
Run Lambda Documentation

Release 0.1.7.2

Ethan Koenig

Sep 10, 2017

Contents

1	Overview	3
1.1	Doesn't something like this already exist?	3
1.2	Features	3
1.3	Installation	3
2	Context Objects	5
2.1	MockLambdaContext class	5
2.2	MockLambdaContext.Builder class	6
2.3	MockCognitoIdentity class	8
2.4	MockClientContext class	8
2.5	MockClientContext.Client class	8
3	Running Lambda Functions	11
3.1	LambdaResult class	11
3.2	LambdaCallSummary class	12
4	Command Line Interface	13
4.1	Context JSON	13
5	Examples	15
5.1	Example Unit Test	15
6	Indices and tables	17

Contents:

`run_lambda` is a Python package for running Python [AWS Lambda](#) functions locally. It offers a Python module for automated testing of Lambda functions, as well as a command-line interface for ad-hoc local invocations.

Doesn't something like this already exist?

Not exactly. There are other programs for locally running Python Lambda functions. However, all of the other utilities (that I know of) only provide a command-line tool. A command-line tool is great for quick manual invocations. However, if you want to create robust, automated tests for your Lambda functions, a Python module that you can import and call is more appropriate. `run_lambda` is unique because it offers both a simple command-line tool for manual invocations, and an importable Python module for automated tests.

Features

`run_lambda` supports

- An interface for examining the result (return value, exception, timeout) of a function call
- A full implementation of AWS Context objects
- Function calls with or without a timeout
- Resource usage profiling (memory and run-time)
- Convenient mocking of objects and services inside Lambda functions

Installation

The easiest way to install is via `pip`:

```
$ pip install run_lambda
```

You can also download the source from [Github](#).

The `run_lambda` module provides classes for mocking AWS Lambda Context objects. See [here](#) for more information about AWS Lambda Context objects.

MockLambdaContext class

class `MockLambdaContext`

aws_request_id

Property AWS request id associated with invocation

Return type str

client_context

Property Information about client application and device when invoked via AWS Mobile SDK.
May be None.

Return type *MockClientContext*

function_name

Property Name of function

Type str

function_version

Property version of Lambda function that is executing

Return type str

get_remaining_time_in_millis()

Returns remaining execution time, in milliseconds. Should be called inside of Lambda function.

Returns Remaining execution time, in milliseconds

Return type int

identity

Property Cognito identity provider. May be None.

Return type *MockCognitoIdentity*

invoked_function_arn

Property ARN used to invoke function

Return type str

log_group_name

Property Name of CloudWatch log group where logs are written

Return type str

log_stream_name

Property Name of CloudWatch log stream where logs are written

Return type str

memory_limit_in_mb

Property Memory limit, in MB, as a string

Return type str

MockLambdaContext.Builder class

Using the *MockLambdaContext.Builder* class to construct *MockLambdaContext* instances is strongly encouraged.

class `MockLambdaContext.Builder`

__init__ ()

Initializes each context field to a reasonable default

build ()

Constructs and returns a *MockLambdaContext* instance represented by the called builder object.

Returns A newly constructed context object

Return type *MockLambdaContext*

set_aws_request_id (*aws_request_id*)

Parameters **aws_request_id** (*str*) – AWS request id associated with invocation

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_client_context (*client_context*)

Parameters **client_context** (*MockClientContext*) – Information about client application and device

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_default_remaining_time_in_millis (*default_remaining_time_in_millis*)

Sets a default value that will be returned from the *get_remaining_time_in_millis()* method of the built *MockLambdaContext* value if a Lambda function is called without a timeout.

Parameters **default_remaining_time_in_millis** (*int*) – default value

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_function_name (*function_name*)

Parameters **function_name** (*str*) – name of Lambda function

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_function_version (*function_version*)

Parameters **function_version** (*str*) – version of executing Lambda function

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_identity (*identity*)

Parameters **identity** (*MockCognitoIdentity*) – Cognito identity provider

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_invoked_function_arn (*invoked_function_arn*)

Parameters **invoked_function_arn** (*str*) – ARN used to invoke function

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_log_group_name (*log_group_name*)

Parameters **log_group_name** (*str*) – Name of CloudWatch log group where logs are written

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_log_stream_name (*log_stream_name*)

Parameters **log_stream_name** (*str*) – Name of CloudWatch log stream where logs are written

Returns the updated builder

Return type *MockLambdaContext.Builder*

set_memory_limit_in_mb (*memory_limit_in_mb*)

Parameters **memory_limit_in_mb** (*str*) – Memory limit, in megabytes, as a string

Returns the updated builder

Return type *MockLambdaContext.Builder*

MockCognitoIdentity class

class `MockCognitoIdentity`

`__init__` (*identity_id=None, identity_pool_id=None*)

cognito_identity_id

Property AWS Cognito identity id. May be None

Return type str

cognito_identity_pool_id

Property AWS Cognito identity pool id. May be None.

Return type str

MockClientContext class

class `MockClientContext`

`__init__` (*client, custom=None, env=None*)

client

Property Information about client

Return type *MockClientContext.Client*

custom

Property Dictionary of custom values set by client application

Return type dict

env

Property Dictionary of environment information provided by AWS Mobile SDK

Return type dict

MockClientContext.Client class

class `MockClientContext.Client`

`__init__` (*installation_id, app_title, app_version_name, app_version_code, app_package_name*)

app_package_name

Property App package name

Return type str

app_title

Property App title

Return type str

app_version_code

Property App version code

Return type str

app_version_name

Property App version name

Return type str

installation_id

Property Installation id

Return type str

Running Lambda Functions

run_lambda (*handle*, *event*, *context=None*, *timeout_in_seconds=None*, *patches=None*)

Run the Lambda function *handle*, with the specified arguments and parameters.

Parameters

- **handle** (*function*) – Lambda function to call
- **event** (*dict*) – dictionary containing event data
- **context** (*MockLambdaContext*) – context object. If not provided, a default context object will be used.
- **timeout_in_seconds** (*int*) – timeout in seconds. If not provided, the function will be called with no timeout
- **patches** (*dict*) – dictionary of name-to-value mappings that will be patched inside the Lambda function

Returns value returned by Lambda function

Return type *LambdaResult*

LambdaResult class

class LambdaResult (*summary*, *value=None*, *timed_out=False*, *exception=None*)

Represents the result of locally running a Lambda function.

exception

Property The exception raised by the call to the Lambda function, or *None* if no exception was raised

Return type Exception

summary

Property Summary of call to Lambda function

Return type *LambdaCallSummary*

timed_out

Property Whether the call to the Lambda function timed out

Return type bool

value

Property The value returned by the call to the Lambda function, or `None` if no value was returned.

Return type any

LambdaCallSummary class

`class LambdaCallSummary (duration_in_millis, max_memory_used_in_mb, log)`

duration_in_millis

Duration of call, in milliseconds. This value may vary from the duration the call would have taken if actually run in AWS.

Property Duration of call, in milliseconds

Return type int

log

Property The contents of the log for this lambda function.

Return type str

max_memory_used_in_mb

Maximum amount of memory used during call to Lambda function, in megabytes. This value is an estimate of how much memory the call would have used if actually run in AWS. We have found that these estimates are almost always within 5MB of the amount of memory used by corresponding remote calls.

Property Maximum amount of memory used during call to Lambda function, in megabytes.

Return type int

Command Line Interface

The `run_lambda` package also offers a command-line tool for running Lambda functions:

```
$ run_lambda path/to/main.py path/to/event.json
```

Installing the `run_lambda` package from the Python Package Index (i.e. via `pip`) should automatically add the tool to your path. For information on how to use the tool, run `run_lambda --help`:

```
$ run_lambda --help
usage: run_lambda [-h] [-f HANDLER_FUNCTION] [-t TIMEOUT]
                 [-c CONTEXT_FILENAME]
                 filename event

Run AWS Lambda function locally

positional arguments:
  filename              name of file containing Lambda function
  event                name of file containing JSON event data

optional arguments:
  -h, --help           show this help message and exit
  -f HANDLER_FUNCTION, --function HANDLER_FUNCTION
                      Name of handler function. Defaults to "handler"
  -t TIMEOUT, --timeout TIMEOUT
                      Timeout (in seconds) for function call. If not
                      provided, no timeout will be used.
  -c CONTEXT_FILENAME, --context CONTEXT_FILENAME
                      Filename of file containing JSON context data
```

Context JSON

The context JSON data can include the following fields:

```
{
  "aws_request_id": "bf77967d-c53a-5659-9d91-2417e2a3ee58",
  "client_context": {
    "client": {
      "app_package_name": null,
      "app_title": null,
      "app_version_code": null,
      "app_version_name": null,
      "installation_id": null
    },
    "custom": {},
    "env": {}
  },
  "function_name": "my_lambda",
  "function_version": "$LATEST",
  "identity": {
    "cognito_identity_id": null,
    "cognito_identity_pool_id": null
  },
  "invoked_function_arn": "arn:aws:lambda:region-1:813876719243:function:my_lambda",
  "log_group_name": "/aws/lambda/my_lambda",
  "log_stream_name": "2016/12/11/[$LATEST]6ac39f0272c07aa3cd548e6d5a9e8881",
  "memory_limit_in_mb": 128
}
```

Any fields that are not present in the provided context JSON will be populated with default values. Any invalid fields (i.e. any fields other than the ones listed above) are ignored.

To generate a template context data JSON file like the one shown above, use the `run_lambda_context_template` command. For information on how to use the command, run `run_lambda_context_template --help`:

```
$ run_lambda_context_template --help
usage: run_lambda_context_template [-h] [-o OUTPUT_FILENAME]

Generate a template context JSON file

optional arguments:
  -h, --help            show this help message and exit
  -o OUTPUT_FILENAME  output file for template, prints to stdout if omitted
```

Example Unit Test

Suppose we have the following Lambda function in `my_function.py`:

```
import logging
import random

def handler(event, context):
    logger = logging.getLogger()
    logger.info("Log group name: %s", context.log_group_name)
    n = event["number"]
    scale = random.randint(1, 10)
    product = n * scale
    return product
```

We can write a unit test for the function as follows:

```
import mock
import run_lambda
import unittest

import my_function

class MyFunctionTest(unittest.TestCase):
    def test(self):
        log_group_name = "test_log_group_name"
        context = run_lambda.MockLambdaContext.Builder()\
            .set_log_group_name(log_group_name)\
            .build()

        # mock random.randint to always return 5
        patches = {"random.randint": mock.MagicMock(return_value=5)}

        result = run_lambda.run_lambda(my_function.handler,
```

```
        event={"number": 10},
        context=context,
        patches=patches)

    # assert that return value is as expected
    self.assertEqual(result.value, 50)

    # assert that log_group_name was logged
    self.assertIn(log_group_name, result.summary.log)
```

CHAPTER 6

Indices and tables

- `genindex`
- `modindex`
- `search`

Symbols

`__init__()` (MockClientContext method), 8
`__init__()` (MockClientContext.Client method), 8
`__init__()` (MockCognitoIdentity method), 8
`__init__()` (MockLambdaContext.Builder method), 6

A

`app_package_name` (MockClientContext.Client attribute), 8
`app_title` (MockClientContext.Client attribute), 8
`app_version_code` (MockClientContext.Client attribute), 8
`app_version_name` (MockClientContext.Client attribute), 9
`aws_request_id` (MockLambdaContext attribute), 5

B

`build()` (MockLambdaContext.Builder method), 6

C

`client` (MockClientContext attribute), 8
`client_context` (MockLambdaContext attribute), 5
`cognito_identity_id` (MockCognitoIdentity attribute), 8
`cognito_identity_pool_id` (MockCognitoIdentity attribute), 8
`custom` (MockClientContext attribute), 8

D

`duration_in_millis` (LambdaCallSummary attribute), 12

E

`env` (MockClientContext attribute), 8
`exception` (LambdaResult attribute), 11

F

`function_name` (MockLambdaContext attribute), 5
`function_version` (MockLambdaContext attribute), 5

G

`get_remaining_time_in_millis()` (MockLambdaContext method), 5

I

`identity` (MockLambdaContext attribute), 6
`installation_id` (MockClientContext.Client attribute), 9
`invoked_function_arn` (MockLambdaContext attribute), 6

L

`LambdaCallSummary` (class in `run_lambda`), 12
`LambdaResult` (class in `run_lambda`), 11
`log` (LambdaCallSummary attribute), 12
`log_group_name` (MockLambdaContext attribute), 6
`log_stream_name` (MockLambdaContext attribute), 6

M

`max_memory_used_in_mb` (LambdaCallSummary attribute), 12
`memory_limit_in_mb` (MockLambdaContext attribute), 6
`MockClientContext` (class in `run_lambda`), 8
`MockClientContext.Client` (class in `run_lambda`), 8
`MockCognitoIdentity` (class in `run_lambda`), 8
`MockLambdaContext` (class in `run_lambda`), 5
`MockLambdaContext.Builder` (class in `run_lambda`), 6

R

`run_lambda()` (in module `run_lambda`), 11

S

`set_aws_request_id()` (MockLambdaContext.Builder method), 6
`set_client_context()` (MockLambdaContext.Builder method), 6
`set_default_remaining_time_in_millis()` (MockLambdaContext.Builder method), 6
`set_function_name()` (MockLambdaContext.Builder method), 7

set_function_version() (MockLambdaContext.Builder method), 7
set_identity() (MockLambdaContext.Builder method), 7
set_invoked_function_arn() (MockLambdaContext.Builder method), 7
set_log_group_name() (MockLambdaContext.Builder method), 7
set_log_stream_name() (MockLambdaContext.Builder method), 7
set_memory_limit_in_mb() (MockLambdaContext.Builder method), 7
summary (LambdaResult attribute), 11

T

timed_out (LambdaResult attribute), 12

V

value (LambdaResult attribute), 12