12/28/22

Next: replace print with logger.info

Next: implement sync

import json

import logging

import boto3

import os

import traceback

import urllib3

import requests

import pprint as pp

logger = logging.getLogger()

logger.setLevel(logging.INFO)

def lambda\_handler(event, context):

print(event)

logger.info(event)

http = urllib3.PoolManager()

s3\_client = boto3.client('s3')

BLS\_BASE\_URL = os.getenv("BLS\_BASE\_URL")

html\_elements = []

bls\_dirs = ['pr']

file\_paths = []

urls = []

try:

requests\_session = requests.Session()

# response = requests\_session.get(BLS\_BASE\_URL + bls\_dirs[0] +'/')

# response = requests\_session.get(BLS\_BASE\_URL + "bd" +'/')

response = requests\_session.get(BLS\_BASE\_URL + "pr" +'/')

if response.status\_code == 200:

print(response.text)

print("")

for html\_element in response.text.split(' <A'):

# print(html\_element)

html\_elements.append(html\_element)

if html\_element.startswith(" HREF"):

file\_path = html\_element[html\_element.find("time.series/") + len("time.series/") : html\_element.find('">')]

file\_paths.append(file\_path)

# print(file\_path)

urls.append(BLS\_BASE\_URL + file\_path)

# HREF="/pub/time.series/pr/pr.class">pr.class</A><br>

print("html\_elements")

pp.pprint(html\_elements)

print("file\_paths:")

pp.pprint(file\_paths)

print("urls:")

pp.pprint(urls)

print("len of file\_paths:", len(file\_paths))

else:

return "Error when requesting page source, " + response.status

except Exception as e:

traceback.print\_exc()

# exception\_message = "Error"

logger.error(e)

return e

"""

Iterate over each pr file:

if it doesn't exist in the s3 bucket, then download to memory, upload to s3, update dynamodb with metadata.

if it exists in s3 bucket, then check metadata in dynamodb table, if different, then replace s3 file and update metadata

Then iterate over each s3 file:

if it doesn't exist on server, then remove it from s3 and delete the metadata.

"""

return {

'statusCode': 200,

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

}