3. Serverless Data Processing Pipeline

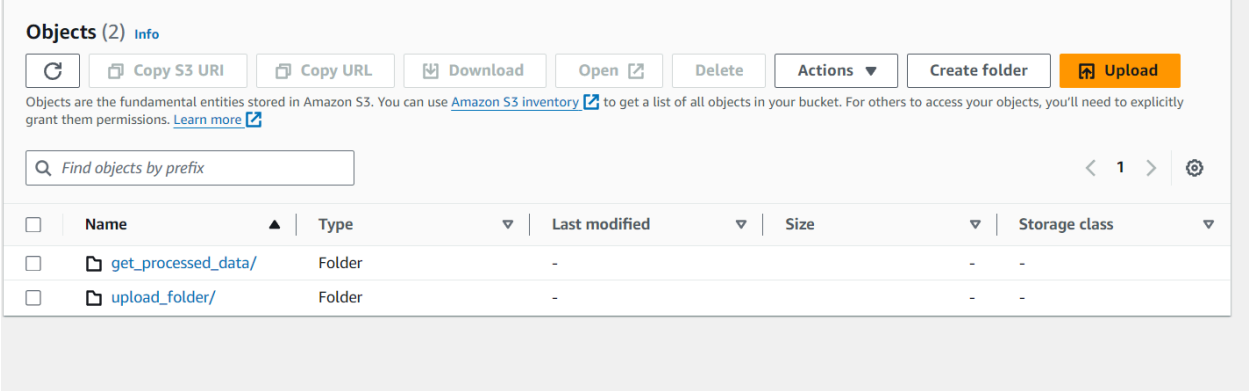
Objective: Build a serverless pipeline for processing data (e.g., log processing or ETL jobs).

Approach:

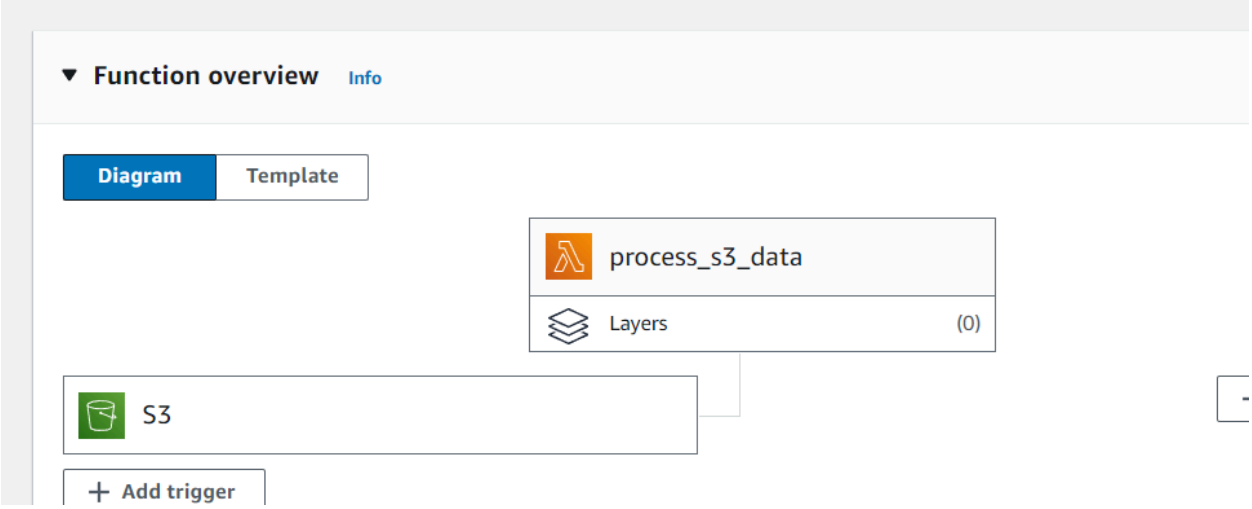
- **Data Ingestion**: Use AWS services like S3 or Kinesis to ingest data.
- **Processing**: Create Lambda functions to process the ingested data.
- **Storage**: Store the processed data in an appropriate AWS service, like S3 or DynamoDB.
- **Monitoring**: Set up CloudWatch to monitor the pipeline's performance and to log any issues.

Goal: Learn to build a serverless data processing pipeline, understanding the flow of data through various AWS services.

Create s3 bucket with two folder source and destination folder:



Create lambda function and add trigger to s3 file upload:



Lambda function code:

```
import boto3
import urllib.parse
import os

def lambda_handler(event, context):
    s3_client = boto3.client('s3')

    # Get bucket name and object key from the S3 event
    bucket_name = event['Records'][0]['s3']['bucket']['name']
    object_key = urllib.parse.unquote_plus(event['Records'][0]['s3']['object']['key'],
encoding='utf-8')

    # Extract the filename from the object key
    filename = os.path.basename(object_key)

    # Define destination key (output folder and file name)
    destination_key = 'get_processed_data/' + filename

    # Get the file from S3
    file_obj = s3_client.get_object(Bucket=bucket_name, Key=object_key)
    file_content = file_obj['Body'].read().decode('utf-8')

    # Convert content to uppercase
    upper_content = file_content.upper()
```

```
    # Upload the modified content back to S3
    s3_client.put_object(Bucket=bucket_name, Key=destination_key, Body=upper_content)

    return {
        'statusCode': 200,
        'body': 'File processed and uploaded successfully'
    }
```

S3 upload code:

```
import boto3

# Initialize an S3 client
s3_client = boto3.client('s3')

# The string you want to upload
my_string = "This is a simple Tw sfso  string."

# The S3 bucket name
bucket_name = 'myfirstbucket-1313'

# The key (file name) to use for the uploaded string, including the folder path
object_key = 'upload_folder/myfile13.txt'

# Upload the string
s3_client.put_object(Bucket=bucket_name, Key=object_key, Body=my_string)
```

Input Test:

```
# The string you want to upload
my_string = "This is a simple Tw sfso  string."
```

Output text:

```
THIS IS A SIMPLE TW SFSO  STRING.
```

Monitoring the lambda log with cloud watch: