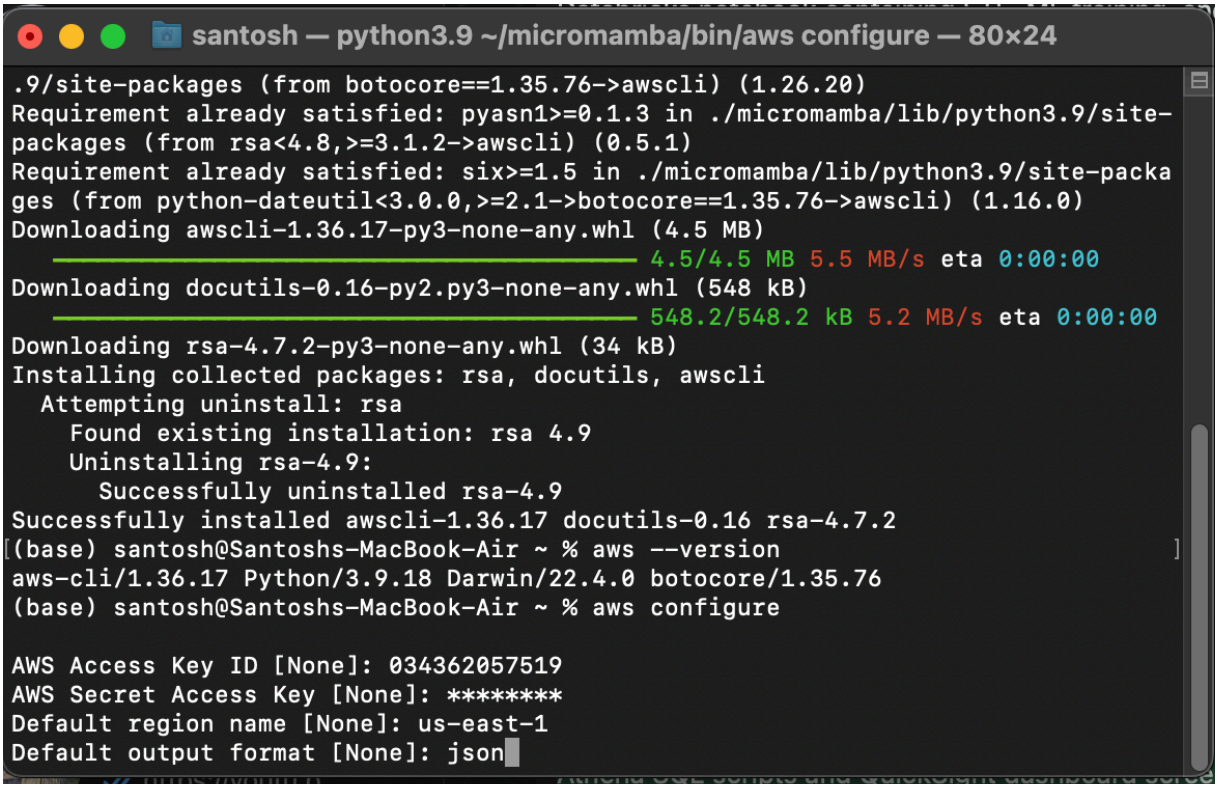
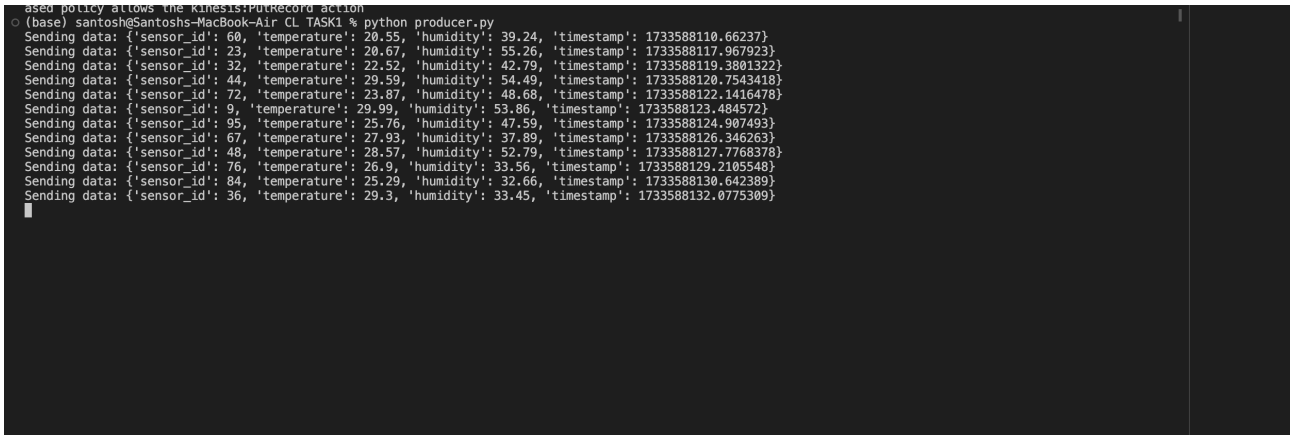
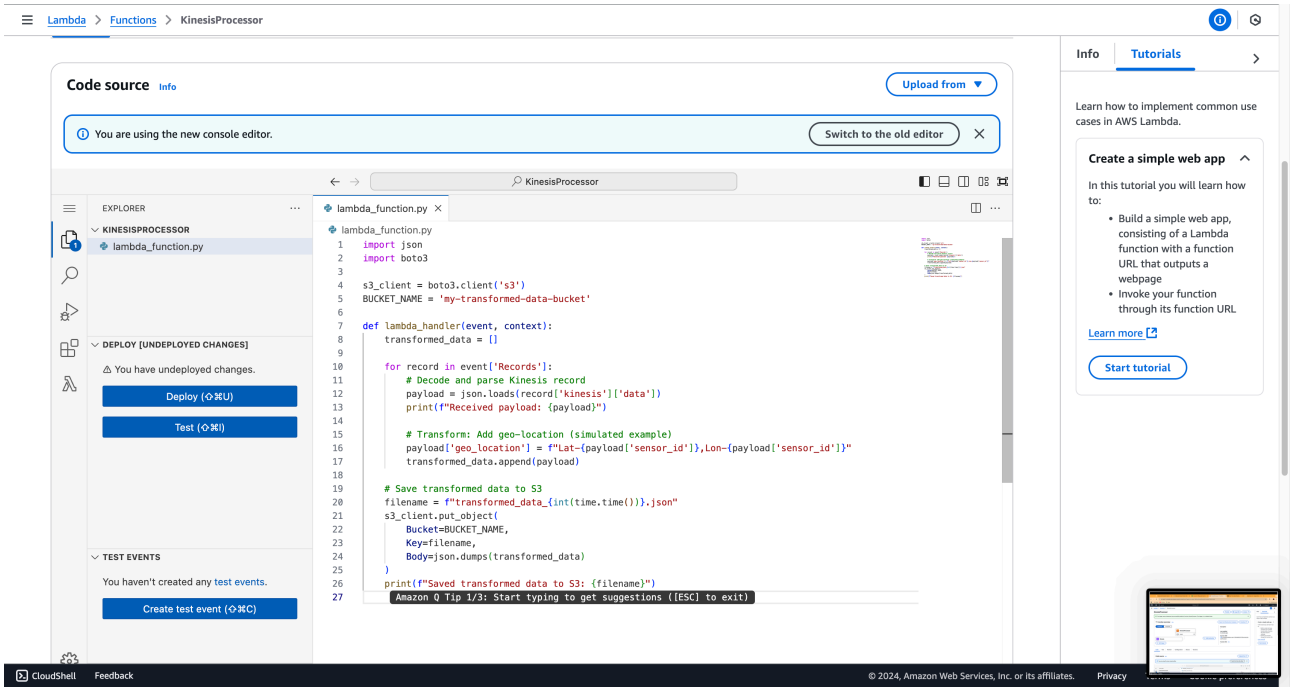
Step 1: Connect to AWS Account with AccountID and Secret Pass Key:

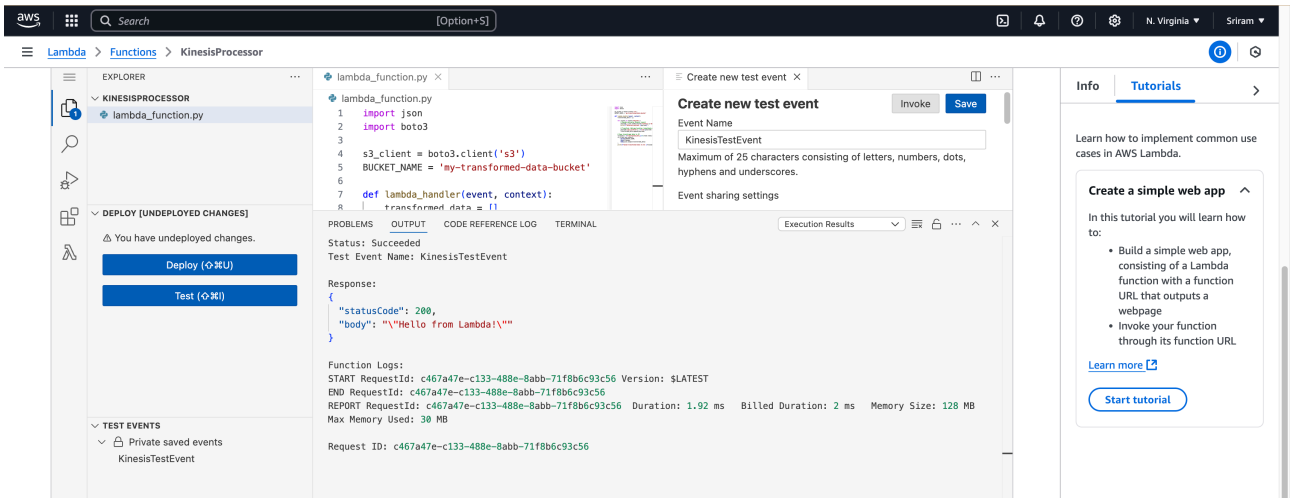


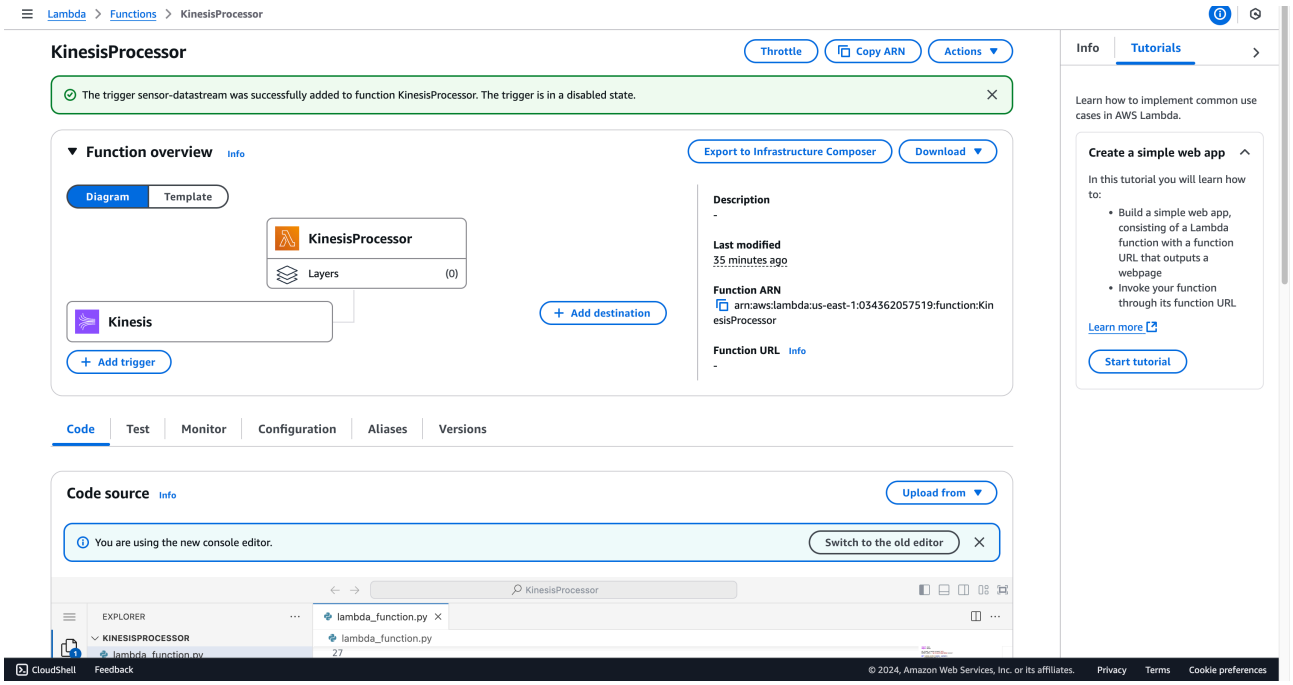
Step 2: run the code “producer.py” and sent the real time data to AWS:



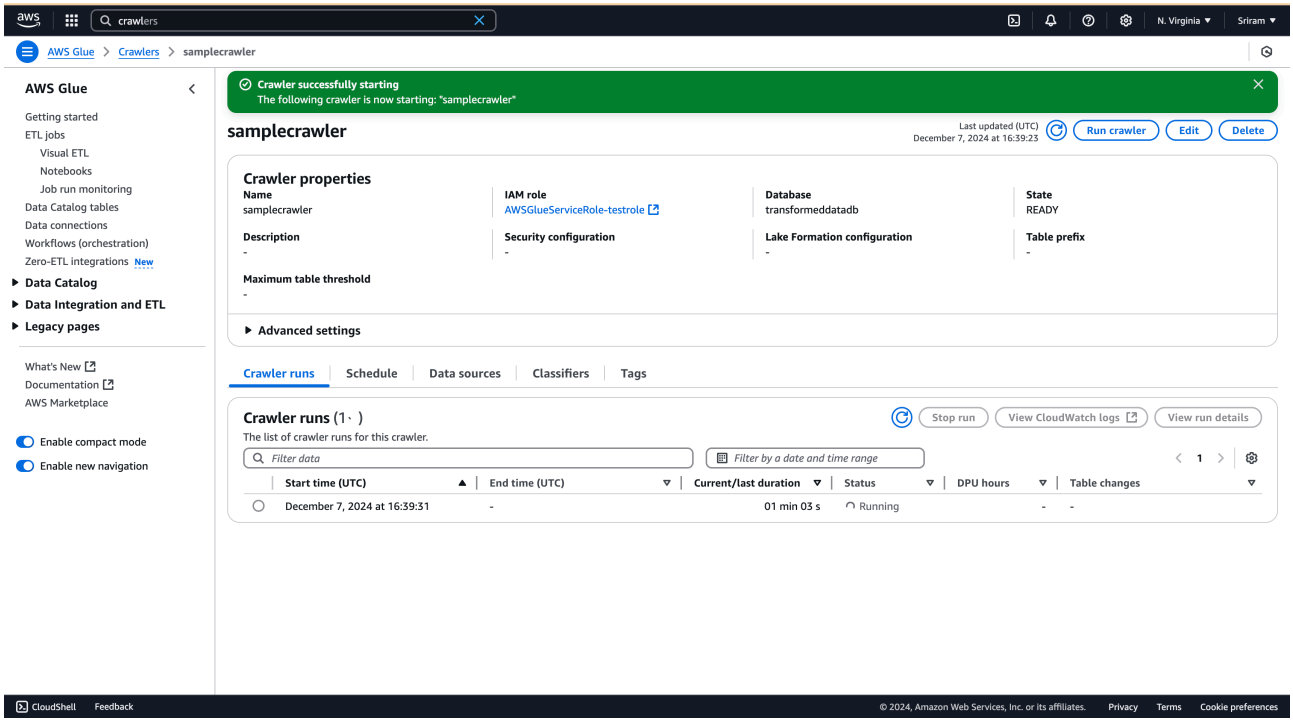
Step 3:Login to AWS and step into kenosis and create a lamb function to get the input data stream and store it in a S3 bucket


Lambda > Functions > KinesisProcessor
Code source Info
Upload from
You are using the new console editor.
Switch to the old editor
KinesisProcessor
EXPLORER
KINESISPROCESSOR
lambda_function.py
DEPLOY [UNDEPLOYED CHANGES]
You have undeployed changes.
Deploy (⇧⌘U)
Test (⇧⌘I)
TEST EVENTS
You haven't created any test events.
Create test event (⇧⌘C)
lambda_function.py
import json
import boto3
s3_client = boto3.client('s3')
BUCKET_NAME = 'my-transformed-data-bucket'
def lambda_handler(event, context):
transformed_data = []
for record in event['Records']:
# Decode and parse Kinesis record
payload = json.loads(record['kinesis']['data'])
print(f"Received payload: {payload}")
# Transform: Add geo-location (simulated example)
payload['geo_location'] = f"Lat-{payload['sensor_id']},Lon-{payload['sensor_id']}"
transformed_data.append(payload)
# Save transformed data to S3
filename = f"transformed_data_{int(time.time())}.json"
s3_client.put_object(
Bucket=BUCKET_NAME,
Key=filename,
Body=json.dumps(transformed_data)
)
print(f"Saved transformed data to S3: {filename}")
Amazon Q Tip 1/3: Start typing to get suggestions ([ESC] to exit)
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy

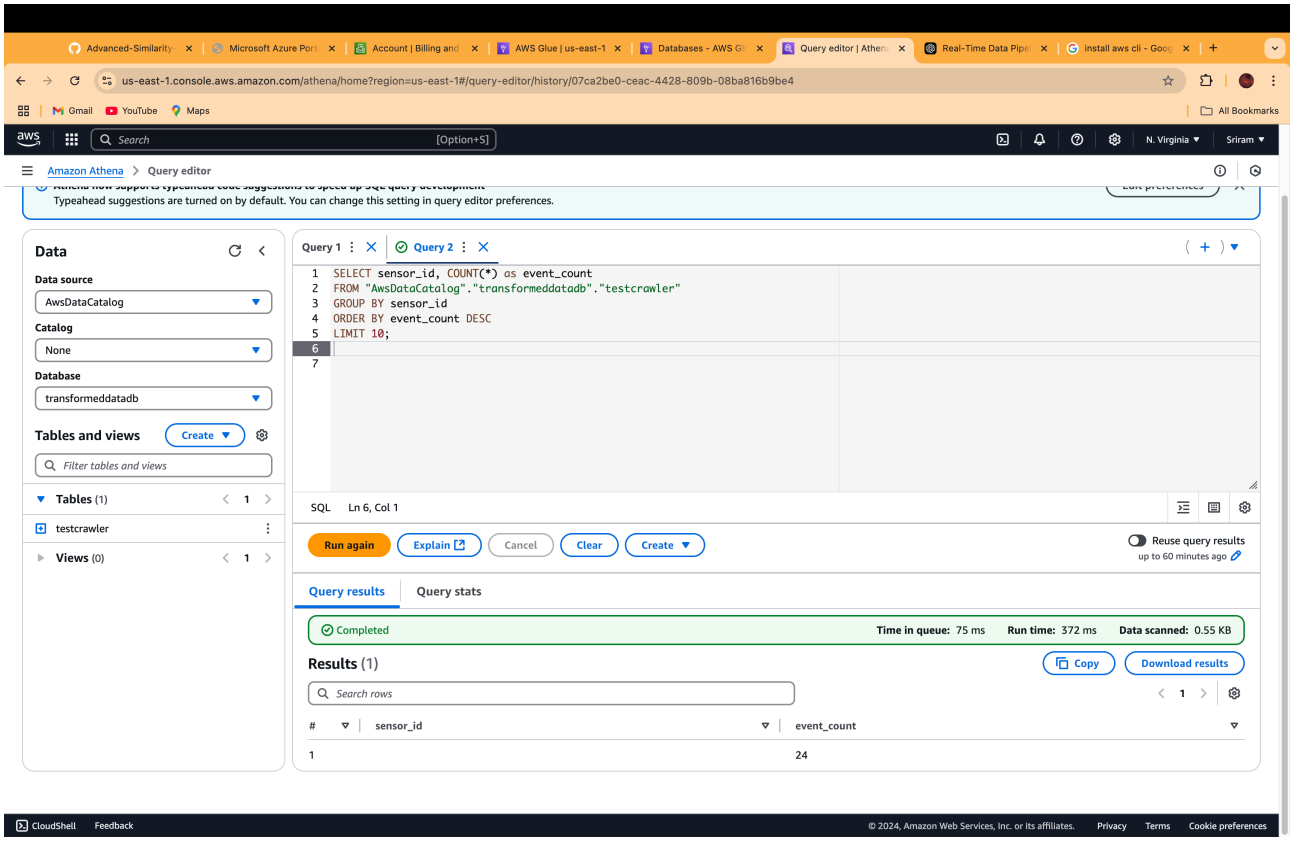

Search
[Option+S]
N. Virginia
Sriram
Lambda > Functions > KinesisProcessor
EXPLORER
KINESISPROCESSOR
lambda_function.py
DEPLOY [UNDEPLOYED CHANGES]
You have undeployed changes.
Deploy (⇧⌘U)
Test (⇧⌘I)
TEST EVENTS
Private saved events
KinesisTestEvent
lambda_function.py
Create new test event
Create new test event
Invoke
Save
Event Name
KinesisTestEvent
Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.
Event sharing settings
PROBLEMS
OUTPUT
CODE REFERENCE LOG
TERMINAL
Execution Results
Status: Succeeded
Test Event Name: KinesisTestEvent
Response:
{
"statusCode": 200,
"body": "\"Hello from Lambda!\""
}
Function Logs:
START RequestId: c467a47e-c133-488e-8abb-71f8b6c93c56 Version: $LATEST
END RequestId: c467a47e-c133-488e-8abb-71f8b6c93c56
REPORT RequestId: c467a47e-c133-488e-8abb-71f8b6c93c56 Duration: 1.92 ms Billed Duration: 2 ms Memory Size: 128 MB
Max Memory Used: 30 MB
Request ID: c467a47e-c133-488e-8abb-71f8b6c93c56
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial


Lambda > Functions > KinesisProcessor
KinesisProcessor
Throttle
Copy ARN
Actions
The trigger sensor-datastream was successfully added to function KinesisProcessor. The trigger is in a disabled state.
Function overview Info
Export to Infrastructure Composer
Download
Diagram
Template
KinesisProcessor
Layers (0)
Kinesis
Add destination
Add trigger
Description
-
Last modified
35 minutes ago
Function ARN
arn:aws:lambda:us-east-1:034362057519:function:KinesisProcessor
Function URL Info
-
Code
Test
Monitor
Configuration
Aliases
Versions
Code source Info
Upload from
You are using the new console editor.
Switch to the old editor
KinesisProcessor
EXPLORER
KINESISPROCESSOR
lambda_function.py
lambda_function.py
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 4:Create a Database and table inside it to store the transformed data in S3 buckets, set an IAM and edit the manage preferences



Step 5:Now Step into to Amazon Athena and connect the Database and start Querrying to retrieve data i.e(Data Analysis)



Step 6: use QuickSight for analysis the datasets from Athena and connect to the database