

Aim: Using AWS Flow Framework develop application that includes a simple workflow. Workflow calls an activity to print hello world to the console. It must define the basic usage of AWS Flow Framework, including defining contracts, implementation of activities and workflow coordination logic and worker programs to host them.

Step 1: Open Terminal and Update and Upgrade your system by command

sudo apt-get update && sudo apt-get upgrade

```

root@lab-Vostro-3268: /home/lab# apt-get update && apt-get upgrade
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease
Get:4 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [3,200 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 http://security.ubuntu.com/ubuntu focal-security/main i386 Packages [730 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [954 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/main Translation-en [510 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/restricted amd64 Packages [2,816 kB]
Get:10 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2,821 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-updates/restricted Translation-en [393 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [1,177 kB]
Get:13 http://security.ubuntu.com/ubuntu focal-security/main Translation-en [427 kB]
Get:14 http://in.archive.ubuntu.com/ubuntu focal-updates/universe i386 Packages [781 kB]
Get:15 http://security.ubuntu.com/ubuntu focal-security/restricted amd64 Packages [2,699 kB]
Get:16 http://in.archive.ubuntu.com/ubuntu focal-updates/universe Translation-en [282 kB]
Get:17 http://security.ubuntu.com/ubuntu focal-security/restricted Translation-en [377 kB]
Get:18 http://security.ubuntu.com/ubuntu focal-security/universe i386 Packages [654 kB]
Get:19 http://security.ubuntu.com/ubuntu focal-security/universe amd64 Packages [952 kB]
Get:20 http://security.ubuntu.com/ubuntu focal-security/universe Translation-en [200 kB]
Fetched 19.2 MB in 23s (844 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  gir1.2-goa-1.0 libfwupdplugin1 libxmlb1
Use 'sudo apt autoremove' to remove them.
The following packages will be upgraded:
  bsdutils fdisk firefox firefox-locale-en libblkid1 libcurl3-gnutls libcurl4 libfdisk1 libmount1 libpython2.7 libpython2.7-dev
  libpython2.7-minimal libpython2.7-stdlib libsmartcols1 libuuid1 mount python2.7 python2.7-dev python2.7-minimal python3-update-manager
  rfskill snapd thunderbird thunderbird-gnome-support thunderbird-locale-en thunderbird-locale-en-us update-manager update-manager-core
  update-notifier update-notifier-common util-linux uuid-runtime
10 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 10.5 MB of archives.
After this operation, 10.5 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

Step 2: Download awscli2.zip with command

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscli2.zip"

```

root@lab-Vostro-3268: /home/lab# curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscli2.zip"
  % Total    % Received % Xferd Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 57.5M  100 57.5M    0     0 2866k      0  0:00:20  0:00:20 --:--:-- 4225k
root@lab-Vostro-3268: /home/lab#

```

Step 3: Download awscli2.sig file with command

curl -o awscli2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig

```

root@lab-Vostro-3268: /home/lab# curl -o awscli2.sig https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip.sig
  % Total    % Received % Xferd Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 566  100 566    0     0 765      0  0:00:00  0:00:00 --:--:-- 765
root@lab-Vostro-3268: /home/lab#

```

Step 4: unzip awscliv2.zip with command**unzip awscliv2.zip**

```

root@lab-Vostro-3268:/home/lab# unzip awscliv2.zip
Archive:  awscliv2.zip
  creating: aws/
  creating: aws/dist/
  inflating: aws/THIRD_PARTY_LICENSES
  inflating: aws/install
  inflating: aws/README.md
  creating: aws/dist/awscli/
  creating: aws/dist/cryptography/
  creating: aws/dist/docutils/
  creating: aws/dist/lib-dynload/
  inflating: aws/dist/aws
  inflating: aws/dist/aws_completer
  inflating: aws/dist/libpython3.11.so.1.0
  inflating: aws/dist/_awscli.abi3.so
  inflating: aws/dist/_cffi_backend.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/_ruamel_yaml.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/libz.so.1
  inflating: aws/dist/liblzma.so.0
  inflating: aws/dist/libbz2.so.1
  inflating: aws/dist/libffi.so.5
  inflating: aws/dist/libsqlite3.so.0
  inflating: aws/dist/base_library.zip
  inflating: aws/dist/lib-dynload/_pickle.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_hashlib.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_sha3.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_blake2.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_sha256.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_md5.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_sha1.cpython-311-x86_64-linux-gnu.so
  inflating: aws/dist/lib-dynload/_sha512.cpython-311-x86_64-linux-gnu.so

```

Step 5: Run command**sudo ./aws/install**

```

root@lab-Vostro-3268:/home/lab# sudo ./aws/install
You can now run: /usr/local/bin/aws --version

```

Step 6: Type**command pip3****install aws-sam-cli**

```

root@lab-Vostro-3268:/home/lab# pip3 install aws-sam-cli
Collecting aws-sam-cli
  Downloading aws_sam_cli-1.113.0-py3-none-any.whl (5.9 MB)
    |#####| 5.9 MB 21 kB/s
Collecting aws-lambda-builders==1.47.0
  Downloading aws_lambda_builders-1.47.0-py3-none-any.whl (130 kB)
    |#####| 130 kB 547 kB/s
Collecting tzlocal==5.2
  Downloading tzlocal-5.2-py3-none-any.whl (17 kB)
Collecting dateparser==1.2
  Downloading dateparser-1.2.0-py2.py3-none-any.whl (294 kB)
    |#####| 294 kB 2.4 MB/s
Collecting Flask<3.1
  Downloading flask-3.0.2-py3-none-any.whl (101 kB)
    |#####| 101 kB 447 kB/s
Collecting boto3<2,>=1.29.2
  Downloading boto3-1.34.76-py3-none-any.whl (139 kB)
    |#####| 139 kB 558 kB/s
Collecting pyopenssl~=24.1.0
  Downloading pyOpenSSL-24.1.0-py3-none-any.whl (56 kB)
    |#####| 56 kB 580 kB/s
Collecting requests~=2.31.0
  Using cached requests-2.31.0-py3-none-any.whl (62 kB)
Collecting click~=8.1
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
    |#####| 97 kB 377 kB/s
Collecting watchdog==4.0.0
  Downloading watchdog-4.0.0-py3-none-manylinux2014_x86_64.whl (82 kB)
    |#####| 82 kB 113 kB/s
Collecting chevron==0.12
  Downloading chevron-0.14.0-py3-none-any.whl (11 kB)
Collecting ruamel-yaml==0.18.6
  Downloading ruamel.yaml-0.18.6-py3-none-any.whl (117 kB)
    |#####| 117 kB 363 kB/s
Collecting aws-sam-translator==1.86.0

```

Step 7: Type command sam init in terminal to launch Sam**CLISelect 1st option to use AWS Quick Ttart Templates**

```
root@lab-Vostro-3268:/home/lab# sam init

SAM CLI now collects telemetry to better understand customer needs.

You can OPT OUT and disable telemetry collection by setting the
environment variable SAM_CLI_TELEMETRY=0 in your shell.
Thanks for your help!

Learn More: https://docs.aws.amazon.com/serverless-application-model/latest/developerguide/serverless-sam-telemetry.html

/usr/lib/python3/dist-packages/paramiko/transport.py:220: CryptographyDeprecationWarning: Blowfish has been deprecated and will be removed in
a future release
  "class": algorithms.Blowfish,

You can preselect a particular runtime or package type when using the `sam init` experience.
Call `sam init --help` to learn more.

Which template source would you like to use?
  1 - AWS Quick Start Templates
  2 - Custom Template Location
Choice: 1
```

Step 8: Select Template no.1 Hello World Example

```
Choose an AWS Quick Start application template
 1 - Hello World Example
 2 - Data processing
 3 - Hello World Example with Powertools for AWS Lambda
 4 - Multi-step workflow
 5 - Scheduled task
 6 - Standalone function
 7 - Serverless API
 8 - Infrastructure event management
 9 - Lambda Response Streaming
10 - Serverless Connector Hello World Example
11 - Multi-step workflow with Connectors
12 - GraphQLApi Hello World Example
13 - Full Stack
14 - Lambda EFS example
15 - Hello World Example With Powertools for AWS Lambda
16 - DynamoDB Example
17 - Machine Learning
Template: 1
```

Step 9: Type “N” if it ask to use most popular runtime and package type

Open new terminal by pressing ctrl+shift+T and check for python version by command
python --version

Select the option according to your python version in my case its option 19- python 3.11


```

Use the most popular runtime and package type? (Python and zip) [y/N]: n

Which runtime would you like to use?
 1 - aot.dotnet7 (provided.al2)
 2 - dotnet8
 3 - dotnet6
 4 - go1.x
 5 - go (provided.al2)
 6 - go (provided.al2023)
 7 - graalvm.java11 (provided.al2)
 8 - graalvm.java17 (provided.al2)
 9 - java21
10 - java17
11 - java11
12 - java8.al2
13 - nodejs20.x
14 - nodejs18.x
15 - nodejs16.x
16 - python3.9
17 - python3.8
18 - python3.12
19 - python3.11
20 - python3.10
21 - ruby3.2
22 - rust (provided.al2)
23 - rust (provided.al2023)

Runtime: 17

```

Step 10: Select package type as Zip

```

What package type would you like to use?
 1 - Zip
 2 - Image
Package type: 1

```

Step 11: Now choose Yes option everytime it ask .

Give project name as per your preference in my case its sam-app-test

```

Based on your selections, the only dependency manager available is pip.
We will proceed copying the template using pip.

Would you like to enable X-Ray tracing on the function(s) in your application? [y/N]: y
X-Ray will incur an additional cost. View https://aws.amazon.com/xray/pricing/ for more details

Would you like to enable monitoring using CloudWatch Application Insights?
For more info, please view https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch-application-insights.html [y/N]: y
AppInsights monitoring may incur additional cost. View https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/appinsights-what-is.html
#appinsights-pricing for more details

Would you like to set Structured Logging in JSON format on your Lambda functions? [y/N]: y
Structured logging in JSON format might incur an additional cost. View https://docs.aws.amazon.com/lambda/latest/dg/monitoring-cloudwatchlogs.html#monitoring-cloudwatchlogs-pricing for more details

Project name [san-app]: sam-app-test

Cloning from https://github.com/aws/aws-sam-cli-app-templates (process may take a moment)

-----
Generating application:
-----
Name: sam-app-test
Runtime: python3.8
Architectures: x86_64
Dependency Manager: pip
Application Template: hello-world
Output Directory: .
Configuration file: sam-app-test/samconfig.toml

Next steps can be found in the README file at sam-app-test/README.md

```

Step 12: Now one folder will be created by your provided project name go into that folder by command `cd` (folder name)

After entering the project folder we will invoke the `HelloWorldFunction` by using command `sam local invoke „HelloWorldFunction“`

```
root@lab-Vostro-3268:/home/lab/sam-app-test# sam local invoke 'HelloWorldFunction'
/usr/lib/python3/dist-packages/paramiko/transport.py:220: CryptographyDeprecationWarning: Blowfish has been deprecated and will be removed in a future release
  "class": algorithms.Blowfish,
Invoking app.lambda_handler (python3.8)
Local image was not found.
Removing rapid images for repo public.ecr.aws/sam/emulation-python3.8
Building image.....
.....
Using local image: public.ecr.aws/lambda/python:3.8-rapid-x86_64.

Mounting /home/lab/sam-app-test/hello_world as /var/task:ro,delegated, inside runtime container
START RequestId: 065fed3-5d41-4b87-bb31-8d820fb635e6 Version: $LATEST
END RequestId: 4b9baa5c-2523-443d-b340-f86e2b139f6a
REPORT RequestId: 4b9baa5c-2523-443d-b340-f86e2b139f6a Init Duration: 0.06 ms Duration: 99.11 ms Billed Duration: 100 ms Memory Size: 128 MB Max Memory Used: 128 MB
{"statusCode": 200, "body": "{\"message\": \"hello world\"}"}
root@lab-Vostro-3268:/home/lab/sam-app-test#
```

It should give you `Status Code: 200`

Use command `sudo snap install docker` if docker error occurs.

Step 13: Type command `sam local start-api` this will give you URL open it in any browser.

```
root@lab-Vostro-3268:/home/lab/sam-app-test# sam local start-api
/usr/lib/python3/dist-packages/paramiko/transport.py:220: CryptographyDeprecationWarning: Blowfish has been deprecated and will be removed in a future release
  "class": algorithms.Blowfish,
Initializing the lambda functions containers.
Local image was not found.
Removing rapid images for repo public.ecr.aws/sam/emulation-python3.11
Building image.....
.....
Using local image: public.ecr.aws/lambda/python:3.11-rapid-x86_64.

Mounting /home/lab/sam-app-test/hello_world as /var/task:ro,delegated, inside runtime container
Containers Initialization is done.
Mounting HelloWorldFunction at http://127.0.0.1:3000/hello [GET]
You can now browse to the above endpoints to invoke your functions. You do not need to restart/reload SAM CLI while working on your functions, changes will be reflected instantly/automatically. If you used sam build before running local commands, you will need to re-run sam build for the changes to be picked up. You only need to restart SAM CLI if you update your AWS SAM template
2024-04-03 09:26:18 WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:3000
2024-04-03 09:26:18 Press CTRL+C to quit
Invoking app.lambda_handler (python3.11)
Reuse the created warm container for Lambda function 'HelloWorldFunction'
Lambda function 'HelloWorldFunction' is already running
START RequestId: 0652f01c-89a7-43d9-85fb-3844c8f1a32b Version: $LATEST
END RequestId: b229bc70-c5b6-4bf2-b9a3-97b2cf9340a1
REPORT RequestId: b229bc70-c5b6-4bf2-b9a3-97b2cf9340a1 Init Duration: 0.04 ms Duration: 1153.99 ms Billed Duration: 1154 ms Memory Size: 128 MB Max Memory Used: 128 MB

No Content-Type given. Defaulting to 'application/json'.
2024-04-03 09:26:38 127.0.0.1 - - [03/Apr/2024 09:26:38] "GET /hello HTTP/1.1" 200 -
2024-04-03 09:26:39 127.0.0.1 - - [03/Apr/2024 09:26:39] "GET /favicon.ico HTTP/1.1" 403 -
```

Output:

