

Lambda Layer

AWS Lambda layers allow the Lambda functions to import additional code or data without including that additional code in the deployment package.


A **layer** is a zip file that contains all the additional code. A layer can contain some third-party libraries, custom runtimes, or data (config Files).

Re-use code across multiple functions

Your Lambda code looks simpler by moving reusable code to the layer.

Easy to maintain and maintains consistency in response to the orchestrator tools.

Requirement was multiple lambda functions fetches files from different sources . There has to be common library that can be used to move the fetched data to s3.

```
> Users > k.thulasi.kumar > OneDrive - Accenture > Desktop > ByteByteGo > Lambda Layer > python >  Move
1  def upload_file(file_name, bucket, object_name=None):
2      """Upload a file to an S3 bucket
3
4      :param file_name: File to upload
5      :param bucket: Bucket to upload to
6      :param object_name: S3 object name. If not specified then file_name is used
7      :return: True if file was uploaded, else False
8      """
9
10     # If S3 object_name was not specified, use file_name
11     if object_name is None:
12         object_name = os.path.basename(file_name)
13
14     # Upload the file
15     s3_client = boto3.client('s3')
16     try:
17         response = s3_client.upload_file(file_name, bucket, object_name)
18     except ClientError as e:
19         logging.error(e)
20         return False
21     return True
```

We need first download **requests** from python module.

mkdir python

pip install requests -t .

```
[ec2-user@ip-172-31-80-40 python]$ ls
bin      certifi-2023.5.7.dist-info  charset_normalizer-3.1.0.dist-info  idna-3.4.dist-info  requests-2.31.0.dist-info  urllib3-2.0.3.dist-info
certifi  charset_normalizer         idna                                  requests            urllib3
```

Now add the custom library

```
[ec2-user@ip-172-31-80-40 python]$ ls
movefileS3.py  bin  certifi  charset_normalizer  idna  requests  urllib3
```

Cd ..

zip -r lambda-package.zip python

```
[ec2-user@ip-172-31-80-40 ~]$ ls
lambda-package.zip  python
```

Move the zip file to s3

```
[ec2-user@ip-172-31-80-40 ~]$ aws s3 cp lambda-package.zip s3://aja-s3/
upload: ./lambda-package.zip to s3://aja-s3/lambda-package.zip
```

Publish Lambda Layer.

```
[ec2-user@ip-172-31-80-40 ~]$ aws lambda publish-layer-version \
  --layer-name lambda_layer_MoveFileS3 \
  --description "AWS movefiles3 and requests module" \
  --license-info "MIT" \
  --content S3Bucket=aja-s3,S3Key=lambda-package.zip \
  --compatible-runtimes python3.10
{
  "Content": {
    "Location": "https://prod-04-2014-layers.s3.us-east-1.amazonaws.com/snapshots/82"
```

Lambda > Layers

Layers (2)

Q Filter layers Matches: 1

Move X Clear filters

Name	Version	Compatible runtimes
lambda_layer_MoveFileS3	1	python3.10

Create Lambda function.

Import our custom library and invoke to move the generated file to s3.

lambda_function x Execution results x +

```
1 import json
2 import MovefileS3
3 import os
4 import boto3
5
6 def lambda_handler(event, context):
7     # TODO implement
8     # Save HTML
9     os.chdir('/tmp')
10    filename = 'test.html'
11    save_path = os.path.join(os.getcwd(), filename)
12    print("save_path: ", str(save_path))
13    f = open(save_path, 'w')
14    html_content = """<!DOCTYPE HTML><html><head>
15    <title>Demo HTML</title>
16    <meta name="viewport" content="width=device-width, initial-scale=1">
17    </head><body>
18    <h3>Demo HTML</h3>
19    <br><br>
20    <form action="/get">
21    <br>
22    Name: <input type="text" name="name">
23    <br>
24    Address: <input type="text" name="address">
25    <input type="submit" value="Submit">
26    </form>
27    </body></html>"""
28
29    f.write(html_content)
30    f.close()
31    respon = MovefileS3.upload_file(save_path, 'aja-s3', 'ola.html')
32    return {
33        'statusCode': 200,
34        'body': json.dumps('Hello from Lambda!')}
35
36
```

aja_main

Throttle Copy ARN Actions

▼ Function overview info

aja_main

Layers (1)

+ Add trigger

+ Add destination

Description-

Last modified10 seconds ago

Function ARNarn:aws:lambda:us-east-1:823989251552:function:aja_main

Function URL info

Code

Test


Monitor

Configuration

Aliases

Versions

File moved to S3.

<input type="checkbox"/>	Name ▲	Type ▼	Last modified ▼
<input type="checkbox"/>	 ola.html	html	June 10, 2023, 23:26:29 (UTC+05:30)

Add layer

Function runtime settings

Runtime
Python 3.10

Architecture
x86_64

Choose a layer

Layer source [Info](#)

Choose from layers with a compatible runtime and instruction set architecture or specify the Amazon Resource Name (ARN) of a layer version. You can also [create a new layer](#).

☐ AWS layers

Choose a layer from a list of layers provided by AWS.

☒ Custom layers

Choose a layer from a list of layers created by your AWS account or organization.

☐ Specify an ARN

Specify a layer by providing the ARN.

Custom layers

Layers created by your AWS account or organization that are compatible with your function's runtime.

lambda_layer_MoveFileS3

Version

1