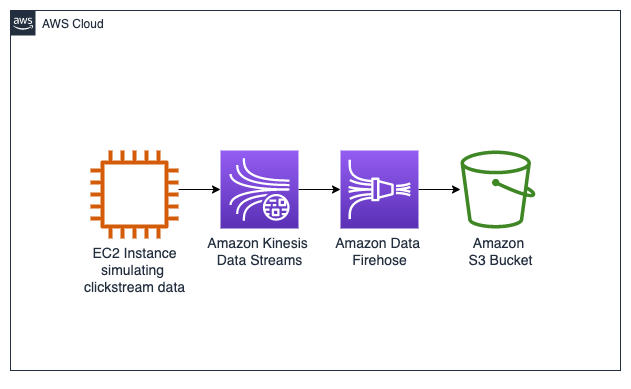
# Lab 1 - Setting up a Streaming Delivery Pipeline with Amazon Kinesis

Objectives

1. Create a Kinesis Data Firehose stream and connect the Kinesis data stream to Kinesis Data Firehose.
2. Configure dynamic partitioning on the Kinesis Data Firehose delivery stream.
3. Deliver data to Amazon S3.



1. EC2 Instance simulating clickstream data: A virtual server in AWS that generates mock click data to mimic user activity on an e-commerce website.
2. Amazon Kinesis Data Streams: A service that collects and processes real-time data streams, capturing the click data generated by the EC2 instance.
3. Amazon Data Firehose: A service that takes data from Kinesis Data Streams and delivers it to destinations such as Amazon S3, applying any necessary transformations.
4. Amazon S3 Bucket: A storage service where the processed clickstream data is stored, organized, and made available for further analysis.

**Task 1: Simulate clickstream data generation**

1.1 open the **EC2 Producer terminal** using URL in lab, a terminal opens.

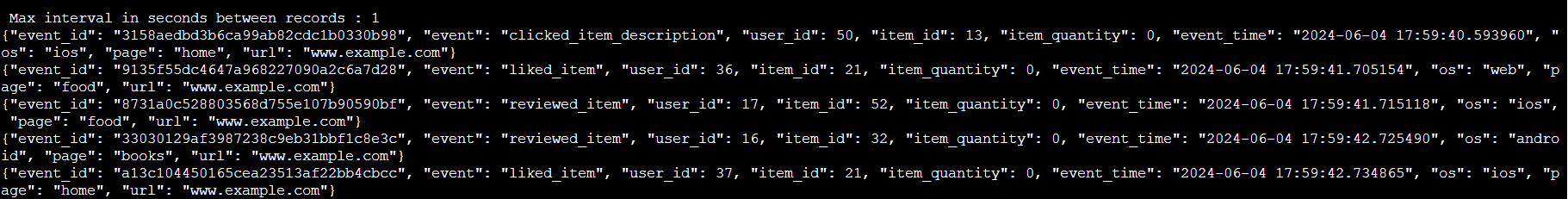
**EC2 PT is like a virtual constumer in this case that will imitate clicks.**

1.2 Use the following code to start the clickstream\_generator\_items.py script

**STREAM\_NAME=$(aws kinesis list-streams --query "StreamNames[?contains(@, 'KdsClickstreamData')]" --output text)**

**echo -e "\n\nThe stream name is : $STREAM\_NAME\n\n"**

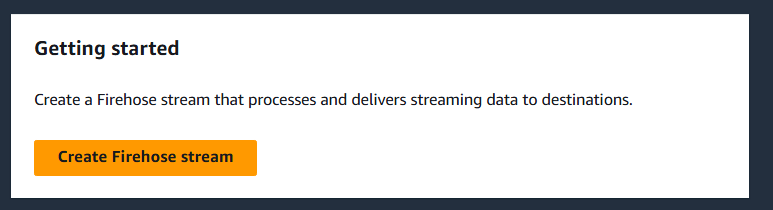
**python3 clickstream\_generator\_items.py $STREAM\_NAME 1 1**



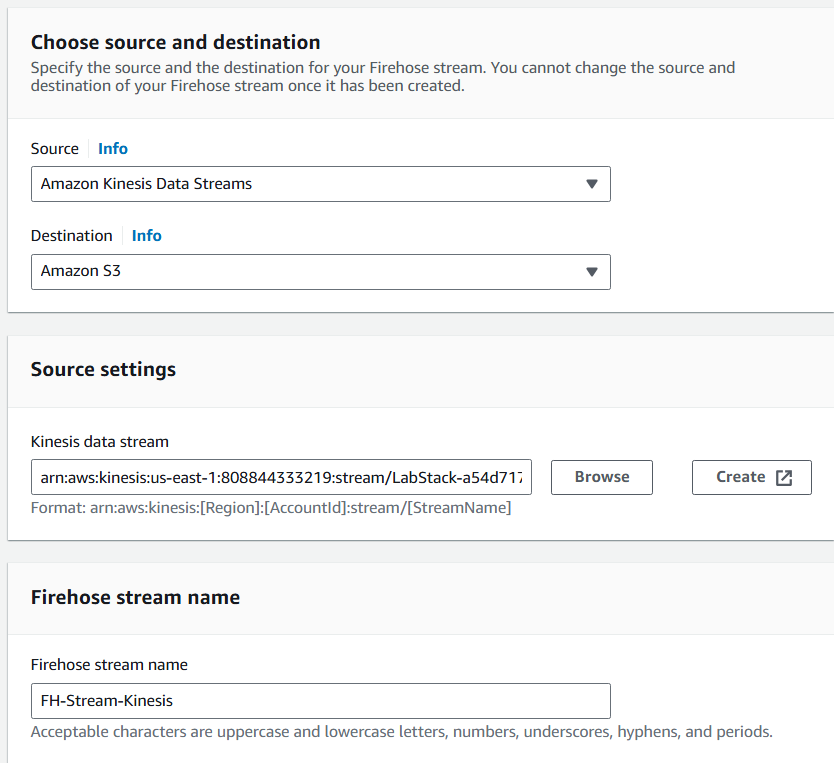
These steps will simulate user activities on a website and send the data to a Kinesis data stream for further analysis.

**Task 2: Create and configure a Kinesis Data Firehose delivery stream**

2.1 Open FIreHose in AWS console, and click on

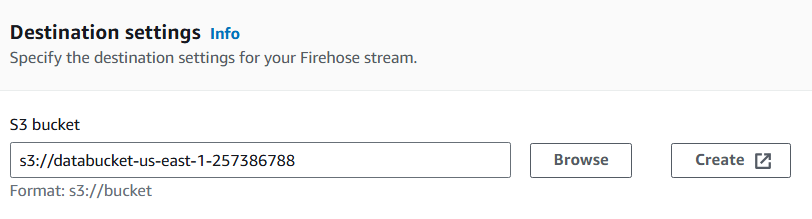


2.2 Setting up the stream



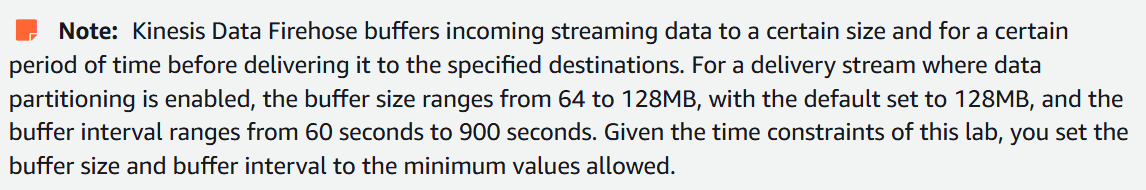
Source - Kinesis

Destination - S3



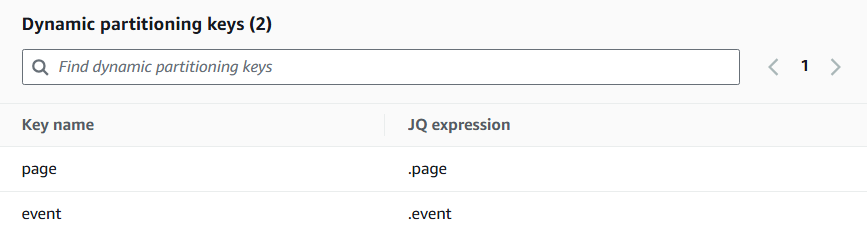
**NOTE - There were bunch of other technical specifications in building the data stream, that I did not note here.**

**Like, we gave values for buffer size and buffer time**

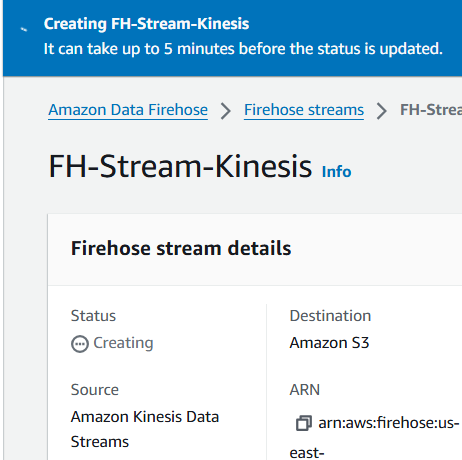
****

**Also we setup IAM role that Kinesis Data Firehose uses to access your S3 bucket.**

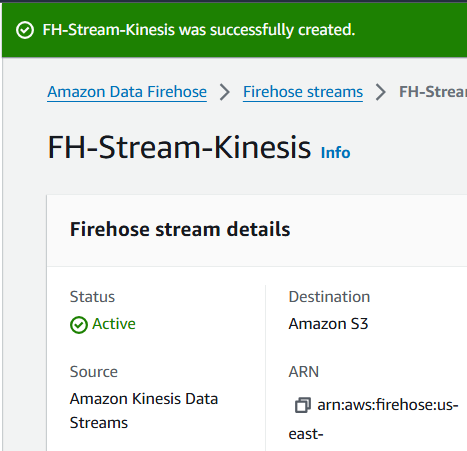
**We also setup dynamic partitioning, which means that data stored in S3 will be categorized based on events (type of product in this example).**

****

2.3 FH stream creating



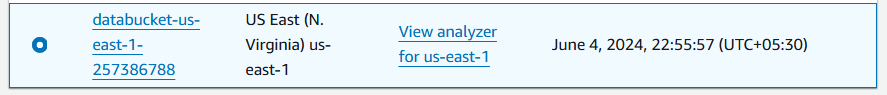
2.4 Created



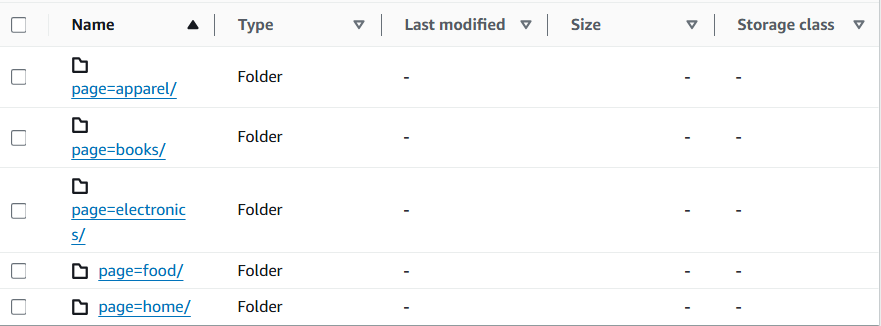
**The output will be stored in Amazon S3 and will be partitioned into page and further into events.**

**Task 3: Verify your output in Amazon S3**

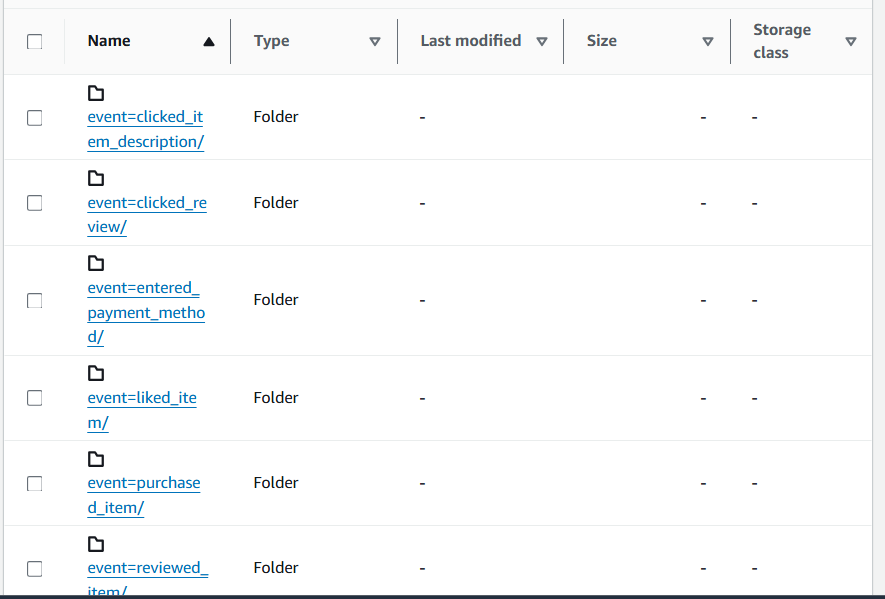
3.1 Open S3 and open this bucket

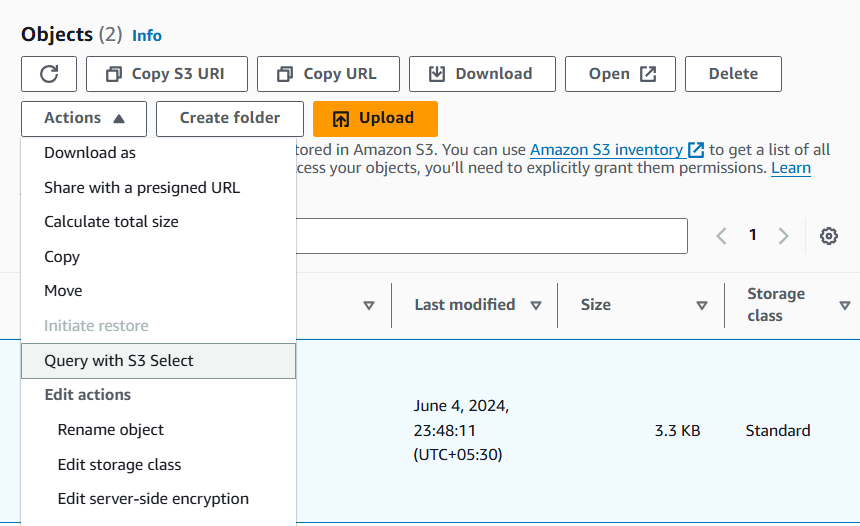


3.2 Folders based on dynamic partitioning appear in the bucket

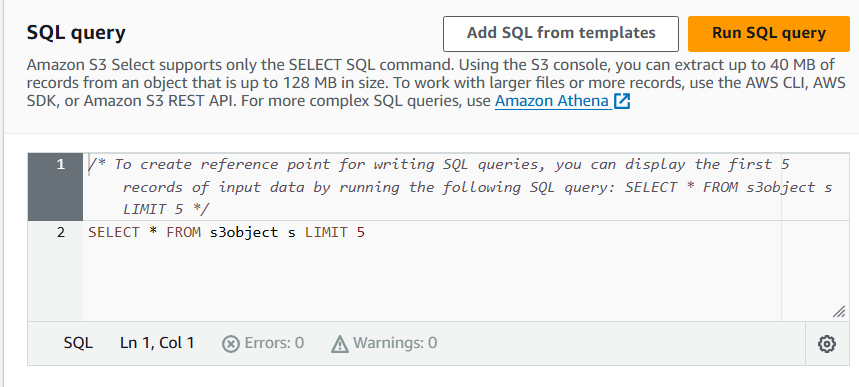


3.3 Inside every page (catoegory in this example), there are following events



3.4 Select any item inside an event, and choose:-

3.5 Run the following SQL query



3.6 Example output

