

Professional Skills
Agile Fundamentals
Jira
Git
Databases Introduction
Java Beginner
Maven
Testing (Foundation)
Java Intermediate
HTML
CSS
Javascript
Spring Boot
Selenium
Sonarqube
Advanced Testing (Theory)
Cucumber
MongoDB
Express
NodeJS
React
Express-Testing
Networking
Security
Cloud Fundamentals
AWS Foundations
AWS Intermediate
<div><div></div>Virtual Private Cloud (VPC)</div>
<div><div></div>EC2 VPC Security Groups</div>
<div><div></div>EC2 VPC Subnets</div>

AWS Lambda Functions

Contents

- [Overview](#)
- [Tutorial](#)
- [Exercises](#)

Overview

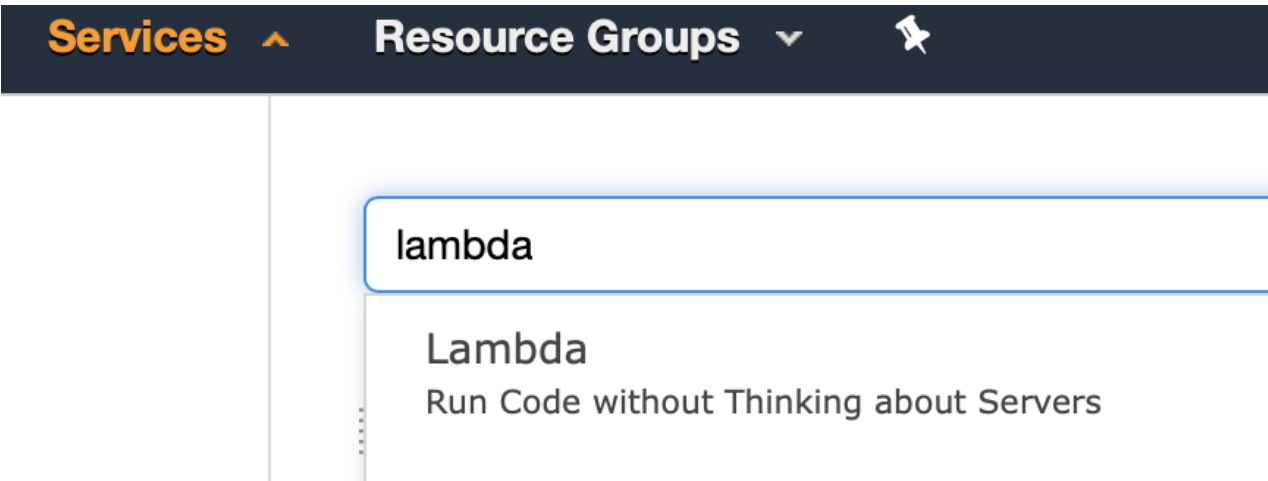
AWS Lambda is a compute service that lets you run code without provising or managing servers. AWS Lambda only executes code when needed and scales automatically.

You only pay for the time that consumed computing power. This means that AWS Lambda is serverless and can be used to construct a serverless architecture comprised of triggers from AWS Codepipeline and invoking Lambda functions.

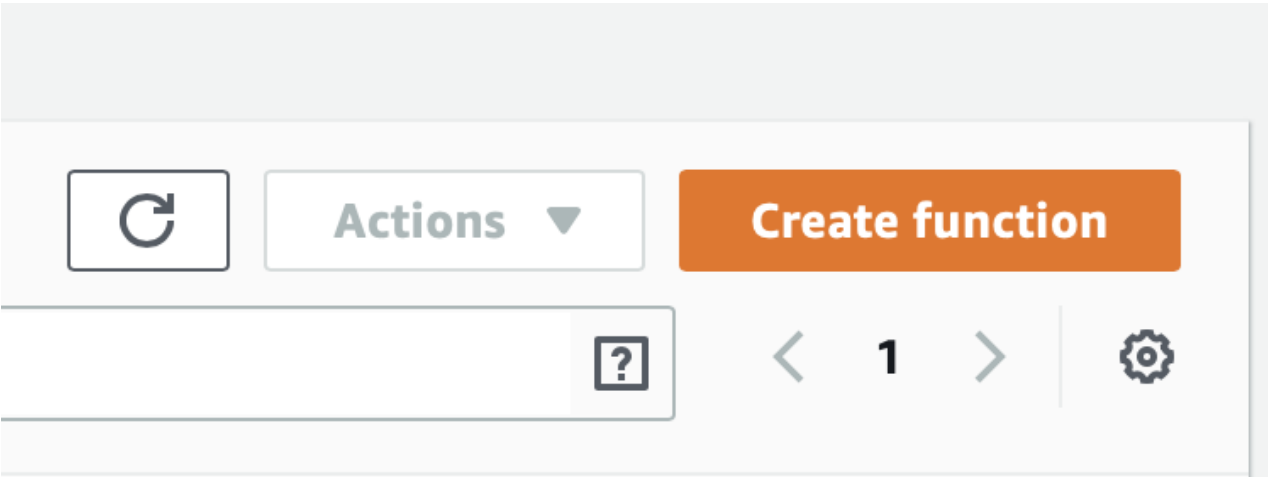
In this tutorial, we will go through developing a simple AWS Lambda function and adding a trigger that will invoke the Lambda function.

Tutorial

- Navigate to the AWS Console and sign in [here](#)
- Search for **Lambda** under the services drop-down menue



- This next window, you will need to click the **Create function** button




- You will then be presented with options on how you would like to create your Lambda function.
 - Check **Author from scratch**
 - Function name, choose your own function name, here it will be, **ExampleLambdaFunction**
 - Runtime, you have a large selection, but we will be making a python function, select **Python 3.8**

The rest can be left as default.

<div><div></div><div>EC2 VPC Internet Gateways</div></div> <div><div></div><div>AWS Route Tables</div></div> <div><div></div><div>AWS Network Address Translation (NAT) Gateway</div></div> <div><div></div><div>AWS Network Access Control Lists (NACLs) CLI</div></div> <div><div></div><div>AWS Java SDK</div></div> <div><div></div><div>AWS DynamoDB</div></div> <div><div></div><div>AWS Lambda Functions</div></div> <div><div></div><div>AWS API Gateway</div></div> <div><div></div><div>SQS Introduction</div></div> <div><div></div><div>AWS Serverless CRUD Solution</div></div> <div><div></div><div>AWS Serverless Solution with DynamoDB</div></div> <div><div></div><div>CloudWatch CLI</div></div> <div><div></div><div>CloudTrail</div></div>	
Linux	
DevOps	
Jenkins Introduction	
Jenkins Pipeline	
Markdown	
IDE Cheatsheet	

Author from scratch

Start with a simple Hello World example.



Basic information

Function name

Enter a name that describes the purpose of your function.

ExampleLambdaFunction

Use only letters, numbers, hyphens, or underscores with no spaces.

Runtime

Info

Choose the language to use to write your function.

Python 3.8

Permissions

Info

Lambda will create an execution role with permission to upload logs to Amazon CloudWatch Logs. You can

▼ Choose or create an execution role

Execution role

Choose a role that defines the permissions of your function. To create a custom role, go to the [IAM console](#)

Create a new role with basic Lambda permissions

Use an existing role

Create a new role from AWS policy templates

Role creation might take a few minutes. Please do not delete the role or edit the

Lambda will create an execution role named ExampleLambdaFunction-role-i8inr0yh, with

5. Proceed by clicking **Create function**
6. This next window, we will be focusing on the **Designer** section.
First we are going to need to add a trigger that will invoke our **Lambda Function**.
Click on **+ Add Trigger** under the **Designer** section

ConfigurationPermissionsMonitoring

▼ Designer

+ Add trigger

7. You will be presented with a drop-down menu which have a lot of methods of invoking a Lambda function. Select **CloudWatch Events**, then an option for **Rule** will appear. Select **Create a new rule**.

8. Fill in the details as follows:

Add trigger

Trigger configuration

CloudWatch Events

aws events management-tools

▼

Rule

Pick an existing rule, or create a new one.

Create a new rule

▼

Select or create a new rule

Rule name*

Enter a name to uniquely identify your rule.

ExampleLambdaRule

Rule description

Provide an optional description for your rule.

Example rule for lambda function

Rule type

Trigger your target based on an event pattern, or based on an automated schedule.

☐ Event pattern

☒ Schedule expression

Schedule expression*

Self-trigger your target on an automated schedule using Cron or rate expressions. Cron expressions are in UTC.

rate(2 minutes)

e.g. rate(1 day), cron(0 17 ? * MON-FRI *)

Lambda will add the necessary permissions for Amazon CloudWatch Events to invoke your Lambda function from this trigger. [Learn more](#) about the Lambda permissions model.

☒ Enable trigger

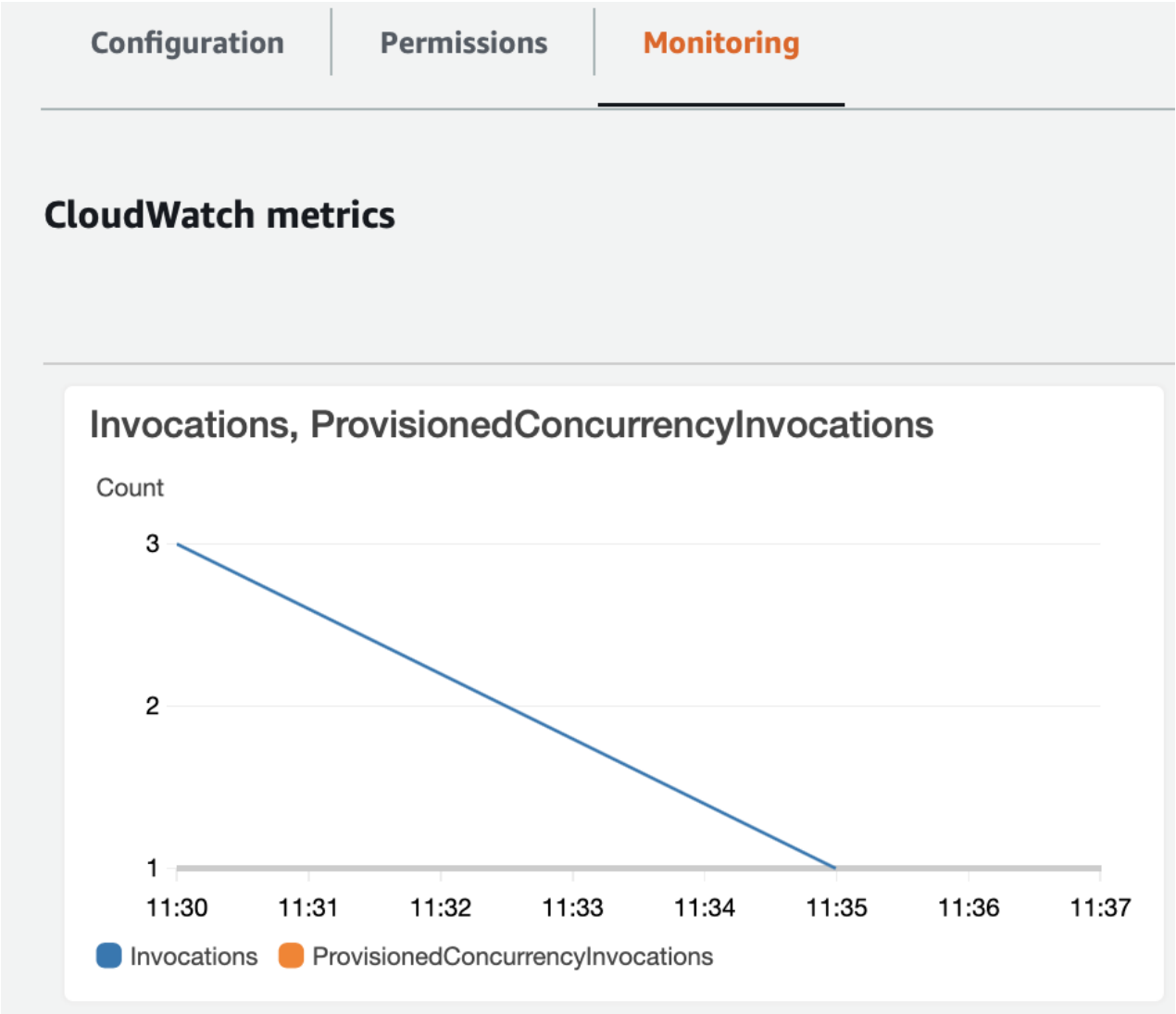
Enable the trigger now, or create it in a disabled state for testing (recommended).

Cancel

Add

9. Proceed to **Add** this trigger

10. Now, if you navigate to the **Monitoring** and wait a few minutes, you will be able to see the number of invocations made.



11. Delete the function, so you are not running this constantly every 2 minutes, as this could potentially charge you without you knowing. Navigate to **Actions** and then **Delete function**

ThrottleQualifiersActionsHelloWorld

Publish new version

Create alias

Delete function

Export function

Exercises

There are no exercises for this module.