# SAM-LOCAL project

1. Configure your local shell for aws.
2. Install Docker (see instructions).
3. Install SAM CLI (see instructions).

Sam cli setup

Update setup tools

1. pip install -U setuptools

2. pip install --upgrade pip

3. bash-3.2$ pip install --user --upgrade aws-sam-cli

1. Create template.yaml

AWSTemplateFormatVersion : '2010-09-09'

Transform: AWS::Serverless-2016-10-31

Resources:

PersonTable:

Type: "AWS::Serverless::SimpleTable"

Person:

Type: "AWS::Serverless::Function"

Properties:

Runtime: python3.6

Handler: src/Person.lambda\_handler

Policies: AmazonDynamoDBFullAccess

Environment:

Variables:

TABLE\_NAME: !Ref PersonTable

Events:

Person:

Type: Api

Properties:

Path: /

Method: post

1. Start local api gw:

On a Mac: sam local start-api --env-vars src/test/resources/test\_environment\_mac.json.

On Windows: sam local start-api --env-vars src/test/resources/test\_environment\_windows.json

On Linux: sam local start-api --env-vars src/test/resources/test\_environment\_linux.json

1. Sample FName world code:

import json

print ('Loading function')

def lambda\_handler(event, context):

# Load the JSON string into an object

personData = json.loads(event['body'])

# Echo back the First Name as success status.

return {'statusCode': 200, 'body': personData['FName']}

1. Start DynamoDB Local:

bash-3.2$ docker run -p 8000:8000 amazon/dynamodb-local

List tables from the command line:

bash-3.2$ aws dynamodb list-tables --endpoint-url http://localhost:8000

{

"TableNames": []

}

1. Create create-person-table.json file as:

{

"TableName": "Person",

"KeySchema": [

{ "AttributeName": "Id", "KeyType": "HASH" },

{ "AttributeName": "FirstName", "KeyType": "RANGE" }

],

"AttributeDefinitions": [

{ "AttributeName": "Id", "AttributeType": "S" },

{ "AttributeName": "FirstName", "AttributeType": "S" }

],

"ProvisionedThroughput": {

"ReadCapacityUnits": 1,

"WriteCapacityUnits": 1

}

}

1. Create Person table in DynamoDB Local:

bash-3.2$ aws dynamodb create-table --cli-input-json file://create-person-table.json --endpoint-url <http://localhost:8000>

[--region us-west-2 if real dynamodb table]

1. Insert Person item in the table (modify code):
2. Testing the code:

curl -d '{"FName": "Mahesh", "LName": "Shankaran", "Age": 36}' http://127.0.0.1:3000/

curl -X GET -d 'f010a134-1a56-4c68-bd84-c3852a7084b5' http://127.0.0.1:3000