**Steps to execute and understand the Testsuite with detailed information**

**Pre-Requisites :**

1. Kindly download this file from the github repo - [InterviewTestQA1.zip](https://github.com/udhaygithub/Udhay_Lorax/blob/main/InterviewTestQA1.zip)
2. Save the file on your desktop and extract it
3. Import this to your VS Code and makesure to have the Nunit libraries

Or please feel free to contact me anytime for a walkthrough (0745 989 2792)

**Detailed information :**

SQS Automation Testing

A screenshot of a computer

Description automatically generated

This is under this folder - SqsConsoleApp

Filename = sqs.cs

Corresponding dependencies can be found in SqsConsoleApp.csproj

I went with setting up a local SQS client option and it worked really well.

Queue Name : Udhay\_demo\_queue1

Queue url : <http://localhost.localstack.cloud:4566/000000000000/Udhay_demo_queue1>

arn:aws:sqs:us-east-1

Please note:

sqs.cs will have 2 sections with comments added.

1. Sending messages to the SQS Queue
2. Validating the messages in the SQS Queue and Deleting from the SQS Queue

Run the first one with the second one commented out. Once sent is successful, Uncomment the second one and comment the first one, to validate and delete the messages from the SQS.

Below is the output of a successful messages posted in the LocalStack desktop :

A screenshot of a computer

Description automatically generated

JSON Automation Testing

JSONTest.cs code is up-to-date

Calculator Automation Testing

CalculatorTest.cs is up-to-date

TestAutomation.cs

*From the file (TestAutomation.cs) given, can you find any problems in the code?*

**Issues:**

1. There are errors due to missing libraries and environment configurations like Lorax.core etc
2. Too many objects created in the constructor.

If any other tests require even a slight variation, constructor must be modified

1. At Line 115 for example, ForEach loop is iterating over LoraxSQS que for sending messages, but doesn't use it or process the result variable in it.

I would simply do this - consoleWriteLin("message sent with status: {Result.Status}");

1. LoraxSQS object is instantiated repeatedly across multiple methods.

This results in code duplication and not a best practice. This will make the test harder to maintain and leads to performance impact due to memory issues.

**Suggestions:**

1. Can use hooks concept to put all the oneTime setup methods under beforeHooks and teardown setup under afterHooks, from Automation standpoint

1. Move the mock setup and logger behavior into a separate method or use a mocking framework configuration that can be reused across tests.

1. Avoid recurrences. For example Objects initialised too many areas that can be minimised.

Thank you !

Happy 2025 !