1. for upload data to s3:

import boto3

import requests

s3 = boto3.client('s3')

def lambda\_handler(event, context):

# Define the S3 bucket and key for the new object

bucket\_name = 'mis-612-project-1-1'

object\_key = 'zillow-data.csv'

# Download the CSV file from the URL

url = 'https://data.nasdaq.com/api/v3/datatables/ZILLOW/DATA.csv?api\_key=W1qhgC\_VySjCT5eFia1y'

response = requests.get(url)

file\_content = response.content

# Upload the file to S3

s3.put\_object(Bucket=bucket\_name, Key=object\_key, Body=file\_content)

# Return a response

return {

'statusCode': 200,

'body': 'File uploaded to S3'

}

2. Tigger glue crawler:

import json

import boto3

glue = boto3.client('glue');

def lambda\_handler(event, context):

#TODO implement

response = glue.start\_crawler(

Name = ''

)

return {

'statusCode': 200,

'body': json.dumps('Hello from Lambda!')

}

3. Tigger Glue Job

import json

import boto3

def lambda\_handler(event, context):

glue = boto3.client('glue');

response = glue.start\_job\_run(JobName = " ")

print("Lambda Invoke")

4. Glue Code: ---------------------

import sys

from awsglue.transforms import \*

from awsglue.utils import getResolvedOptions

from pyspark.context import SparkContext

from awsglue.context import GlueContext

from awsglue.job import Job

## @params: [JOB\_NAME]

args = getResolvedOptions(sys.argv, ['JOB\_NAME'])

sc = SparkContext()

glueContext = GlueContext(sc)

spark = glueContext.spark\_session

job = Job(glueContext) job.init(args['JOB\_NAME'], args)

datasource0 = glueContext.create\_dynamic\_frame.from\_catalog(database = "{}", table\_name = "{}", transformation\_ctx = "datasource0")

datasink4 = glueContext.write\_dynamic\_frame.from\_options(frame = datasource0, connection\_type = "s3",

connection\_options = {"path": "s3://{}/{}/"}, format = "parquet", transformation\_ctx = "datasink4")

job.commit()

5. For Uploading data from Website

import boto3

import requests

import csv

import io

def lambda\_handler(event, context):

# Define your API key and URL

api\_key = 'W1qhgC\_VySjCT5eFia1y'

url = 'https://data.nasdaq.com/api/v3/datatables/ZILLOW/DATA.csv'

# Make a request to the API

params = {'api\_key': api\_key}

response = requests.get(url, params=params)

# Parse the CSV data

reader = csv.reader(io.StringIO(response.content.decode('utf-8')))

data = [row for row in reader]

# Upload the data to S3

s3 = boto3.client('s3')

bucket\_name = 'mis-612-pro'

folder\_name = 'inputfolder/'

file\_name = 'data.csv'

key = folder\_name + file\_name

s3.put\_object(Bucket=bucket\_name, Key=key, Body=response.content)

# Log the success

print(f'Data uploaded to S3: s3://{bucket\_name}/{key}')