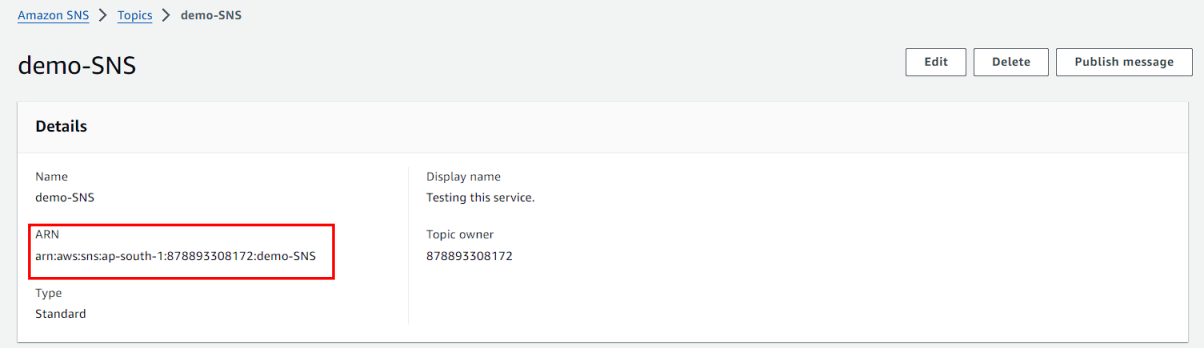
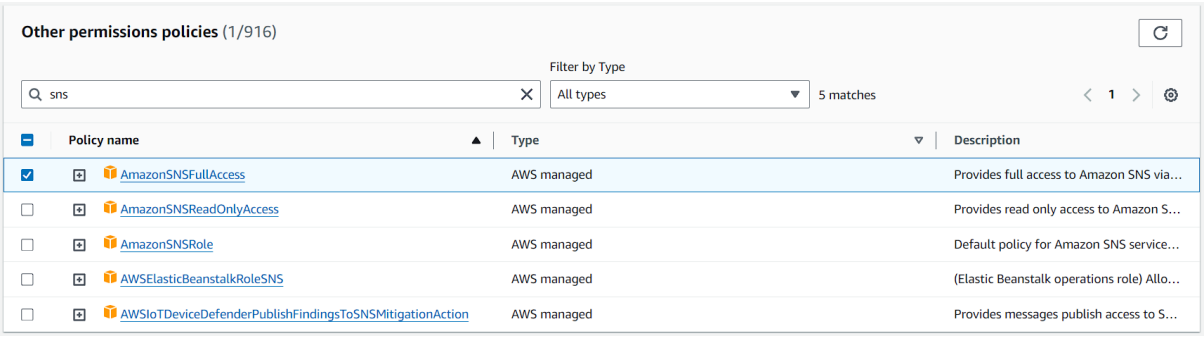
😊 Simple Notification Service

1. Now in this you are going to setup program for SNS.
2. For that again you need to download the zip file from GitHub and unzip it then open it in Visual Studio.
3. For this lab you should be having a topic in SNS then copy its ARN and paste it in visual studio.

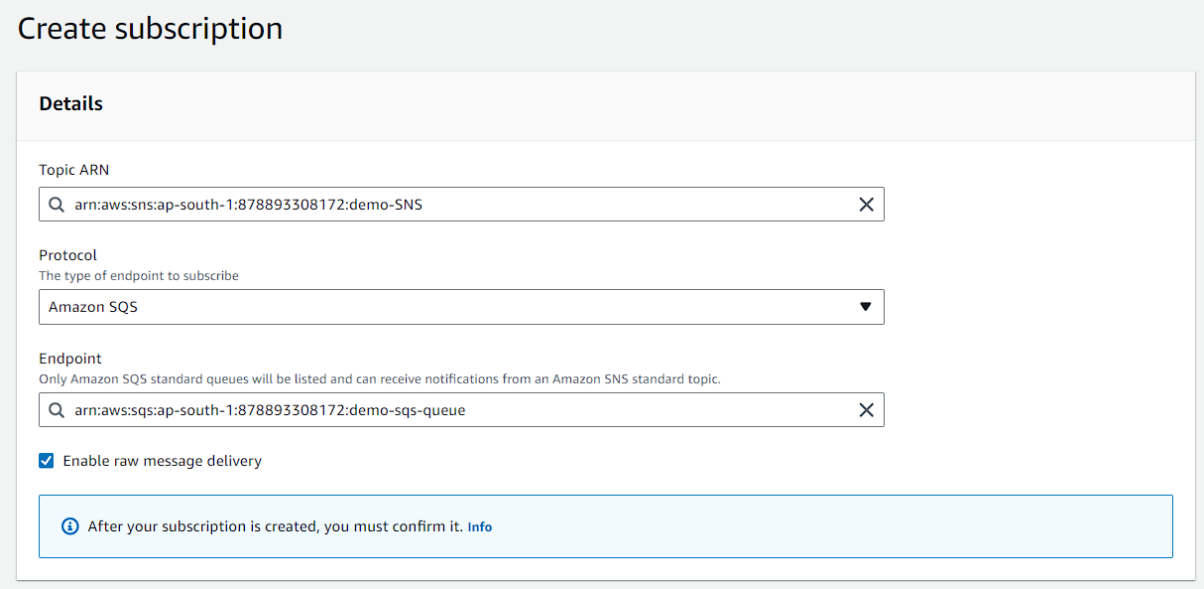



```
using Amazon.SimpleNotificationService;
using Amazon.SimpleNotificationService.Model;
using Newtonsoft.Json;
using SendMessage_SNS;

string topicARN = "arn:aws:sns:ap-south-1:878893308172:demo-SNS";

IAmazonSimpleNotificationService snsClient = new AmazonSimpleNotificationServiceClient();
```

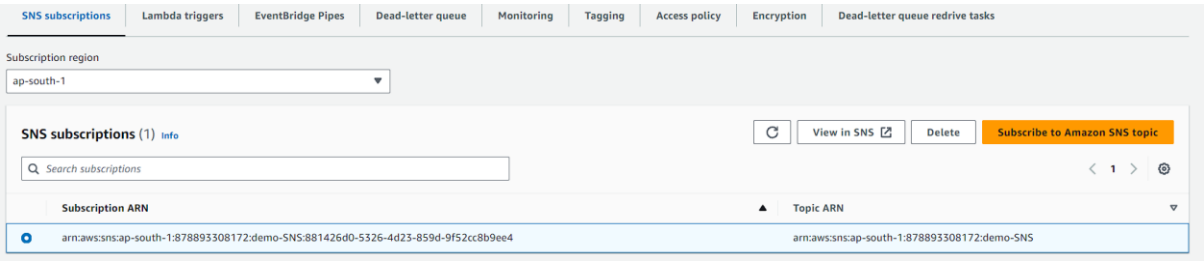
4. Now open your IAM and add policy for SNS in your role which is running in your Lambda function which you used in previous lab for SQS.
5. So, you need to add this permission in both you IAM role and User.



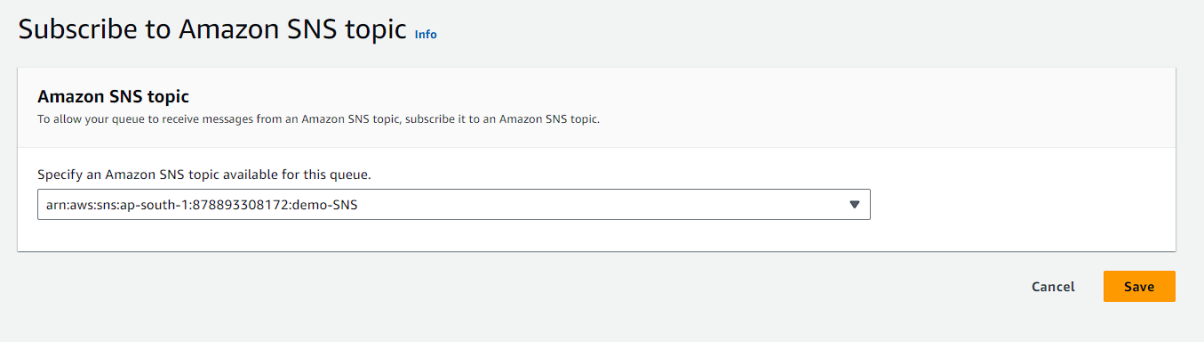
6. No go back to SNS and create a new subscription.
7. Choose protocol as amazon SQS and create your subscription.
8. Here you are subscribing your SQS queue onto your SNS topic.



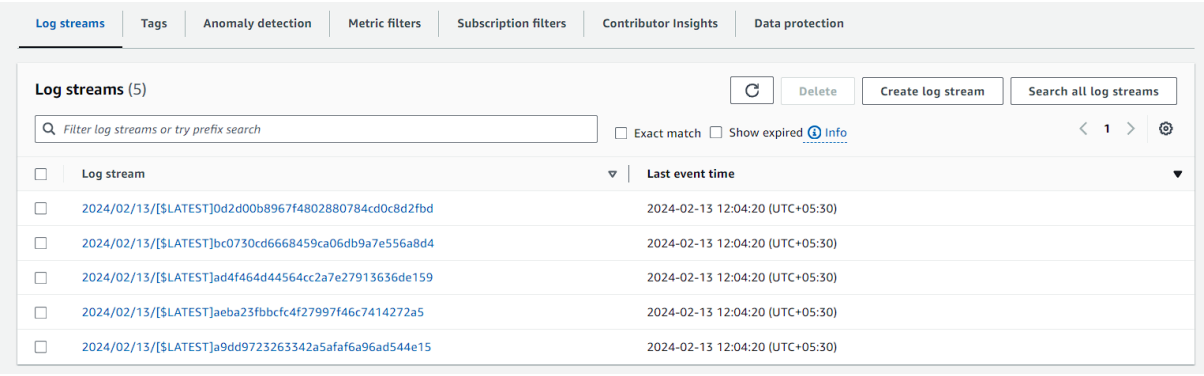
9. Afterwards go to SQS and there scroll down and select your subscription ARN and subscribe to SNS topic.

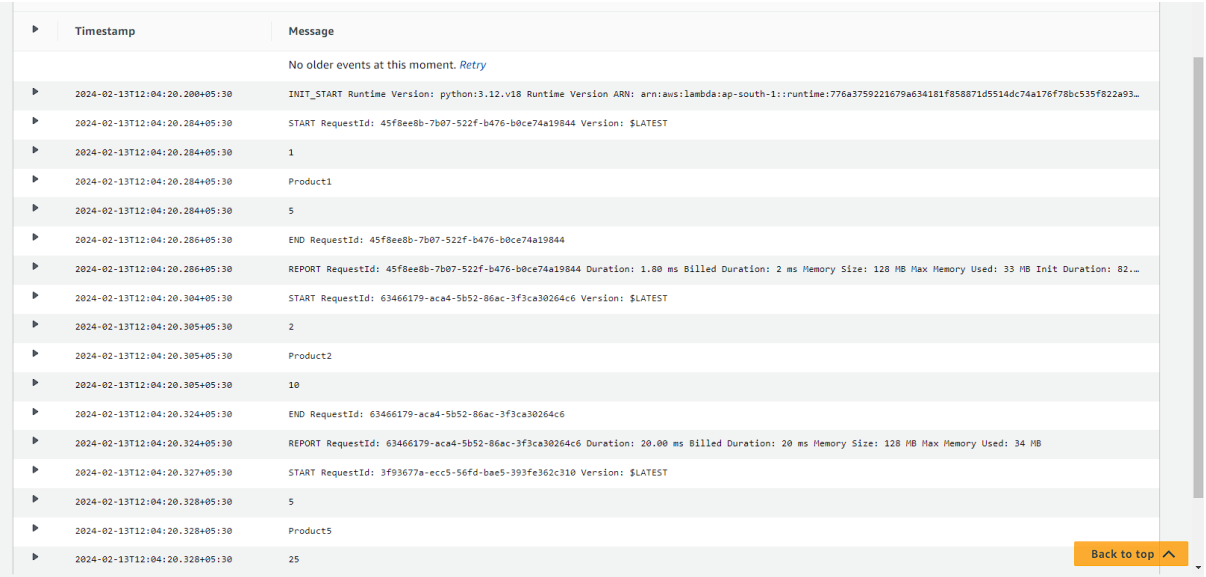


10. Now select your topic and click on Save.



11. Once this is done go to the program and run the program locally.
12. Then go back to AWS Console and navigate to CloudWatch and open your logs for your lambda function.
13. You will see multiple logs and if you will open them, you can see your data in them.


Log streams
Tags
Anomaly detection
Metric filters
Subscription filters
Contributor Insights
Data protection
Log streams (5)
Delete
Create log stream
Search all log streams
Filter log streams or try prefix search
Exact match
Show expired
Info
1
Log stream
Last event time
2024/02/13/[$LATEST]0d2d00b8967f4802880784cd0c8d2fbd
2024-02-13 12:04:20 (UTC+05:30)
2024/02/13/[$LATEST]bc0730cd6668459ca06db9a7e556a8d4
2024-02-13 12:04:20 (UTC+05:30)
2024/02/13/[$LATEST]ad4f464d44564cc2a7e27913636de159
2024-02-13 12:04:20 (UTC+05:30)
2024/02/13/[$LATEST]aeba23fbbcfc4f27997f46c7414272a5
2024-02-13 12:04:20 (UTC+05:30)
2024/02/13/[$LATEST]a9dd9723263342a5afaf6a96ad544e15
2024-02-13 12:04:20 (UTC+05:30)


Timestamp
Message
No older events at this moment. Retry
2024-02-13T12:04:20.200+05:30 INIT_START Runtime Version: python:3.12.v18 Runtime Version ARN: arn:aws:lambda:ap-south-1::runtime:776a3759221679a634181f858871d5514dc74a176f78bc535f822a93…
2024-02-13T12:04:20.284+05:30 START RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844 Version: $LATEST
2024-02-13T12:04:20.284+05:30 1
2024-02-13T12:04:20.284+05:30 Product1
2024-02-13T12:04:20.284+05:30 5
2024-02-13T12:04:20.286+05:30 END RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844
2024-02-13T12:04:20.286+05:30 REPORT RequestId: 45f8ee8b-7b07-522f-b476-b0ce74a19844 Duration: 1.80 ms Billed Duration: 2 ms Memory Size: 128 MB Max Memory Used: 33 MB Init Duration: 82…
2024-02-13T12:04:20.304+05:30 START RequestId: 63466179-aca4-5b52-86ac-3f3ca30264c6 Version: $LATEST
2024-02-13T12:04:20.305+05:30 2
2024-02-13T12:04:20.305+05:30 Product2
2024-02-13T12:04:20.305+05:30 10
2024-02-13T12:04:20.324+05:30 END RequestId: 63466179-aca4-5b52-86ac-3f3ca30264c6
2024-02-13T12:04:20.324+05:30 REPORT RequestId: 63466179-aca4-5b52-86ac-3f3ca30264c6 Duration: 20.00 ms Billed Duration: 20 ms Memory Size: 128 MB Max Memory Used: 34 MB
2024-02-13T12:04:20.327+05:30 START RequestId: 3f93677a-ecc5-56fd-bae5-393fe362c310 Version: $LATEST
2024-02-13T12:04:20.328+05:30 5
2024-02-13T12:04:20.328+05:30 Product5
2024-02-13T12:04:20.328+05:30 25
Back to top