

# Lambda Function Creation using AWS

## Pre-requisites:

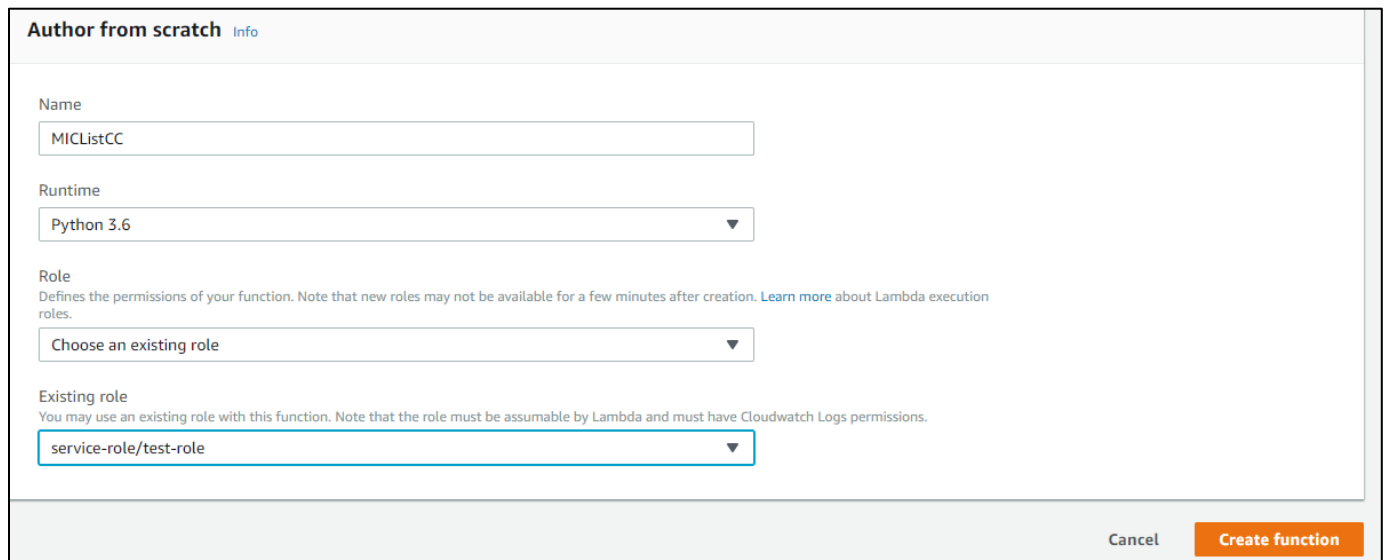
- ZIP file containing code components
- S3 bucket where data can be transferred and upload Zip files

## Libraries used:

- Pandas
- Numpy
- Requests
- JSON
- Boto3

## Steps:

- Create a new Function available under Lambda Service. Fill the required data in the field.
  - o Name: MyListCC
  - o Runtime: Python 3.6
  - o Role: service-role/test-role – Create log files and full access of S3 bucket.



The screenshot shows the 'Author from scratch' form in the AWS Lambda console. The form is titled 'Author from scratch' with an 'Info' link. It contains the following fields:

- Name:** A text input field containing 'MICListCC'.
- Runtime:** A dropdown menu set to 'Python 3.6'.
- Role:** A dropdown menu with the text 'Choose an existing role'. Below it is a note: '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.'
- Existing role:** A dropdown menu set to 'service-role/test-role'. Below it is a note: 'You may use an existing role with this function. Note that the role must be assumable by Lambda and must have Cloudwatch Logs permissions.'

At the bottom right of the form, there are two buttons: 'Cancel' and 'Create function'.

- Since there are multiple dependencies, all the required code components should be zipped to a single folder. Upload zip file to a S3 bucket and feed the link to the Function Code. Update the handler as **'MICListCC.lambda\_handler'**. The file name is MICListCC and the function name is lambda\_handler.

**Function code** Info

**i** The deployment package of your Lambda function "MICListCC" is too large to enable inline code editing. However, you can still invoke your function right now.

Code entry type: Upload a file from Amazon S3 ▼

Runtime: Python 3.6 ▼

Handler Info: MICListCC.lambda\_handler

S3 link URL\*  
Paste an S3 link URL to your function code .ZIP.  
https://s3.us-east-2.amazonaws.com/test-bucket1994/MICListCC.zip

- Create a test event – Since there are no run time parameters, create an empty test event.

**Configure test event**

A function can have up to 10 test events. The events are persisted so you can switch to another computer or web browser and test your function with the same events.

Create new test event  
 Edit saved test events

Event template: Hello World ▼

Event name: MICListCCTest

```
1 {  
2 }
```

- Test the function and check for the execution result. If the program returns 'File uploaded successfully', then the file would be uploaded to S3. Any error can be viewed in the logs.

Execution result: succeeded (logs)

▼ Details

The area below shows the result returned by your function execution.

```
"File uploaded successfully"
```

**Summary**

<b>Code SHA-256</b>	rMDv2bCVglKuwcgfl7/kyNNyh/yWanLzlVouHHK5RM=	<b>Request ID</b>	dec5dacb-3fe1-11e8-b3d8-0bc3969bfd76
<b>Duration</b>	6139.57 ms	<b>Billed duration</b>	6200 ms
<b>Resources configured</b>	128 MB	<b>Max memory used</b>	68 MB

**Log output**

The area below shows the logging calls in your code. These correspond to a single row within the CloudWatch log group corresponding to this Lambda function. [Click here](#) to view the CloudWatch log group.

```
START RequestId: dec5dacb-3fe1-11e8-b3d8-0bc3969bfd76 Version: $LATEST
END RequestId: dec5dacb-3fe1-11e8-b3d8-0bc3969bfd76
REPORT RequestId: dec5dacb-3fe1-11e8-b3d8-0bc3969bfd76 Duration: 6139.57 ms Billed Duration: 6200 ms Memory Size: 128 MB Max Memory Used: 68 MB
```

- Navigate to target Bucket to check for the .JSON file created.

Amazon S3 > bucket-miclistcc

Overview Properties Permissions Management

🔍 Type a prefix and press Enter to search. Press ESC to clear.

Upload Create folder More

US East (Ohio) ↻

Viewing 1 to 1

<input type="checkbox"/>	Name ↑	Last modified ↑	Size ↑	Storage class ↑
<input type="checkbox"/>	MICListCC.json	Apr 14, 2018 5:05:31 PM GMT+0530	300.6 KB	Standard

Viewing 1 to 1