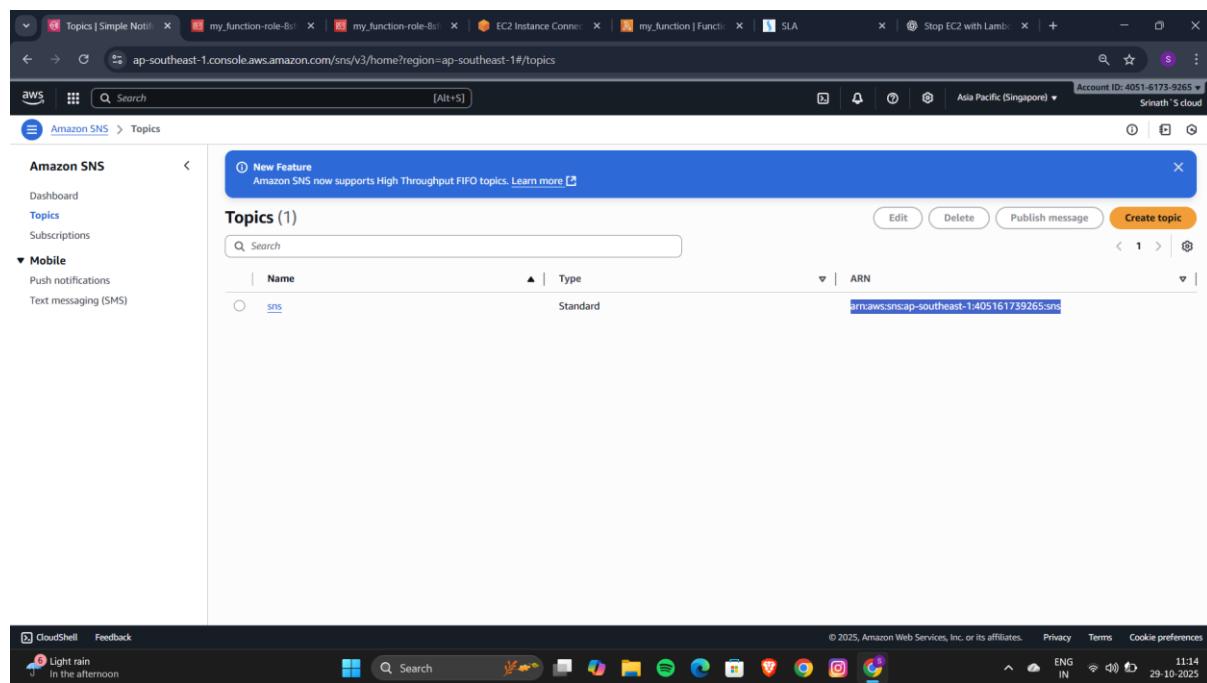


Task 22

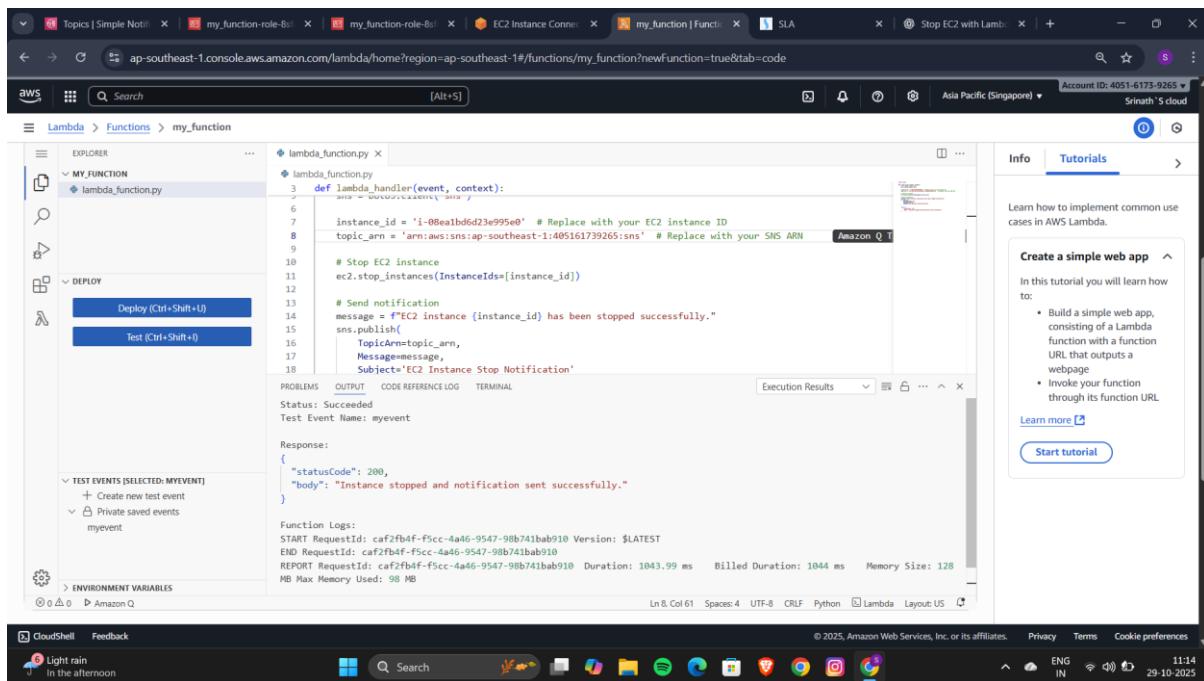
Stop the EC2 instance with help of lambda and get the notification through the cloud watch as instance stopped

Creation of SNS



The screenshot shows the AWS SNS Topics page. The left sidebar has 'Amazon SNS' selected, with options for Dashboard, Topics, Subscriptions, and Mobile (Push notifications, Text messaging (SMS)). The main area shows a 'Topics (1)' section with a table. The table has columns for Name, Type, and ARN. One row is listed: Name is 'sns', Type is 'Standard', and ARN is 'arn:aws:sns:ap-southeast-1:405161739265:sns'. There are buttons for Edit, Delete, Publish message, and Create topic. A blue banner at the top says 'New Feature: Amazon SNS now supports High Throughput FIFO topics. Learn more.' The browser status bar at the bottom shows 'CloudShell Feedback', 'Light rain In the afternoon', and system icons.

Creation of Lambda



The screenshot shows the AWS Lambda console interface. In the left sidebar, under 'Lambda > Functions > my_function', there are sections for 'EXPLORER' (showing 'lambda_function.py'), 'MY FUNCTION' (with a file icon and 'lambda_function.py'), and 'DEPLOY' (with 'Deploy (Ctrl+Shift+U)' and 'Test (Ctrl+Shift+I)' buttons). The main area displays the Python code for the function:

```
def lambda_handler(event, context):
    instance_id = 'i-08ea1bd6d23e995e0' # Replace with your EC2 instance ID
    topic_arn = 'arn:aws:sns:ap-southeast-1:405161739265:sns' # Replace with your SNS ARN

    # Stop EC2 Instance
    ec2.stop_instances(InstanceIds=[instance_id])

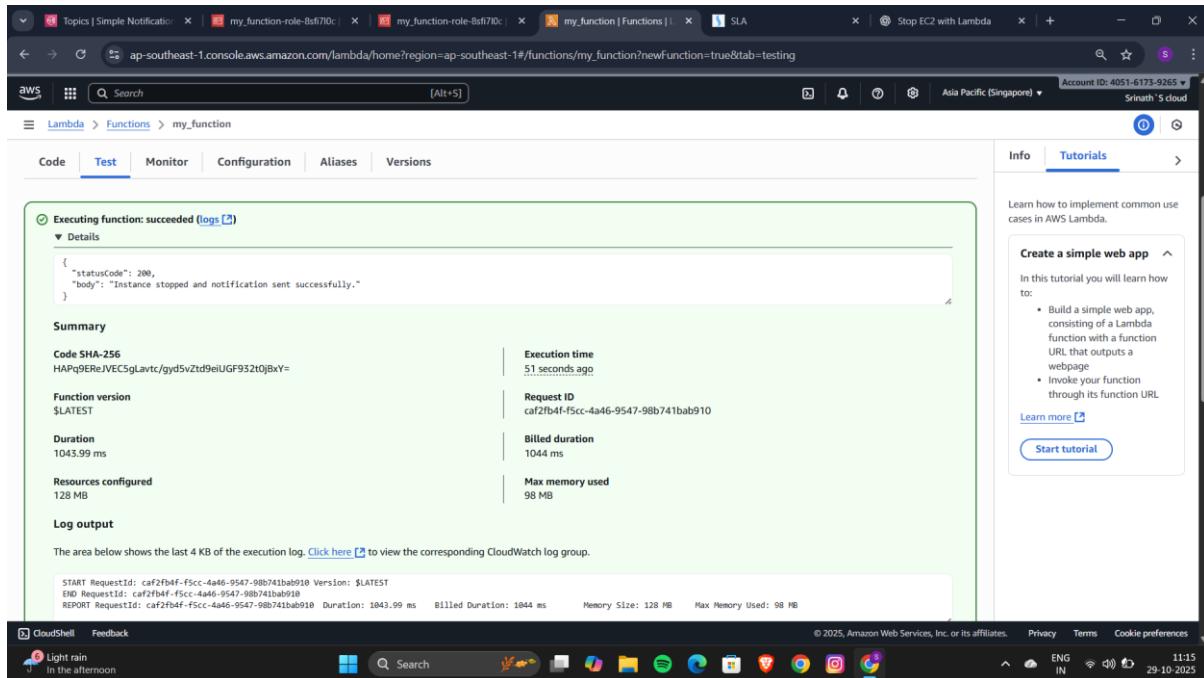
    # Send notification
    message = f'EC2 instance {instance_id} has been stopped successfully.'
    sns.publish(
        TopicArn=topic_arn,
        Message=message,
        Subject='EC2 Instance Stop Notification'
    )
```

The 'TEST EVENTS [SELECTED: MYEVENT]' section shows a single event named 'myevent' selected. The 'Function Logs' section shows the execution details:

```
START RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910 Version: $LATEST
END RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910
REPORT RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910 Duration: 1043.99 ms Billed Duration: 1044 ms Memory Size: 128 MB Max Memory Used: 98 MB
```

The status bar at the bottom indicates 'Status: Succeeded' and 'Test Event Name: myevent'. On the right side, there's a 'Info' tab with a 'Create a simple web app' tutorial.

Output



The screenshot shows the AWS Lambda console interface with the 'Test' tab selected. The 'Logs' section displays the successful execution log:

```
Executing function: succeeded (logs)
```

The log content is:

```
{ "statusCode": 200, "body": "Instance stopped and notification sent successfully." }
```

The 'Summary' section provides execution details:

Code SHA-256	Execution time
HAPq9EREjVECsLgavtc/gyd5vZtd9eiUGF932t0BxY=	51 seconds ago
Function version	Request ID
\$LATEST	caf2fb4f-f5cc-4a46-9547-98b741bab910
Duration	Billed duration
1043.99 ms	1044 ms
Resources configured	Max memory used
128 MB	98 MB

The 'Log output' section shows the last 4 KB of the execution log:

```
START RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910 Version: $LATEST
END RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910
REPORT RequestId: caf2fb4f-f5cc-4a46-9547-98b741bab910 Duration: 1043.99 ms Billed Duration: 1044 ms Memory Size: 128 MB Max Memory Used: 98 MB
```

The status bar at the bottom indicates 'Status: Succeeded' and 'Test Event Name: myevent'. On the right side, there's a 'Info' tab with a 'Create a simple web app' tutorial.

