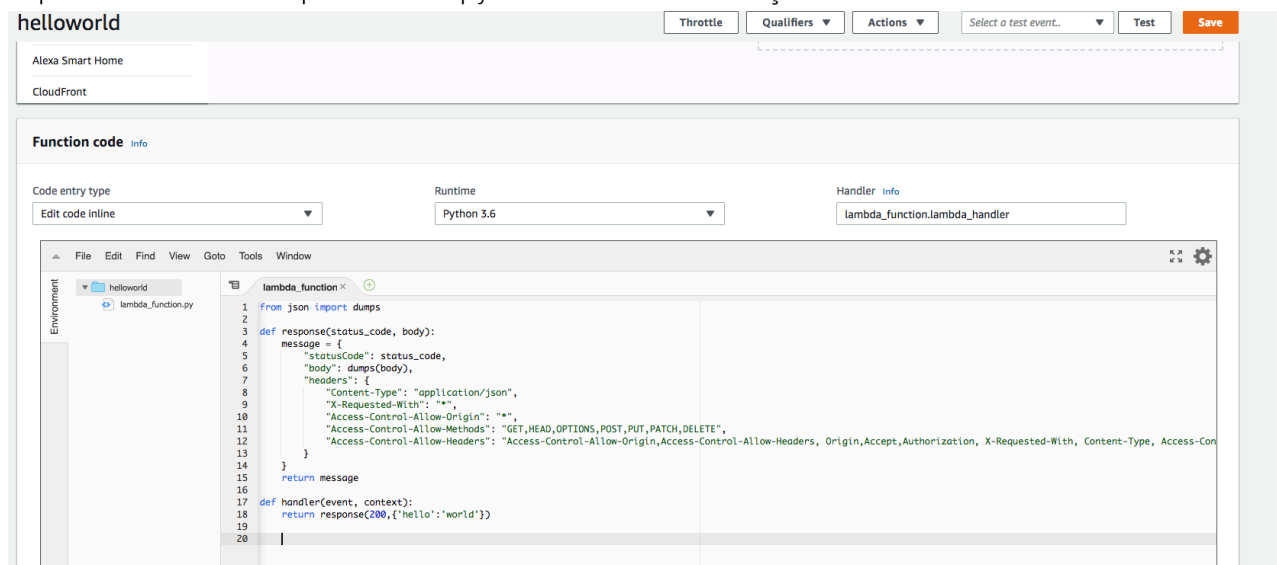
Role: Defines the permissions of your function. Note that new roles may not be available for a few minutes after creation. [Learn more](#) about Lambda execution roles.

Create new role from template(s)

Role name: hello-world

Policy templates: Choose one or more policy templates. A role will be generated for you before your function is created. [Learn more](#) about the permissions that each policy template will add to your role.

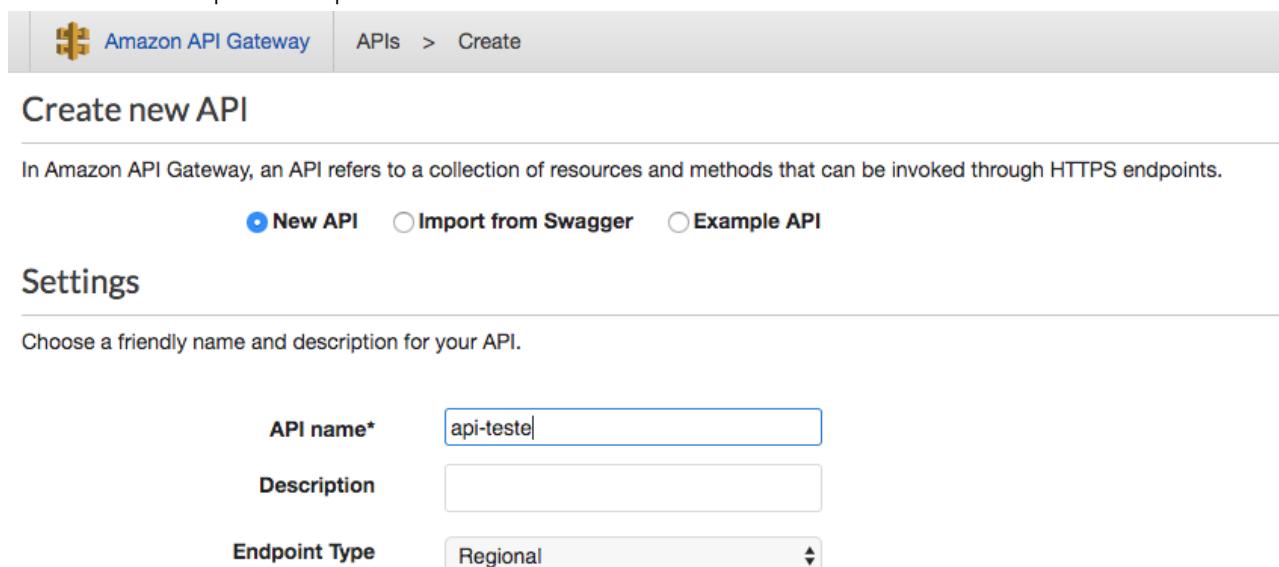
3. copie o conteudo do arquivo lambda.py e cole no editor da função



4. Clique em 'Save' no topo da página

Criando a api

1. Selecione 'new api' e coloque o nome:



Amazon API Gateway APIs > Create

Create new API

In Amazon API Gateway, an API refers to a collection of resources and methods that can be invoked through HTTPS endpoints.

☒ New API ☐ Import from Swagger ☐ Example API

Settings

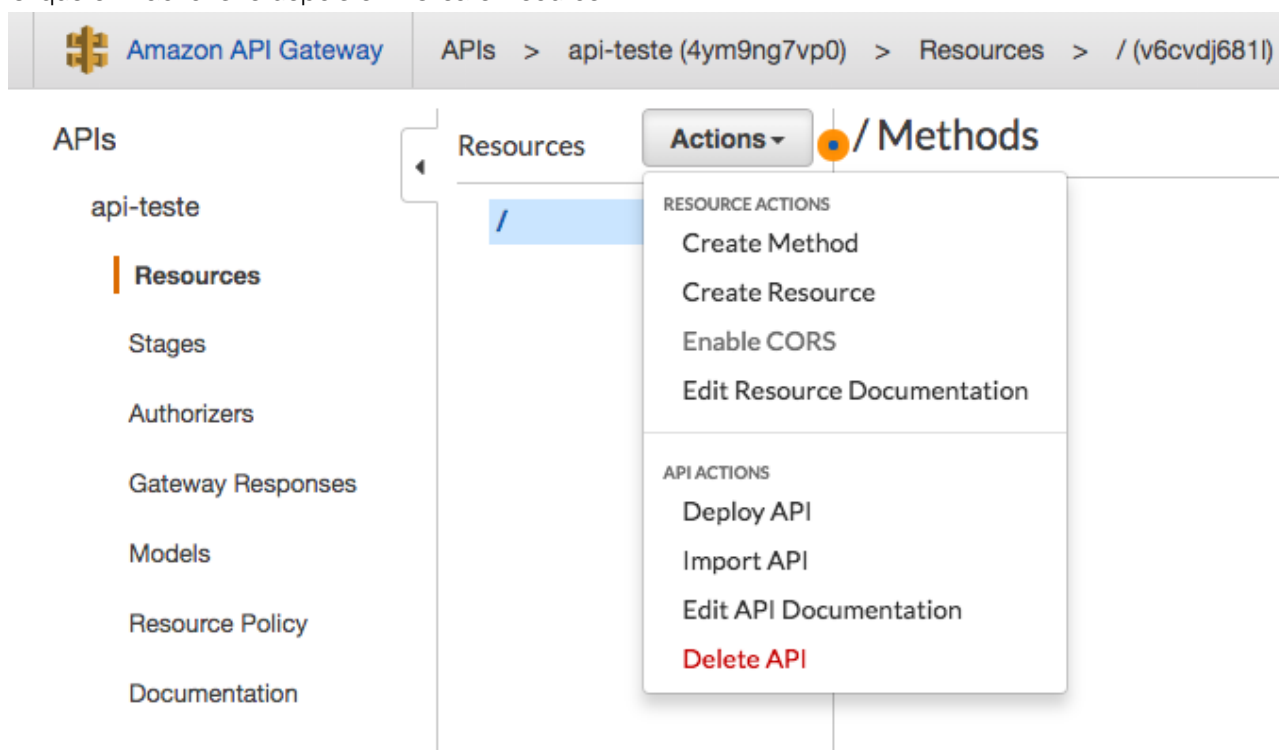
Choose a friendly name and description for your API.

API name*

Description

Endpoint Type

2. Clique em 'actions' e depois em 'Create Resource'



Amazon API Gateway APIs > api-teste (4ym9ng7vp0) > Resources > / (v6cvdj681l)

APIs

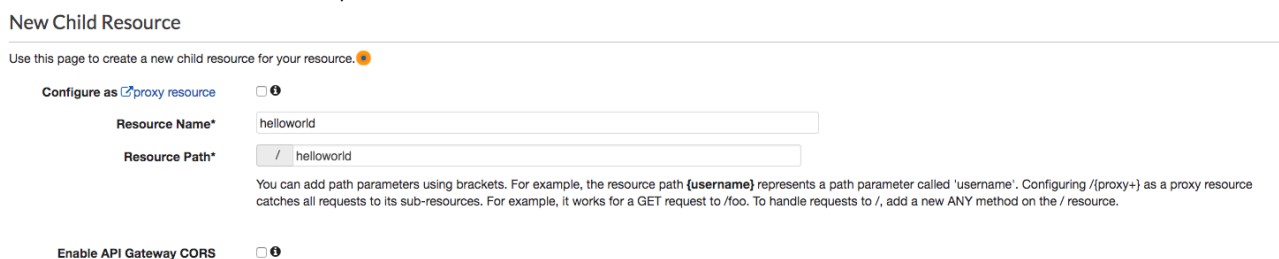
- api-teste
- Resources**
- Stages
- Authorizers
- Gateway Responses
- Models
- Resource Policy
- Documentation

Resources

Actions ▾

- RESOURCE ACTIONS
 - Create Method
 - Create Resource
 - Enable CORS
 - Edit Resource Documentation
- API ACTIONS
 - Deploy API
 - Import API
 - Edit API Documentation
 - Delete API

3. Preencha o formulario e clique no botão 'Create Resource'



APIs > api-teste (4ym9ng7vp0) > Resources > / (v6cvdj681l) > New Child Resource

New Child Resource

Use this page to create a new child resource for your resource.

Configure as [proxy resource](#) ☐

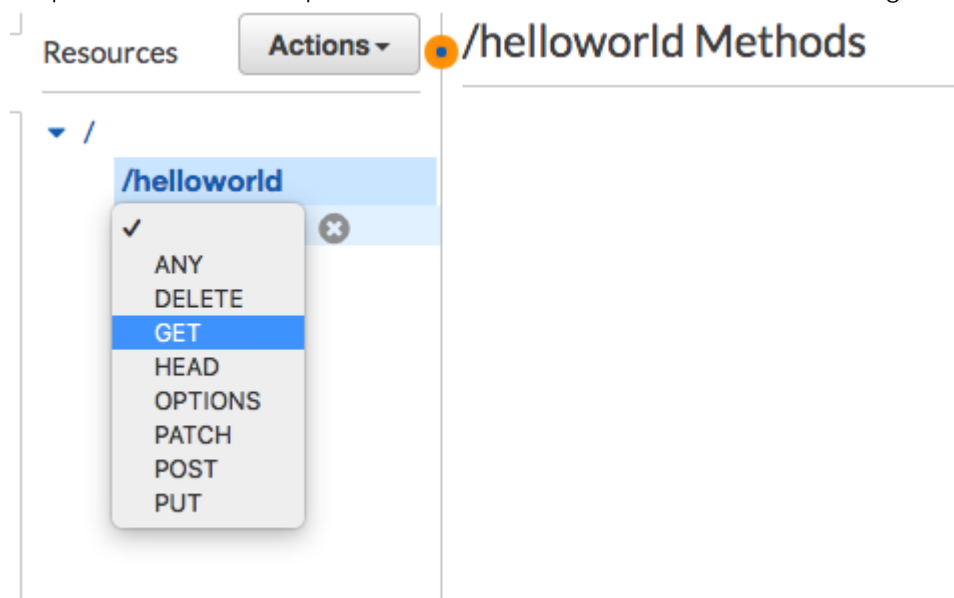
Resource Name*

Resource Path*

You can add path parameters using brackets. For example, the resource path **{username}** represents a path parameter called 'username'. Configuring **/ {proxy+}** as a proxy resource catches all requests to its sub-resources. For example, it works for a GET request to /foo. To handle requests to /, add a new ANY method on the / resource.

Enable API Gateway CORS ☐

4. Clique em 'Actions' e deplois em 'Create Method' e selecione o metodo get



5. Preencha os parametros conforme a imagem, após clique em save

/helloworld - GET - Setup

Choose the integration point for your new method.

Integration type ☒ Lambda Function ⓘ

☐ HTTP ⓘ

☐ Mock ⓘ

☐ AWS Service ⓘ

☐ VPC Link ⓘ

Use Lambda Proxy integration ☐ ⓘ


Lambda Region

Lambda Function ⓘ

Use Default Timeout ☒ ⓘ

6. Clique em 'Actions' e selecione 'Deploy API'

7. Em 'Deployment Stage' selecione '[New Stage]', preencha o que é pedido, e clique em 'Deploy'

Deploy API 

Choose a stage where your API will be deployed. For example, a test version of your API could be deployed to a stage named beta.

Deployment stage	<input type="text" value="[New Stage]"/>
Stage name*	<input type="text" value="Dev"/>
Stage description	<input type="text" value="Dev"/>
Deployment description	<input type="text"/>

Cancel **Deploy**

8. Para testar é só utilizar a url que foi fornecida e adicionar /helloworld no navegador

