

Project 03 Read Me

Streaming Data with AWS Lambda and tools

Introduction:

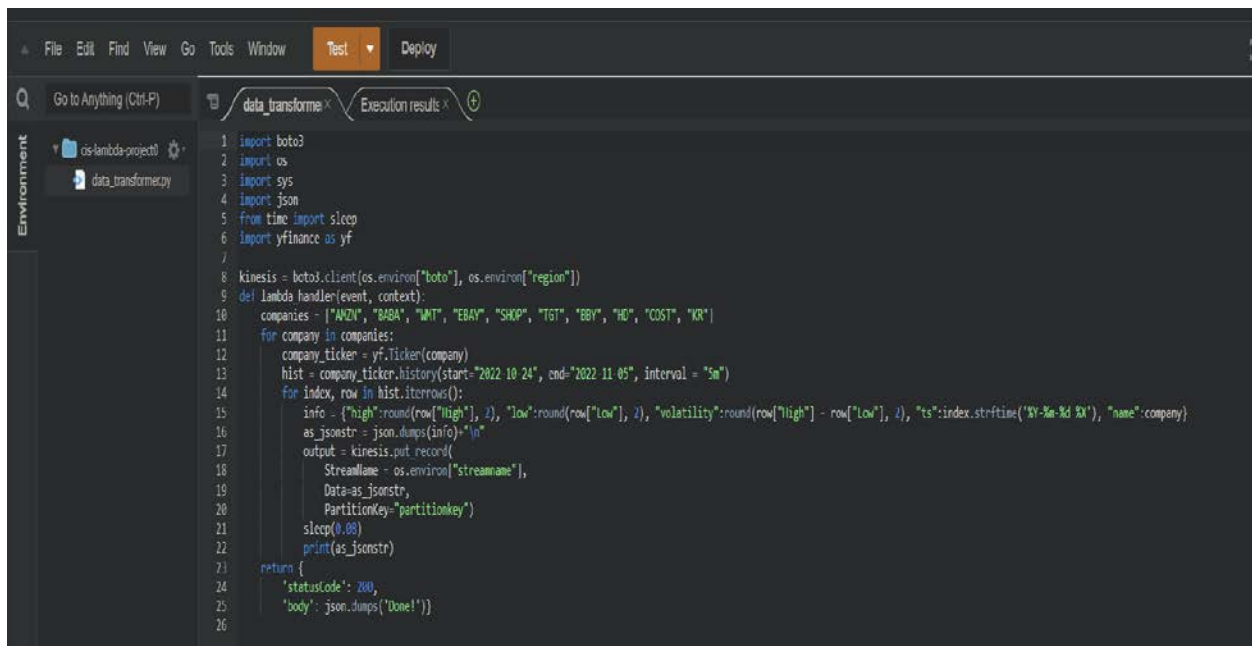
The goal of the project is to use various tools in AWS to perform analysis on stocks, by using the yfinance function from python as well as using the Lambda function in AWS; alongside with crawler, streaming tool such as Kinesis and finally query tool of Athena to get to result we want.

Data Collection:

We used yfinance to collect the data, for specific companies: AMZN, BABA, WMT, EBAY, SHOP, TGT, BBY, HD, COST, KR

The total data collected was throughout a 10-day duration, and we specifically had “high” and “low” of the day, we added another column of “volatility” to show how volatile the increase or decrease of the stock price.

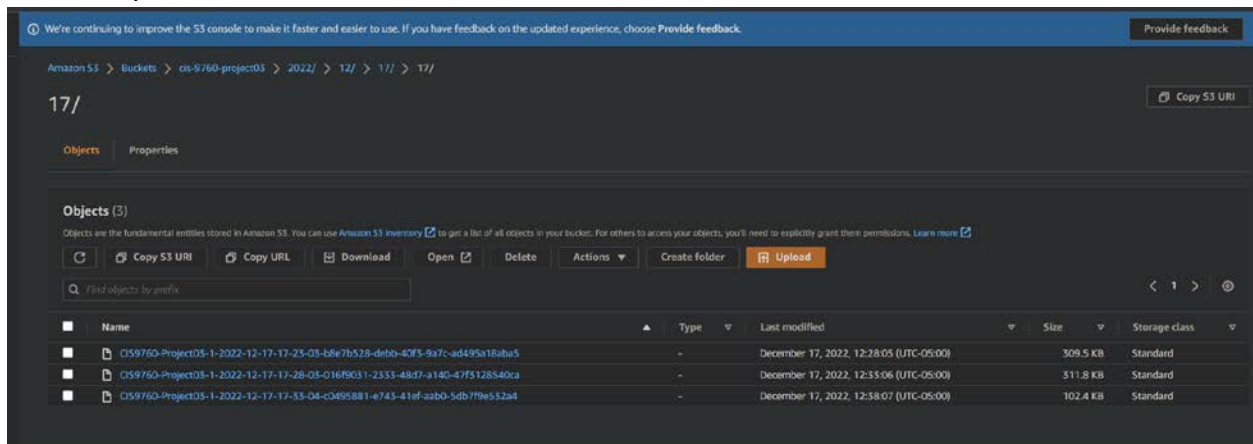
The Lambda function we used is as below:

The image is a screenshot of the AWS Lambda console's code editor. The interface shows a dark-themed editor with a file explorer on the left, a menu bar at the top, and tabs for 'data_transformer' and 'Execution results'. The code is written in Python and is designed to fetch stock data from Yahoo Finance (yfinance) for a list of companies (AMZN, BABA, WMT, EBAY, SHOP, TGT, BBY, HD, COST, KR) and stream it to an Amazon Kinesis stream. The code includes imports for boto3, os, sys, and json. It defines a lambda handler that iterates over the companies, fetches their stock history for a 10-day period, calculates the volatility (difference between high and low prices), and then puts the resulting JSON records into a Kinesis stream. The stream name and partition key are retrieved from environment variables. The code also includes a sleep function to control the rate of data streaming and a return statement with a status code and a 'Done!' message.

```
1 import boto3
2 import os
3 import sys
4 import json
5 from time import sleep
6 import yfinance as yf
7
8 kinesis = boto3.client(os.environ["boto"], os.environ["region"])
9 def lambda_handler(event, context):
10     companies = ["AMZN", "BABA", "WMT", "EBAY", "SHOP", "TGT", "BBY", "HD", "COST", "KR"]
11     for company in companies:
12         company_ticker = yf.Ticker(company)
13         hist = company_ticker.history(start="2022-10-24", end="2022-11-05", interval="5m")
14         for index, row in hist.iterrows():
15             info = {"high":round(row["High"], 2), "low":round(row["Low"], 2), "volatility":round(row["High"] - row["Low"], 2), "ts":index.strftime("%Y-%m-%d %X"), "name":company}
16             as_jsonstr = json.dumps(info)+"\n"
17             output = kinesis.put_record(
18                 StreamName = os.environ["streamname"],
19                 Data=as_jsonstr,
20                 PartitionKey="partitionkey")
21             sleep(0.08)
22             print(as_jsonstr)
23     return {
24         'statusCode': 200,
25         'body': json.dumps('Done!')}
26
```

It resulted in the following data that were streamed using AWS Kinesis:

And finally, the information that was streamed from Kinesis into S3 bucket:



We then use this information and query it using the AWS Athena tool to get the final result we want, that we export it as csv and at the end, use Jupyter notebook to perform graphical analysis.

