

## Guided Lab: Aliases and Versions in AWS Lambda

### Description

In AWS Lambda, versions and aliases are closely related tools that help you manage and deploy your functions more effectively. Versions allow you to lock in a specific iteration of your function, while aliases act as pointers to these versions, making it easier to manage different environments such as development, staging, and production. Aliases can also be used for weighted traffic routing, enabling gradual rollouts or A/B testing of new function versions.

This lab will guide you through the process of creating and managing versions and aliases in AWS Lambda. You will also learn how to implement weighted traffic routing between different versions of your Lambda function.

### Prerequisites

This lab assumes you have basic understanding of AWS Lambda and Node.js programming language.

If you find any gaps in your knowledge, consider taking the following lab:

- Creating an AWS Lambda function
- Creating a Node.js Function in AWS Lambda

### Objectives

By the end of this lab, you will be able to:

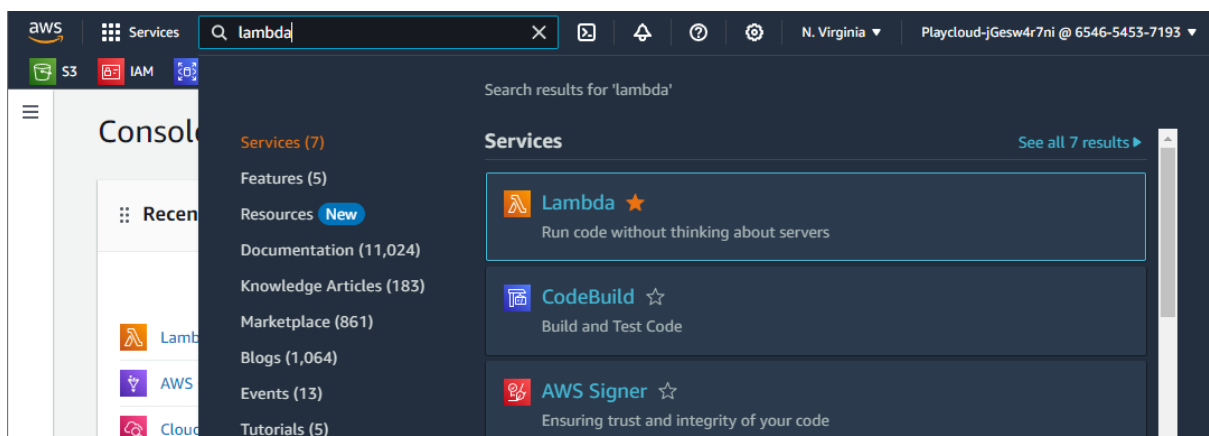
- Create a version of your Lambda function.
- Modify the function code and publish a new version.
- Create and update an alias to point to specific versions.
- Implement weighted traffic routing between different versions using aliases.

### Lab Steps

#### Creating and Testing an AWS Lambda Function

##### 1. CREATE AN AWS LAMBDA FUNCTION

- Navigate to AWS Lambda Console



- **Create Function using the following configurations:**
  - **Select Use a blueprint**
  - **Blueprint name : Hello world function nodejs18.x**
  - **Function name: Enter a name for your function (e.g., VersionAliasLambda).**
  - **Execution role:**
    - **Select Use an Existing Role: PlayCloud-Sandbox**

[Lambda](#) > [Functions](#) > Create function

## Create function Info

Choose one of the following options to create your function.

☐ **Author from scratch**  
 Start with a simple Hello World example.

☒ **Use a blueprint**  
 Build a Lambda application from sample code and configuration presets for common use cases.

☐ **Container image**  
 Select a container image to deploy for your function.

---

### Basic information Info

**Blueprint name**

A starter AWS Lambda function.

**Function name**  
 Enter a name that describes the purpose of your function.

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

**Runtime**  
 nodejs18.x

**Architecture**  
 x86\_64

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

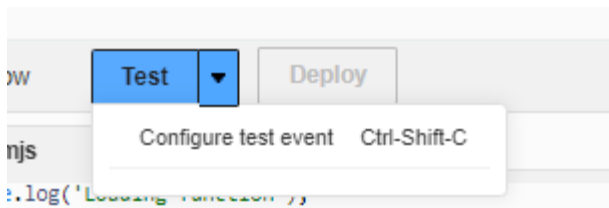
☐ Create a new role with basic Lambda permissions  
☒ Use an existing role  
☐ Create a new role from AWS policy templates

**Existing role**  
 Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

[View the PlayCloud-Sandbox role](#) on the IAM console.

- - **Click on Create Function**

**2. Once your function is created. Click the arrow dropdown of the BLUE Test button**



3. Click on Configure test event, and follow the configuration below:

- Event name: VersionTest
- Paste the following JSON to the Event JSON field:

```
{
  "key1": "Version Boracay",
  "key2": "Version Palawan",
  "key3": "Version Manila"
}
```

**Configure test event**

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action

☒ Create new event ☐ Edit saved event

Event name

VersionTest

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

☒ Private  
This event is only available in the Lambda console and to the event creator. You can configure a total of 10. [Learn more](#)

☐ Shareable  
This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

Template - optional

hello-world

**Event JSON** [Format JSON](#)

```
1 {
2   "key1": "Version Boracay",
3   "key2": "Version Palawan",
4   "key3": "Version Manila"
5 }
```

Cancel Invoke **Save**

- Click on Save

4. Now, click on Test. The current response should be:

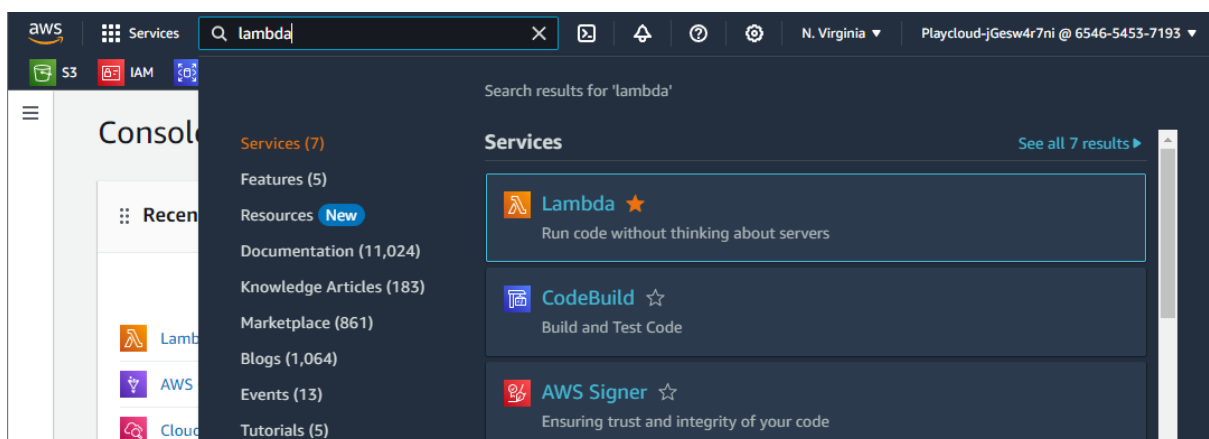
index.mjs	Environment Vari	Execution result
Execution results		
Status: <b>Succeeded</b> Max memory used: 67 MB Time: 4.79 ms		
Test Event Name		
VersionTest		
Response		
"Version Boracay"		
Function Logs		
2024-08-13T08:53:57.099Z undefined INFO Loading function		
START RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c Version: \$LATEST		
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value1 = Version Borac		
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value2 = Version Palaw		
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value3 = Version Manil		
END RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c		
REPORT RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c Duration: 4.79 ms Billed Duration: 5 ms		
Request ID		
4c3f897d-00c0-49e0-8d51-06e715be390c		

## Lab Steps

### Creating and Testing an AWS Lambda Function

#### 1. CREATE AN AWS LAMBDA FUNCITON

- Navigate to AWS Lambda Console



- Create Function using the following configurations:
  - Select Use a blueprint
  - Blueprint name : **Hello world function nodejs18.x**
  - Function name: Enter a name for your function (e.g., **VersionAliasLambda**).
  - **Execution role:**
    - Select Use an Existing Role: **PlayCloud-Sandbox**

[Lambda](#) > [Functions](#) > Create function

## Create function [Info](#)

Choose one of the following options to create your function.

☐ **Author from scratch**  
Start with a simple Hello World example.

☒ **Use a blueprint**  
Build a Lambda application from sample code and configuration presets for common use cases.

☐ **Container image**  
Select a container image to deploy for your function.

---

### Basic information [Info](#)

**Blueprint name**

Hello world function  
A starter AWS Lambda function.

nodejs18.x ▼

**Function name**  
Enter a name that describes the purpose of your function.

VersionAliasLambda

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

**Runtime**  
nodejs18.x

**Architecture**  
x86\_64

**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

**Existing role**  
Choose an existing role that you've created to be used with this Lambda function. The role must have permission to upload logs to Amazon CloudWatch Logs.

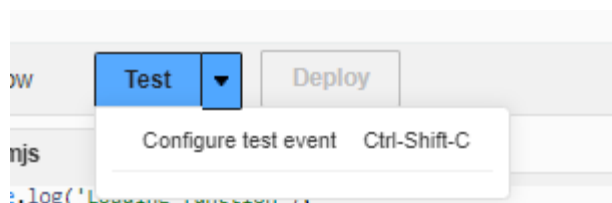
PlayCloud-Sandbox ▼

↻

[View the PlayCloud-Sandbox role](#) on the IAM console.

- - Click on **Create Function**

2. Once your function is created. Click the arrow dropdown of the BLUE **Test** button



3. Click on **Configure test event**, and follow the configuration below:

- Event name: VersionTest
- Paste the following JSON to teh **Event JSON field**:

```
{  
  "key1": "Version Boracay",  
  "key2": "Version Palawan",  
  "key3": "Version Manila"  
}
```

**Configure test event** [X]

A test event is a JSON object that mocks the structure of requests emitted by AWS services to invoke a Lambda function. Use it to see the function's invocation result.

To invoke your function without saving an event, configure the JSON event, then choose Test.

Test event action

☒ Create new event ☐ Edit saved event

Event name

VersionTest

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

☒ Private  
This event is only available in the Lambda console and to the event creator. You can configure a total of 10. [Learn more](#)

☐ Shareable  
This event is available to IAM users within the same account who have permissions to access and use shareable events. [Learn more](#)

Template - optional

hello-world

**Event JSON** [Format JSON]

```
1 {  
2   "key1": "Version Boracay",  
3   "key2": "Version Palawan",  
4   "key3": "Version Manila"  
5 }
```

Cancel Invoke Save

- Click on Save

4. Now, click on **Test**. The current response should be:

index.mjs × Environment Vari × Execution result × +

▼ Execution results Status: **Succeeded** Max memory used: 67 MB Time: 4.79 ms

**Test Event Name**  
VersionTest

**Response**  
"Version Boracay"

**Function Logs**

```
2024-08-13T08:53:57.099Z undefined INFO Loading function
START RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c Version: $LATEST
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value1 = Version Borac
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value2 = Version Palaw
2024-08-13T08:53:57.104Z 4c3f897d-00c0-49e0-8d51-06e715be390c INFO value3 = Version Manil
END RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c
REPORT RequestId: 4c3f897d-00c0-49e0-8d51-06e715be390c Duration: 4.79 ms Billed Duration: 5 ms
```

**Request ID**  
4c3f897d-00c0-49e0-8d51-06e715be390c

## Create a New Version

### 1. Modify the Function Code:

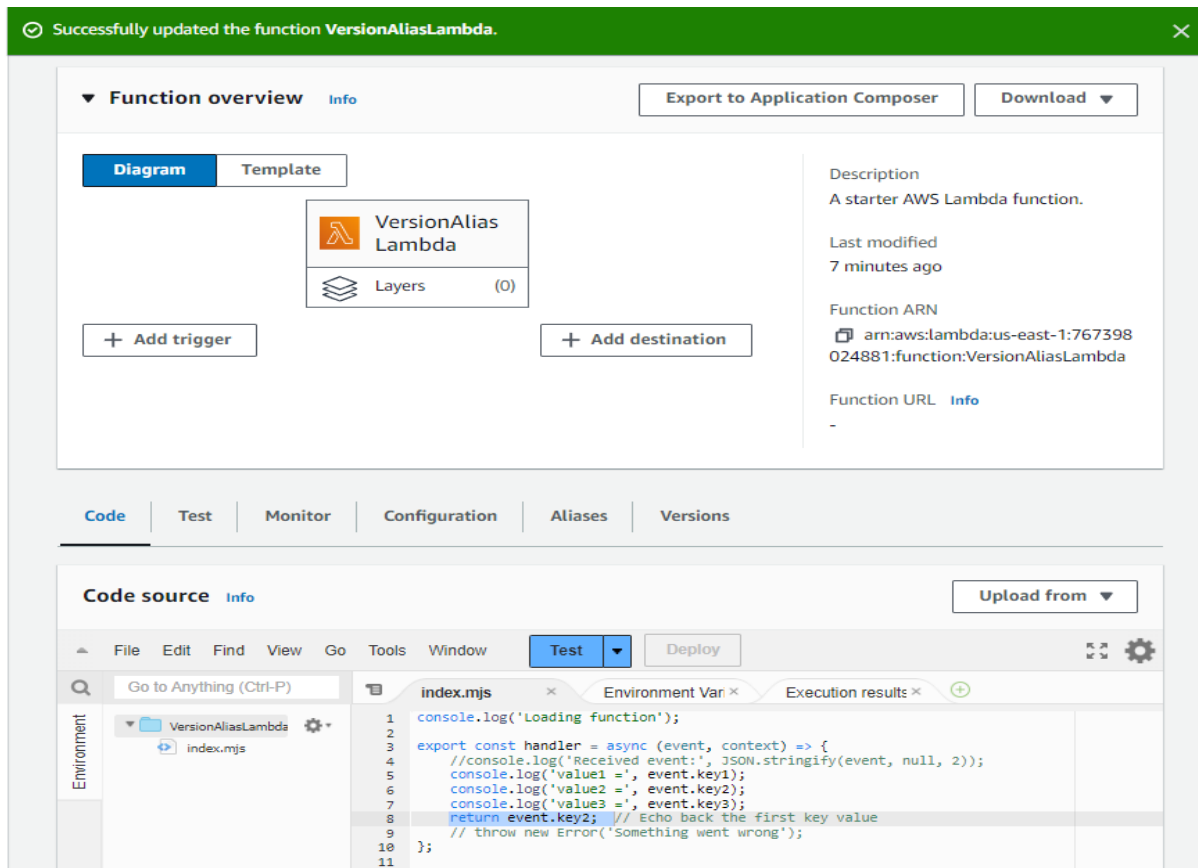
- Scroll down to the **Code source** section.

**Note:** We are using the **old console editor** for this lab. You can switch to the **new or old editor** as you desire; the process remains the same, but the interface may look slightly different.

- Locate **Line 8** in the code and modify it to:

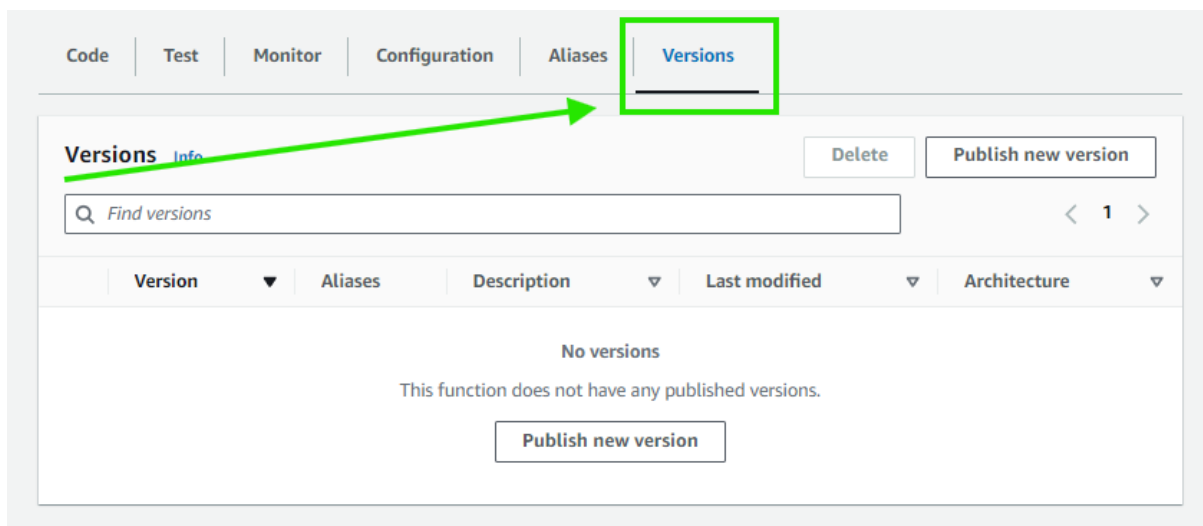
```
return event.key2;
```

- Click on **Deploy** to apply the changes.



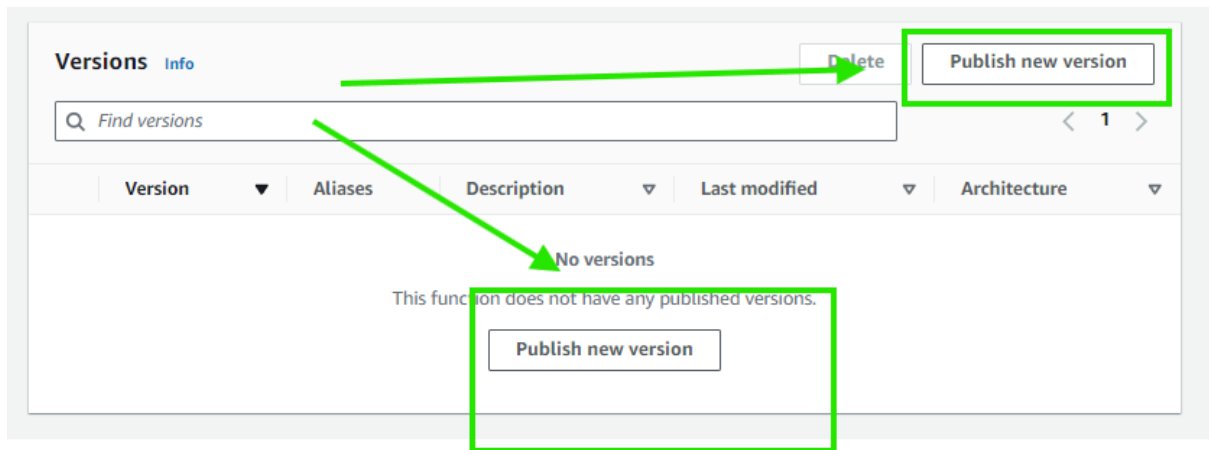
## 2. Publish a New Version:

- After deploying the changes, navigate to the Versions Tab:

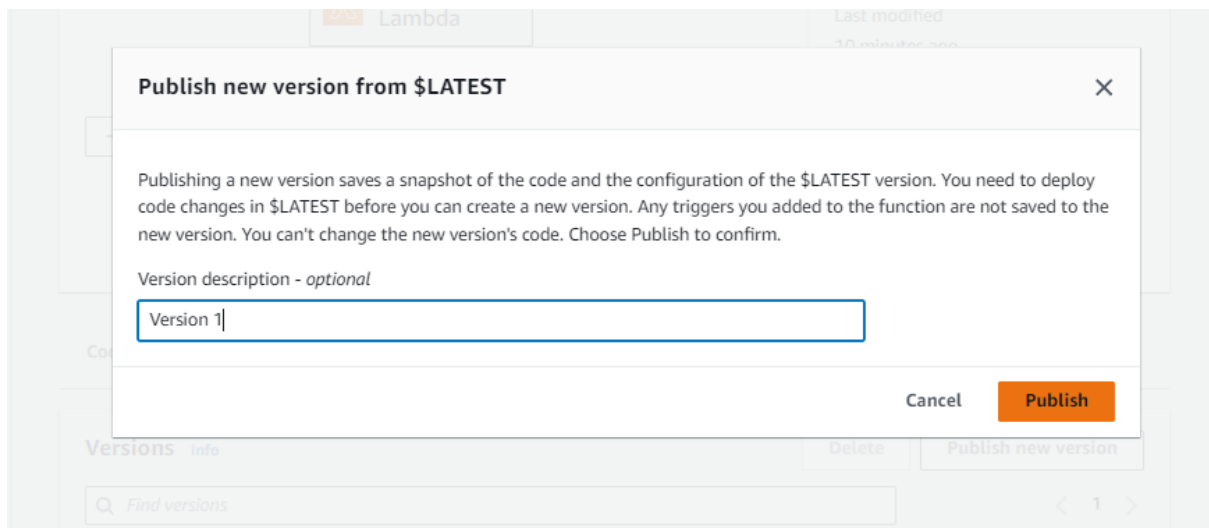


- Click on either of the two **Publish new version**

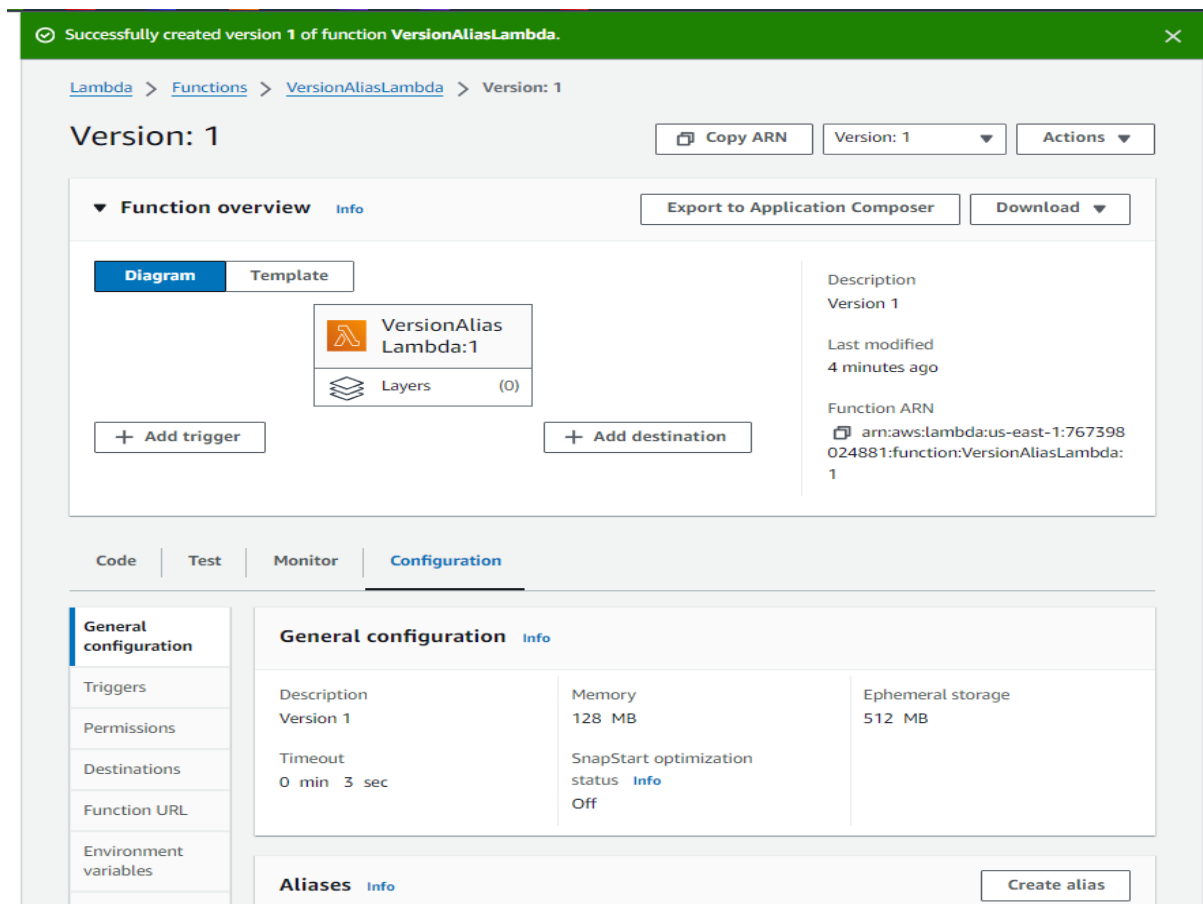




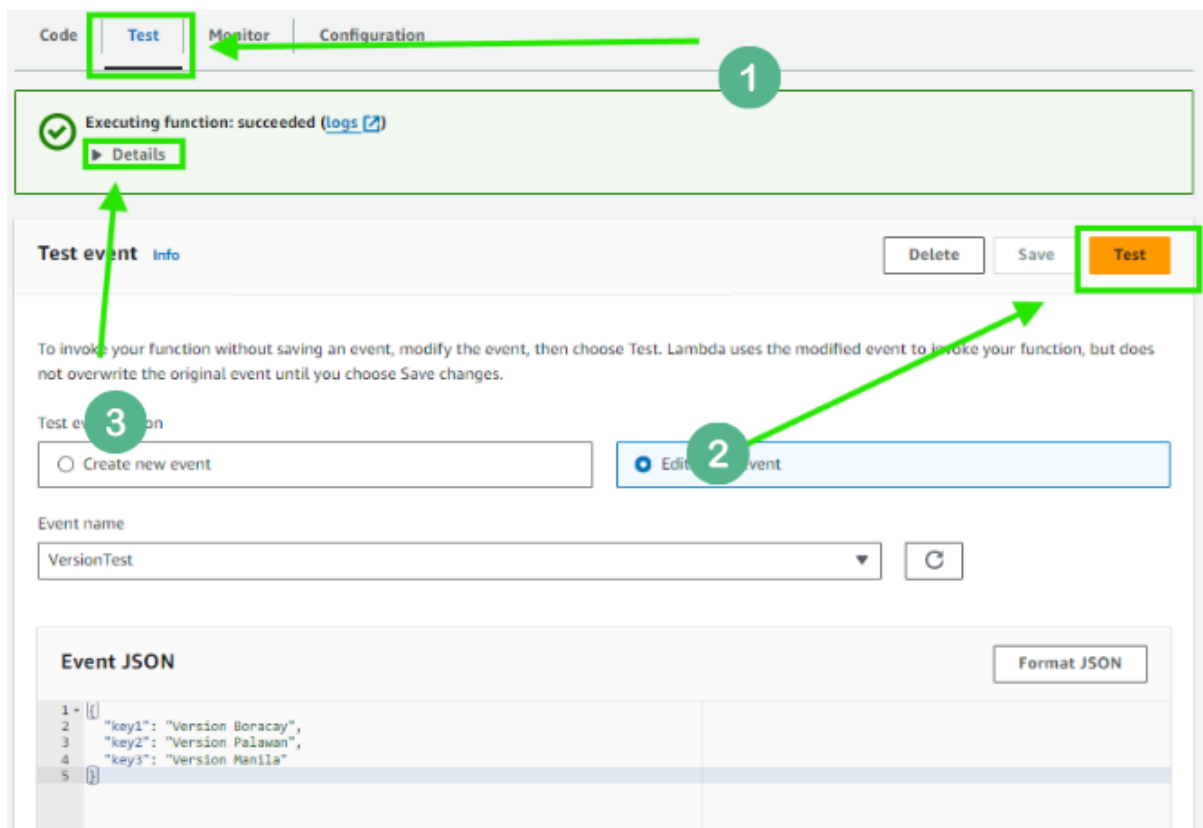
- Add an optional description (e.g., Version 1) and click **Publish**.



3. You will be redirected to this page:



4. Navigate to the **Test Tab**, click on **Test**, and click the Dropdown **Details**



✓

Executing function: succeeded (logs [🔗](#))

▼ Details

The area below shows the last 4 KB of the execution log.

"Version Palawan"

←

Summary

Code SHA-256

A121Z92OGPLg9R5P4mCAw3VPPMmSAethbEgk0F2Cunc=

Request ID

be675167-a741-4ade-9c44-5931d0d77bcc

Init duration

184.30 ms

Billed duration

13 ms

Max memory used

67 MB

Execution time

3 minutes ago (August 13, 2024 at 04:59 PM GMT+8)

Function version

1

Duration

12.41 ms

Resources configured

128 MB

Log output

The section below shows the logging calls in your code. [Click here 🔗](#) to view the corresponding CloudWatch log group.

2024-08-13T08:59:55.973ZundefinedINFOLoading function

START RequestId: be675167-a741-4ade-9c44-5931d0d77bcc Version: 1

2024-08-13T08:59:55.978Zbe675167-a741-4ade-9c44-5931d0d77bccINFOvalue1 = Version Boracay

2024-08-13T08:59:55.978Zbe675167-a741-4ade-9c44-5931d0d77bccINFOvalue2 = Version Palawan

2024-08-13T08:59:55.978Zbe675167-a741-4ade-9c44-5931d0d77bccINFOvalue3 = Version Manila

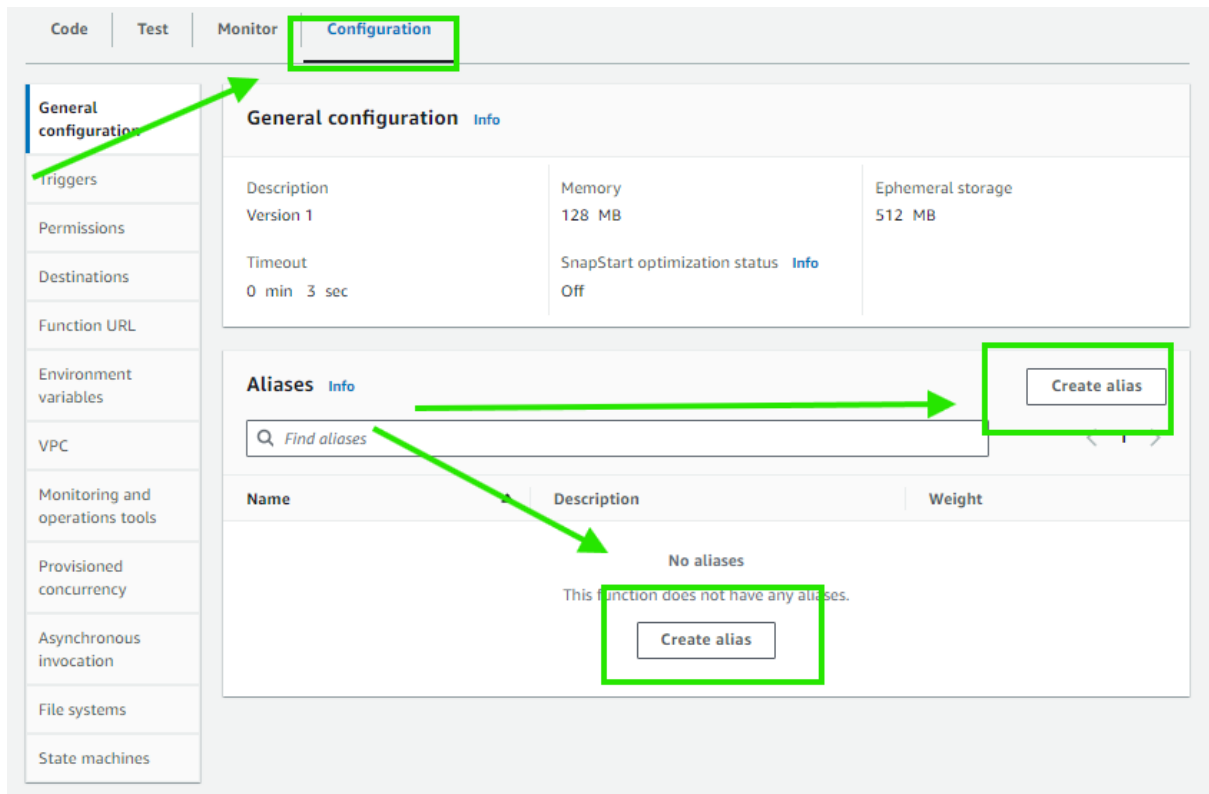
END RequestId: be675167-a741-4ade-9c44-5931d0d77bcc

REPORT RequestId: be675167-a741-4ade-9c44-5931d0d77bcc Duration: 12.41 msBilled Duration: 13 msMemory Size: 128 MBMaxMemory Used: 67 MBInit Duration: 184.30 ms

## Create an Alias

Lambda aliases provide a flexible way to manage function versions. An alias serves as a reference to a particular version of your Lambda function and can be modified to point to different versions. Users interact with the function through the alias's Amazon Resource Name (ARN). When a new version is deployed, you can adjust the alias to direct traffic to this version or balance the load between the existing and new versions.

1. Navigate to the Configuration Tab of your Lambda function's dashboard, and click **Create alias**.



2. Follow the following setting:

- **Name:** Enter a name for the alias (e.g., Dev).
- **Version:** Select the version you just published (e.g., 1).

## Create alias

### Alias configuration

An alias is a pointer to one or two versions. Choose each version that you want the alias to point to.

Name

Description - *optional*

Version

► Weighted alias

Cancel

Save

- Click **Save**.

3. Click on **Test** and observe that the function returns “**Version Palawan**” the same as our previous test in the Create a New Version Step Section



Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

```
"Version Palawan"
```

### Summary

Code SHA-256

A121Z92OGLg9R5P4mCAw3VPPMmSAelhEgk0F2Cunc=

Request ID

7090f974-9d7b-4cac-a3b2-91bebfa1d8c9

Init duration

183.87 ms

Billed duration

4 ms

Max memory used

67 MB

Execution time

1 minute ago (August 13, 2024 at 05:09 PM GMT+8)

Function version

1

Duration

3.36 ms

Resources configured

128 MB

### Log output

The section below shows the logging calls in your code. [Click here](#) to view the corresponding CloudWatch log group.

```
2024-08-13T09:09:47.047Z      undefined      INFO      Loading function
START RequestId: 7090f974-9d7b-4cac-a3b2-91bebfa1d8c9 Version: 1
2024-08-13T09:09:47.053Z      7090f974-9d7b-4cac-a3b2-91bebfa1d8c9      INFO      value1 = Version Boracay
2024-08-13T09:09:47.053Z      7090f974-9d7b-4cac-a3b2-91bebfa1d8c9      INFO      value2 = Version Palawan
2024-08-13T09:09:47.053Z      7090f974-9d7b-4cac-a3b2-91bebfa1d8c9      INFO      value3 = Version Manila
END RequestId: 7090f974-9d7b-4cac-a3b2-91bebfa1d8c9
REPORT RequestId: 7090f974-9d7b-4cac-a3b2-91bebfa1d8c9  Duration: 3.36 ms      Billed Duration: 4 ms      Memory Size: 128 MB      Max
Memory Used: 67 MB      Init Duration: 183.87 ms
```

## Modify the Function and Create a New Version

1. Go back to the **Code source** section.

Lambda > Functions > **VersionAliasLambda** > Alias: Dev

Alias: Dev

Copy ARN


Alias: Dev

Actions

▼ Function overview

Info

+ Add trigger

 VersionAliasLa  
mbda:Dev

+ Add destination

Description


-

Version

1

Additional version

Function ARN

 arn:aws:lambda:us-east-1:76739802488:1:function:VersionAliasLambda:Dev

Function URL

Info

-

Test

Monitor

Configuration

✓

Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

"Version Palawan"

Summary

Code SHA-256

A121Z92OGLg9R5P4mCAw3VPPMmSAelhEgk0F2Cunc=

Request ID

7090f974-9d7b-4cac-a3b2-91bebfa1d8c9

Init duration

183.87 ms

Billed duration

Execution time

2 minutes ago (August 13, 2024 at 05:11 PM GMT+8)

Function version

1

Duration

3.36 ms

Resources configured

Lambda > Functions > VersionAliasLambda

## VersionAliasLambda

Throttle Copy ARN Actions

Function overview Info

Export to Application Composer Download

Diagram Template

VersionAliasLambda

Layers (0)

+ Add trigger + Add destination

Description  
A starter AWS Lambda function.

Last modified  
17 minutes ago

Function ARN  
arn:aws:lambda:us-east-1:767398024881:function:VersionAliasLambda

Function URL Info

Code Test Monitor Configuration Aliases Versions

Test event Info

Delete Save Test

- Change the return statement to:  
return event.key3;
- Click **Deploy** to apply the changes.

☑ Successfully updated the function VersionAliasLambda.

Lambda > Functions > VersionAliasLambda

## 2. Publish the New Version:

- As before, go to the **Versions Tab**, and click on **Publish new version**.

**Note:** You should be able to view the latest version we published.

Code Test Monitor Configuration Aliases Versions

Versions (1) Info

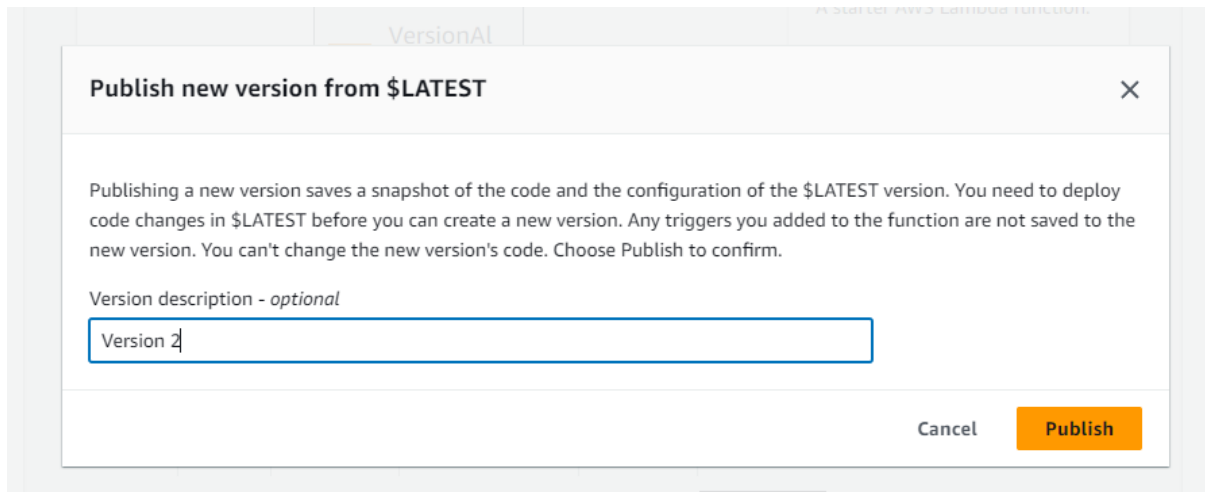
Delete Publish new version

Find versions

	Version	Aliases	Description	Last modified	Architecture
○	1	alias: Dev	Version 1	21 minutes ago	x86_64

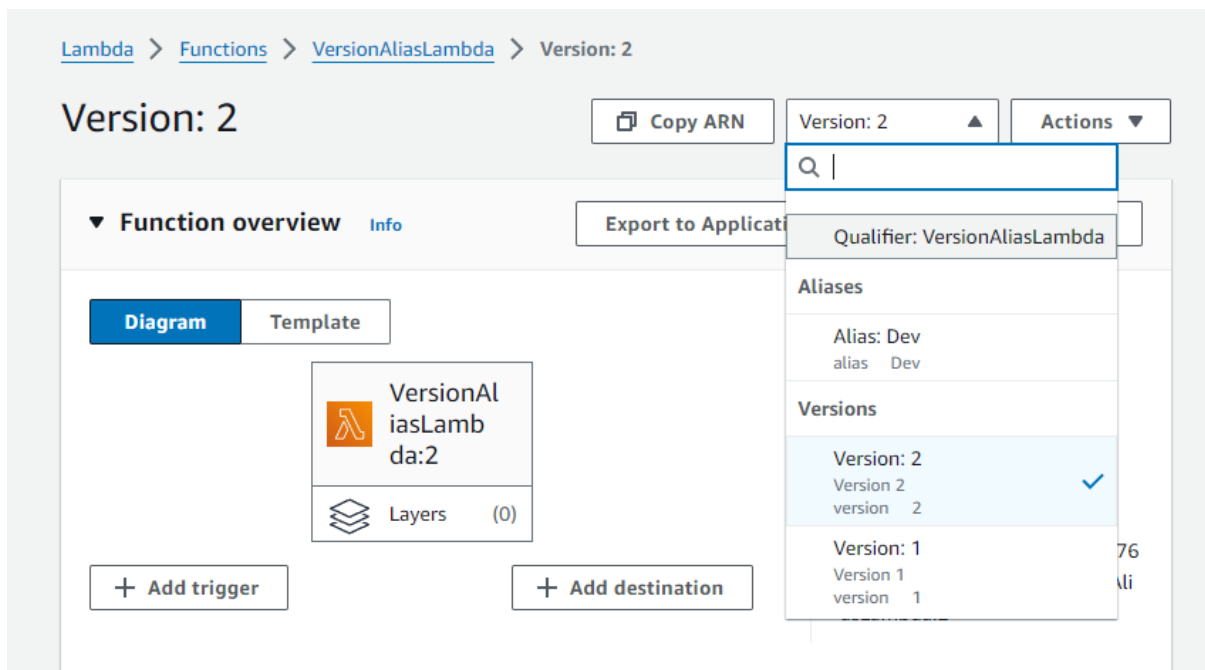


- Describe it as Version 2 and click **Publish**.

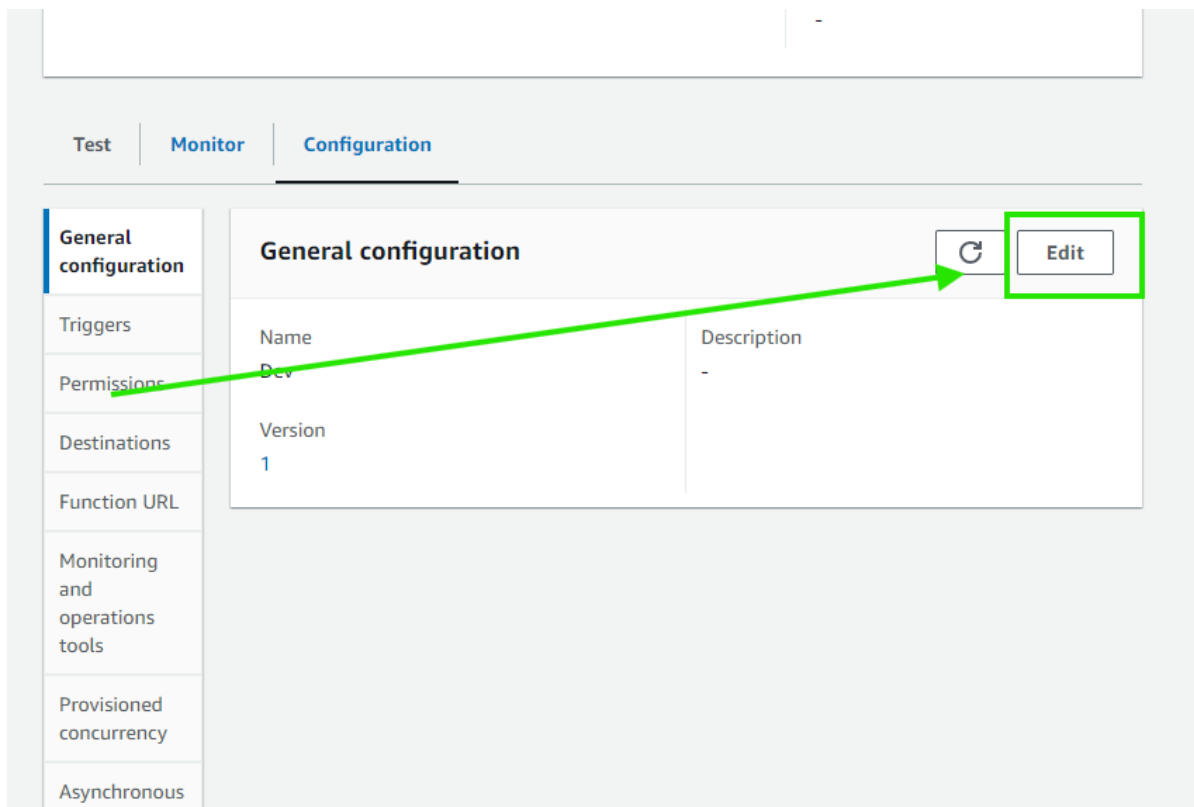


### 3. Update the Alias:

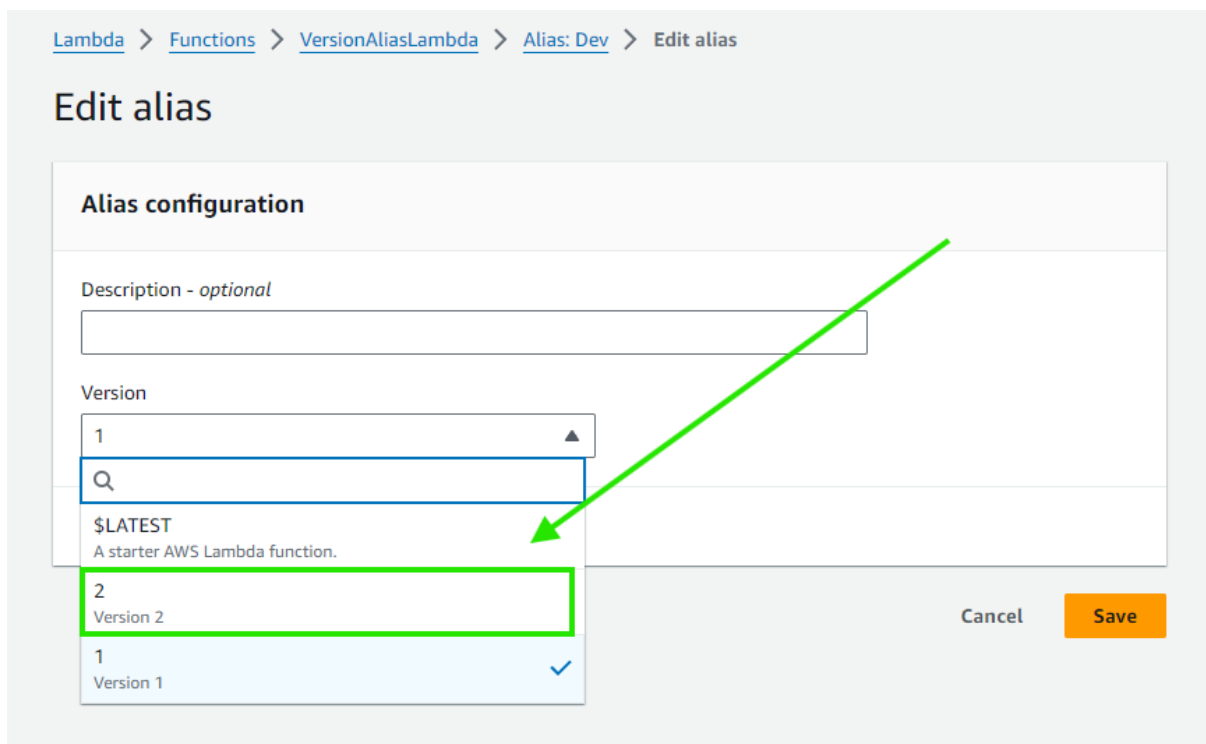
- Scroll up and Click on the **Version: 2 Dropdown**



- Select **Alias: Dev**, scroll down and click on **Edit**



- Change the version it points to from 1 to 2.



- Click **Save** to update the alias.

✔ Your changes have been saved.

Function ARN  
arn:aws:lambda:us-east-1:767398024881:function:VersionAliasLambda:Dev  
Function URL [Info](#)  
-

Test | Monitor | Configuration

General configuration

Triggers  
Permissions  
Destinations  
Function URL  
Monitoring and operations tools  
Provisioned concurrency  
Asynchronous invocation

General configuration

↻

Edit

Name	Description
Dev	-
Version	
2	

4. Click on **Test** and observe that the function returns “**Version Manila**” the same as our previous test in the Create a New Version Step Section

Test | Monitor | Configuration

✓ Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

"Version Manila"

Summary

Code SHA-256	Execution time
A121Z92OGPLg9R5P4mCAw3VPPMmSAelhbgk0F2Cunc=	1 minute ago (August 13, 2024 at 05:24 PM GMT+8)
Request ID	Function version
6a1ef946-d622-4ec0-8d0c-15c62718e123	2
Init duration	Duration
178.25 ms	17.73 ms
Billed duration	Resources configured
18 ms	128 MB

## Implement Weighted Traffic Routing

1. Go to the **Configuration** tab and click on **Edit**

Test | Monitor | Configuration

General configuration

Triggers

Permissions

Destinations

Function URL

Monitoring and operations tools

Provisioned concurrency

Name	Description
Dev	-
Version	
2	

Edit

2. Click on **Weighted alias Dropdown**

[Lambda](#) > [Functions](#) > [VersionAliasLambda](#) > [Alias: Dev](#) > Edit alias

## Edit alias

### Alias configuration

Description - *optional*

Version

#### ▼ Weighted alias

You can shift traffic between two versions, based on weights (%) that you assign. [Learn more](#)

Additional version - *optional*

Cancel

Save

- Set the traffic weight between the two versions (e.g., 50% to Version 2 and 50% to Version 1).

[Lambda](#) > [Functions](#) > [VersionAliasLambda](#) > [Alias: Dev](#) > Edit alias

## Edit alias

### Alias configuration

Description - *optional*

Version

Weight (%)

50

#### ▼ Weighted alias

You can shift traffic between two versions, based on weights (%) that you assign. [Learn more](#)

Additional version - *optional*

Weight (%)

Cancel

Save

- Click **Save**.

✓ Your changes have been saved.
✕

Function ARN

📄 `arn:aws:lambda:us-east-1:767398024881:function:VersionAliasLambda:Dev`

Function URL [Info](#)

-

Test
Monitor
Configuration

General configuration

Triggers

Permissions

Destinations

Function URL

Monitoring and operations tools

Provisioned concurrency

Asynchronous invocation

General configuration

↻
Edit

<p>Name</p> <p>Dev</p> <p>Version</p> <p>2 - Weight: 50%</p>	<p>Description</p> <p>-</p> <p>Additional version</p> <p>1 - Weight: 50%</p>
---	---

### 3. Test the Weighted Routing:

- Use the **Test** button with the same test event as before.



Executing function: succeeded ([logs](#))

▼ Details

The area below shows the last 4 KB of the execution log.

```
"Version Manila"
```

### Summary

Code SHA-256

A121Z92OGPLg9R5P4mCAw3VPPMmSAelhbgk0F2Cunc=

Request ID

4247bd93-f109-4794-835e-680f489ff8da

Duration

115.31 ms

Resources configured

128 MB

Execution time

11 seconds ago (August 13, 2024 at 05:29 PM GMT+8)

Function version

2

Billed duration

116 ms

Max memory used



68 MB


### Log output

The section below shows the logging calls in your code. [Click here](#) to view the corresponding CloudWatch log group.

```
START RequestId: 4247bd93-f109-4794-835e-680f489ff8da Version: 2
2024-08-13T09:29:30.267Z      4247bd93-f109-4794-835e-680f489ff8da    INFO    value1 = Version
Boracay
2024-08-13T09:29:30.267Z      4247bd93-f109-4794-835e-680f489ff8da    INFO    value2 = Version
Palawan
2024-08-13T09:29:30.267Z      4247bd93-f109-4794-835e-680f489ff8da    INFO    value3 = Version
Manila
```

- Run the test multiple times to observe that the responses are split according to the traffic weight.

 **Executing function: succeeded** ([logs](#) )

 Details


The area below shows the last 4 KB of the execution log.

`"Version Palawan"`

### Summary

Code SHA-256	Execution time
A121Z92OGPLg9R5P4mCAw3VPPMmSAelhbgk0F2Cunc=	1 second ago (August 13, 2024 at 05:30 PM GMT+8)
Request ID	Function version
773cf72a-13e1-4d8d-813b-5bfd1bf0a136	1
Init duration	Duration
196.81 ms	2.95 ms
Billed duration	Resources configured
3 ms	128 MB
Max memory used	
67 MB	

### Log output

The section below shows the logging calls in your code. [Click here](#)  to view the corresponding CloudWatch log group.

That's it! Congratulations! You just learned how to manage versions and aliases in AWS Lambda, including how to create new versions, set up aliases, and implement weighted traffic routing.

This lab is an introduction to using versions and aliases in AWS Lambda, providing you with essential skills for managing function deployments and testing different versions in a controlled manner. As you continue, you can explore more advanced versioning strategies and alias configurations to further enhance your serverless applications. Happy Learning!