

Result : [Lambda]

Task 02

The screenshot shows the AWS Lambda console for a function named 'lambda-s3'. The 'Function overview' tab is active, displaying the function's configuration. The 'Layers' section shows '(0)' layers. The 'Triggers' section shows an 'S3' trigger. The 'Destinations' section shows an 'Amazon SNS' destination. The 'Code source' tab is also visible, showing the function's code. The code is a Python script that imports 'json', 'urllib.parse', and 'boto3'. It defines a 'lambda_handler' function that takes 'event' and 'context' as arguments. The handler prints the event and then gets the object, key, and event name from the event to show its content type. The code is as follows:

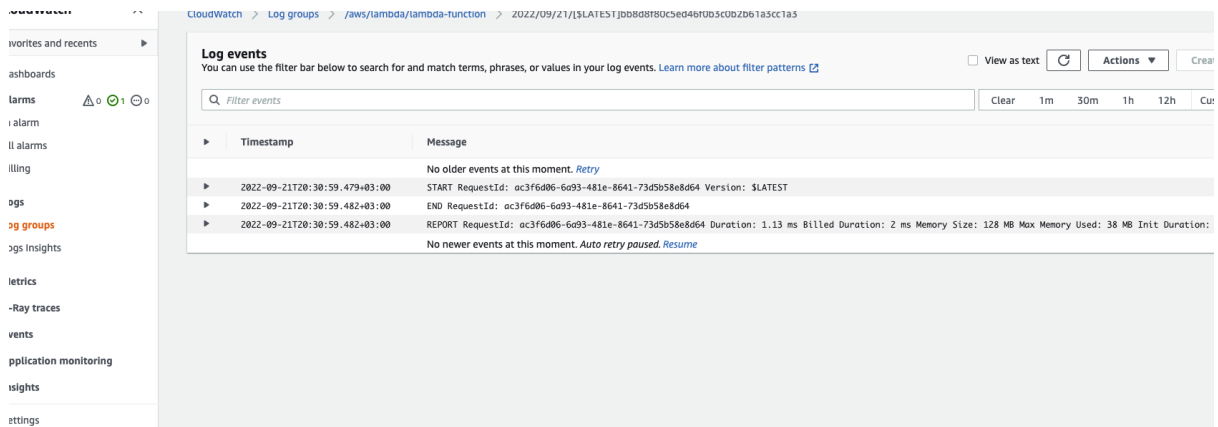
```
1 import json
2 import urllib.parse
3 import boto3
4
5 print('Loading function')
6
7 s3 = boto3.client('s3')
8 sns = boto3.client('sns')
9
10 def lambda_handler(event, context):
11     # Print the event
12     #print("Received event: " + json.dumps(event, indent=2))
13
14     # Get the object, Key and event name from the event and show its content type
15     bucket = event['Records'][0]['s3']['bucket']['name']
```

I've created new role that have a permission to put logs and create logs on cloudwatch :

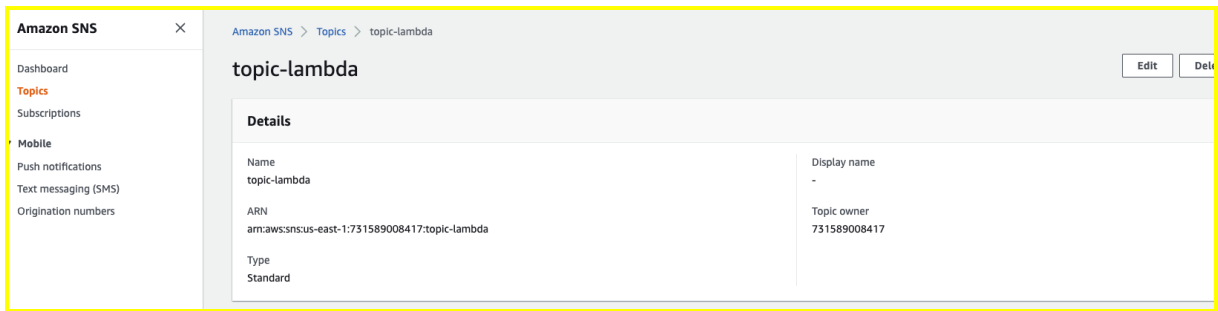
The screenshot shows the AWS IAM console for a role named 'role-lambda'. The 'Summary' tab is active, displaying the role's configuration. The 'Permissions' tab is also visible, showing the role's permissions. The role is named 'role-lambda' and has the description 'Allows Lambda functions to call AWS services on your behalf.' The role was created on September 23, 2022, at 14:40 (UTC+03:00). The role has no last activity and no permissions. The 'Permissions policies' section shows one policy named 'policy_lambda'. The policy is a customer-managed policy. The policy's JSON definition is as follows:

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": [
8         "logs:DisassociateKmsKey",
9         "logs>DeleteSubscriptionFilter",
10        "logs:UntagLogGroup",
```

I can see the logs inside CloudWatch -> Log groups :



Created SNS that publish a message and trigger the Lambda :



Once I've upload a file :

