AWS SOLUTION ARCHITECT COURSE-END PROJECT

ASHUTOSH SRIVASTAVA

12ASHU12@GMAIL.COM

WWW.LINKEDIN.COM/IN/ASHUTOSH-SRIVASTAVA-12ASHU12

PROBLEM STATEMENT AND MOTIVATION CONTENT

- Real-World Scenario:
  TELEMAX is a company expanding its network into underserved markets globally.
  They provide innovative communications hardware for efficient networking links.

- Need:
  Deploy an effective NoSQL-based data warehousing architecture.
  Handle real-time data for future analysis to optimize network topologies.

- Solution Approach:
  Utilize AWS cloud services to build a real-time data management system.

SUMMARY OF STEPS

Step 1: Create AWS Kinesis Data Stream

Setup Kinesis stream to handle incoming data.

Step 2: Create AWS Lambda Function for Data Ingestion

Develop Lambda function to pull data from a URL and stream it to Kinesis.

Step 3: Add Python Layer to Lambda Function

Include necessary libraries like requests in a Lambda layer.

Step 4: Deploy and Verify Kinesis Data Stream

Check if data is streaming correctly using Kinesis Data Viewer.

Step 5: Create Lambda Function for Data Loading

Develop Lambda function to read from Kinesis and write to DynamoDB.

Step 6: Configure Permissions for Lambda Functions

Assign appropriate roles and permissions to Lambda functions.

Step 7: Monitor CloudWatch Logs

Review logs to ensure Lambda functions are running correctly and debug issues.

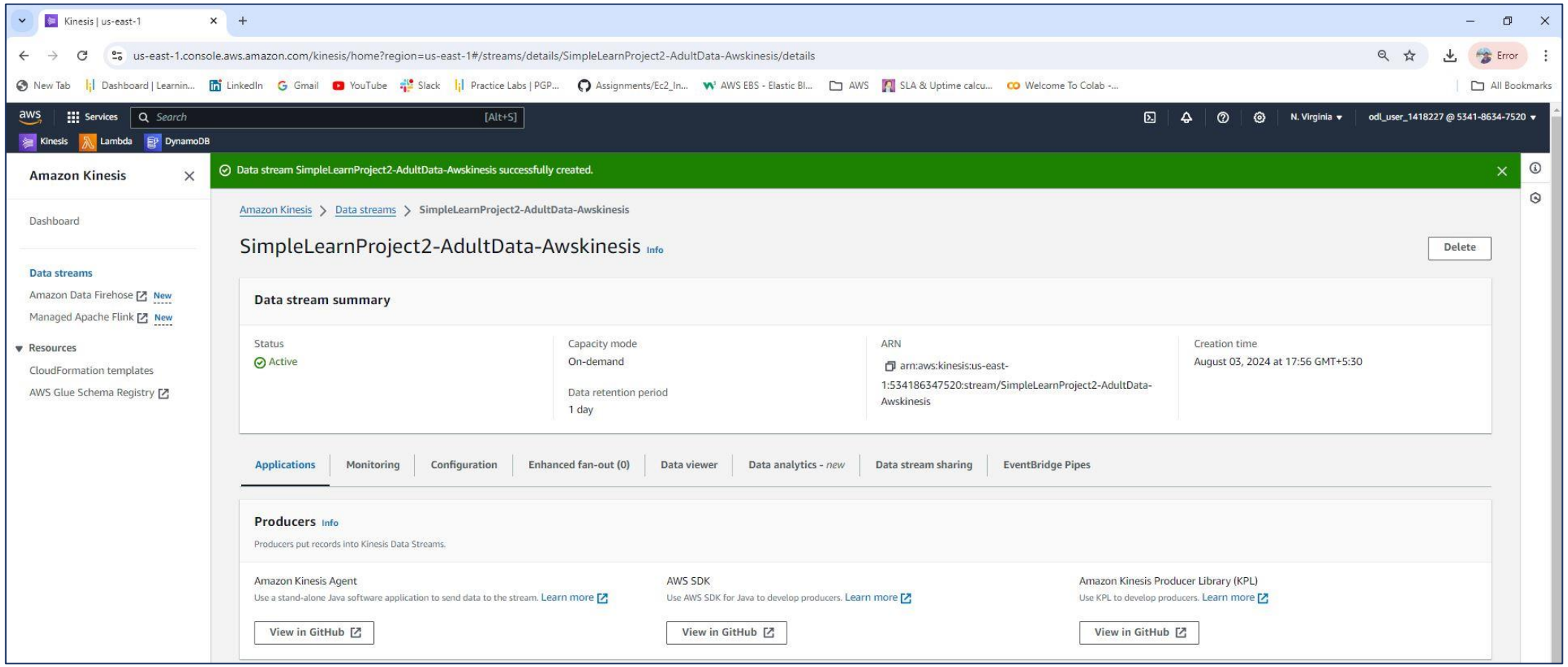
Step 8: Verify Data in DynamoDB

Check DynamoDB to confirm data is loaded and accessible.

STEP 1 : AWS KINESIS DATA STREAM CREATION

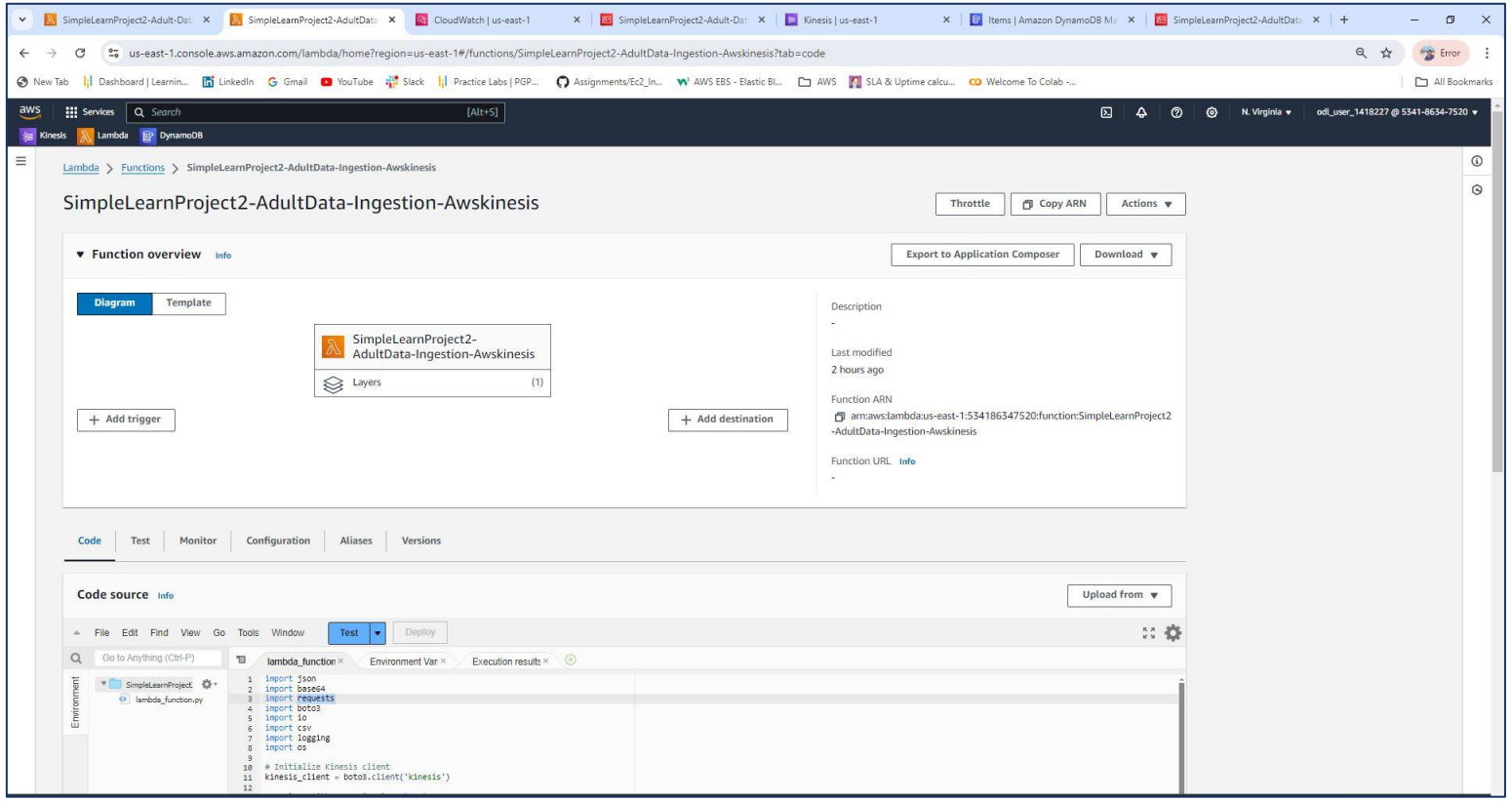
Created AWS Kinesis Data Stream named: SimpleLearnProject2-AdultData-Awskinesis

- Screenshot: [Include screenshot of Kinesis Data Stream creation]



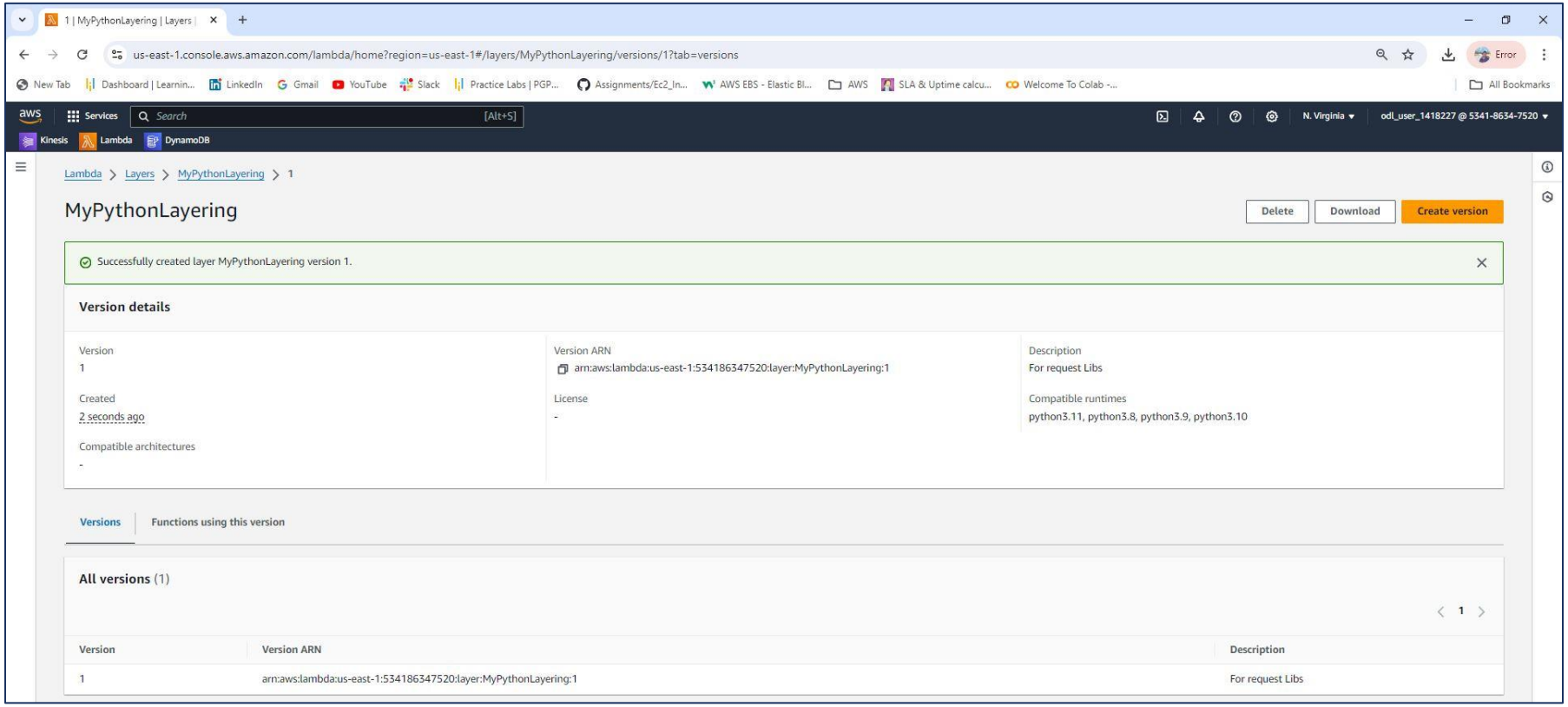
STEP 2: CREATE PYTHON CODE ON AWS LAMBDA FUNCTION

Lambda function to ingest data from URL:
https://archive.ics.uci.edu/ml/machine-learningdatabases/adult/adult.data
Details:
Function processes data and streams it to Kinesis
Screenshot: [Include screenshot of Lambda function code]



STEP 3: AWS LAMBDA FUNCTION WITH PYTHON LAYER

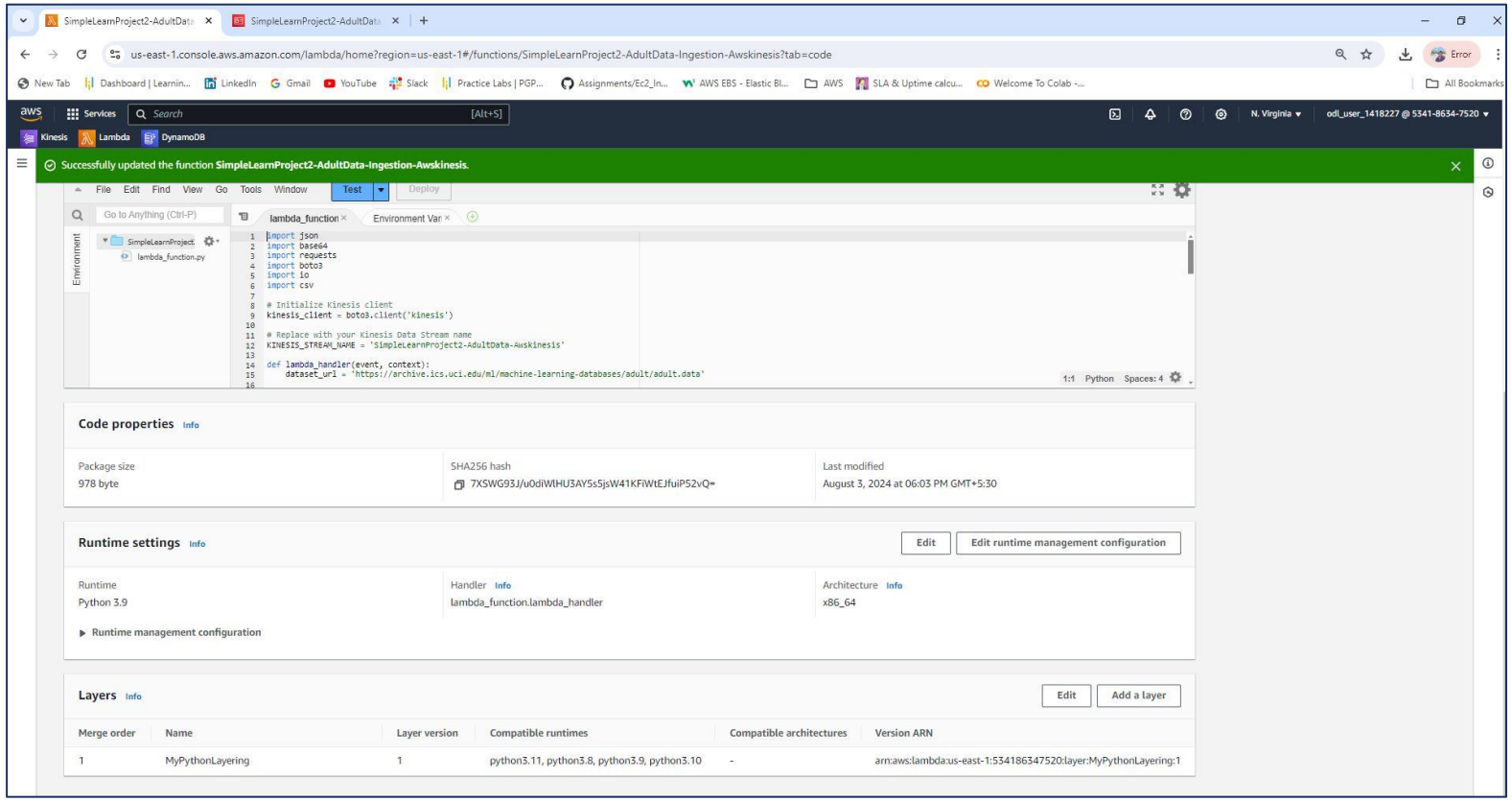
- Lambda function name: SimpleLearnProject2-AdultData-Ingestion-Awskinesis
- Added Python layer for requests library (not available by default)
- Screenshots:



STEP 3: AWS LAMBDA FUNCTION WITH PYTHON LAYER

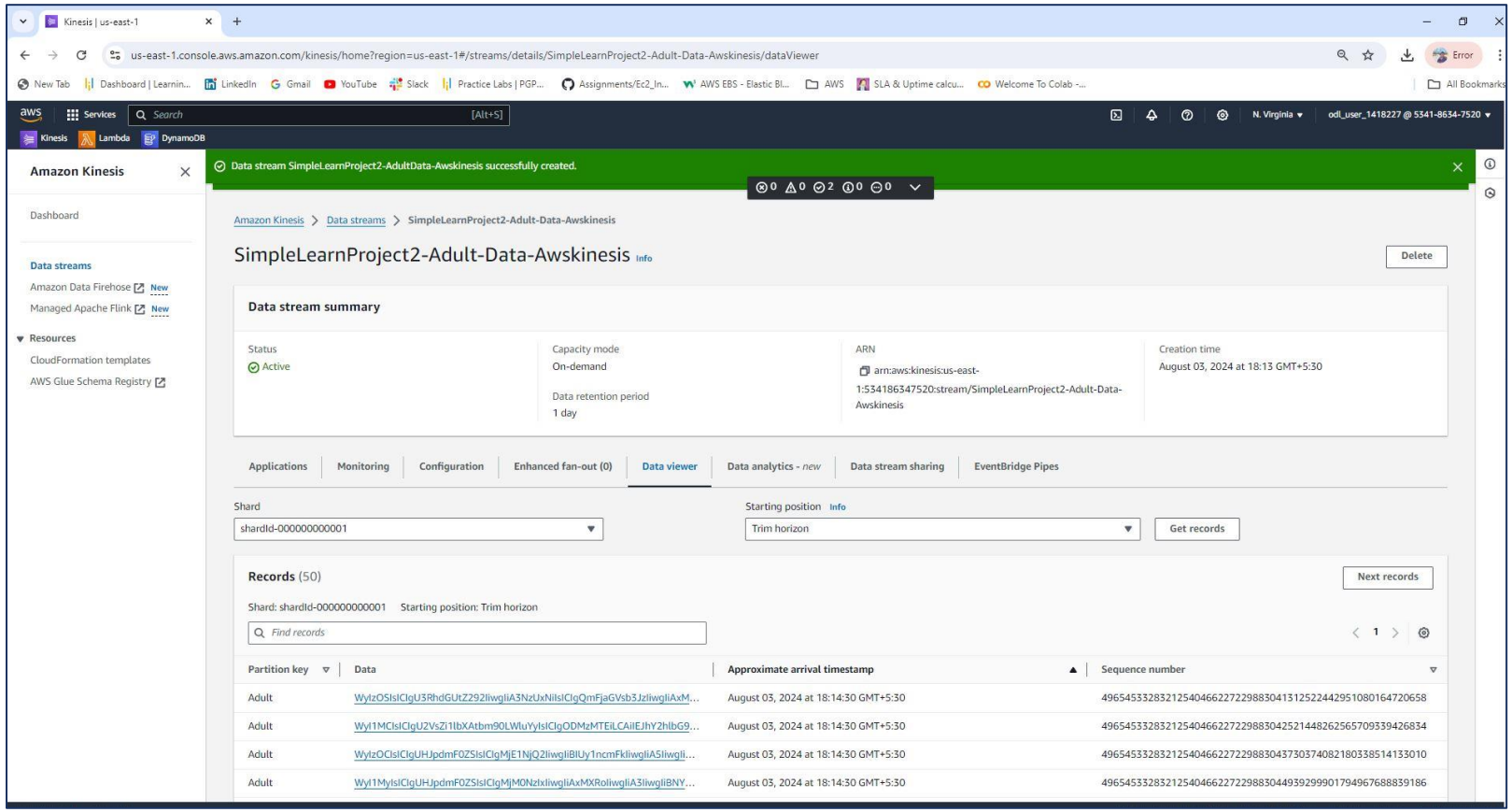
CONTINUE

- Layer added to Lambda Function
- Screenshots:



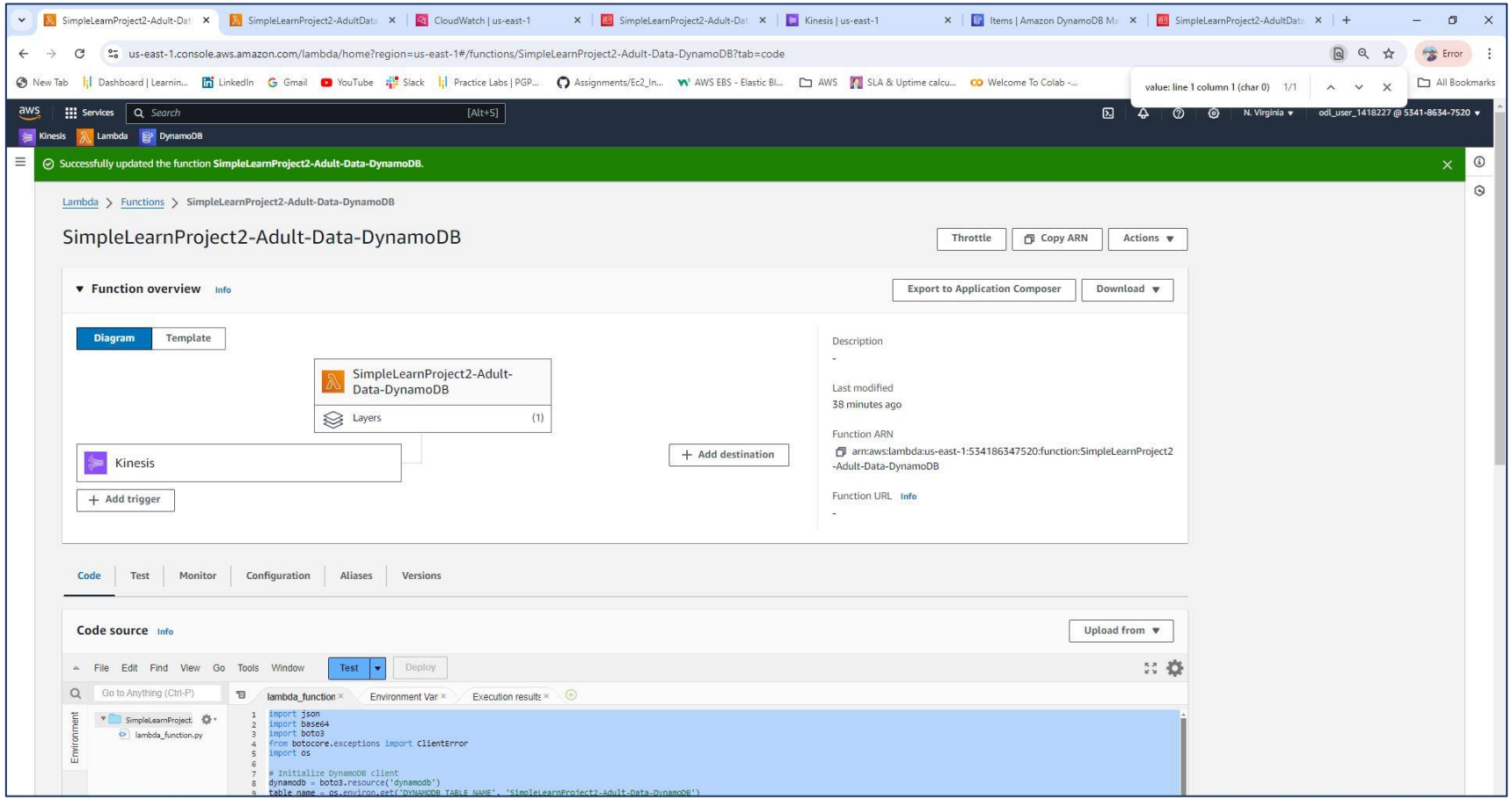
STEP 4: DEPLOY & VERIFY KINESIS DATA STREAM

- Deployed Lambda function and verified data streaming using Kinesis Data Viewer



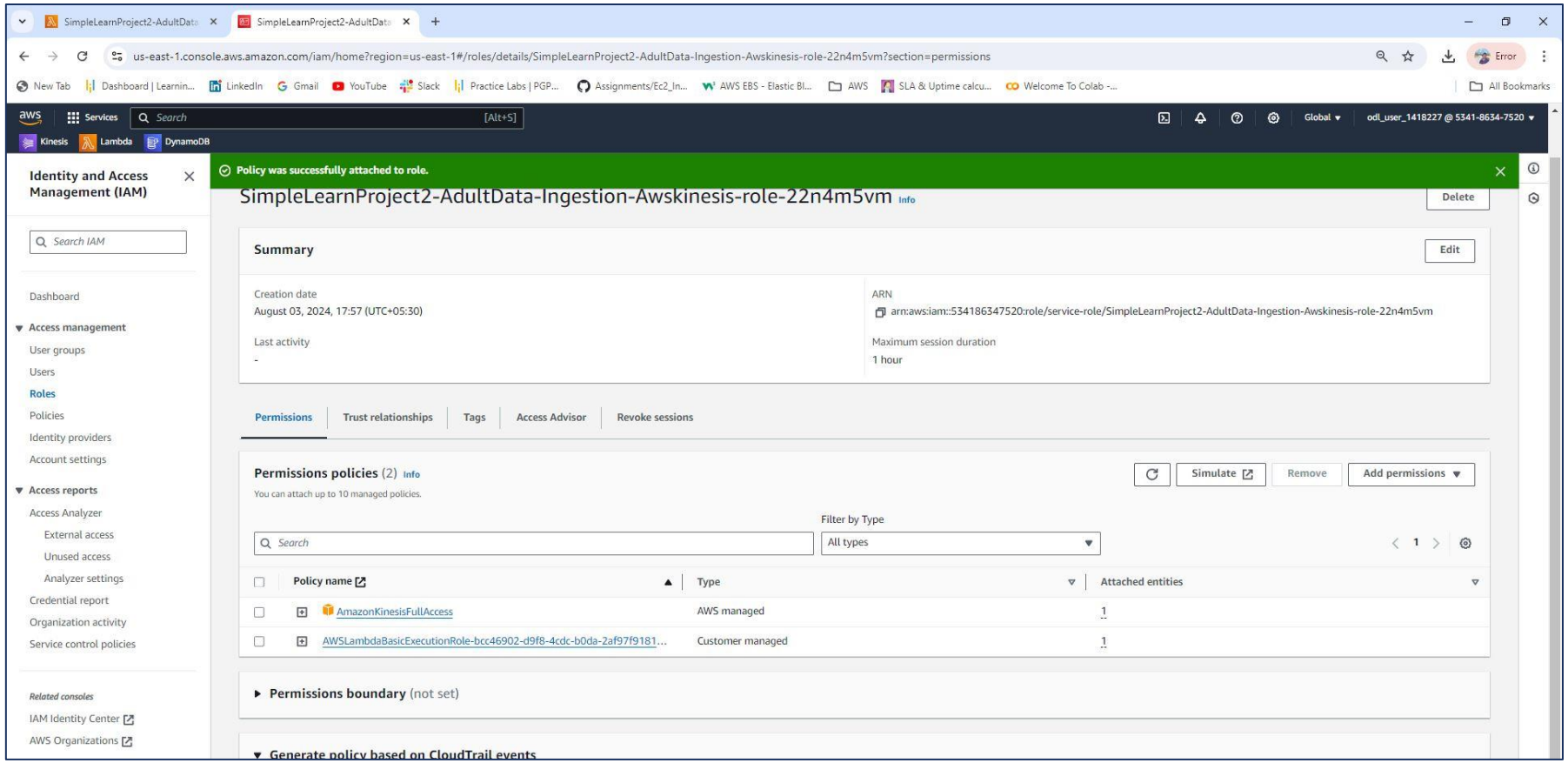
STEP 5: AWS LAMBDA FUNCTION FOR DATA LOADING

- Created Lambda function to read from Kinesis and load data into DynamoDB Lambda function name: SimpleLearnProject2-Adult-Data-DynamoDB-LambdaFunction



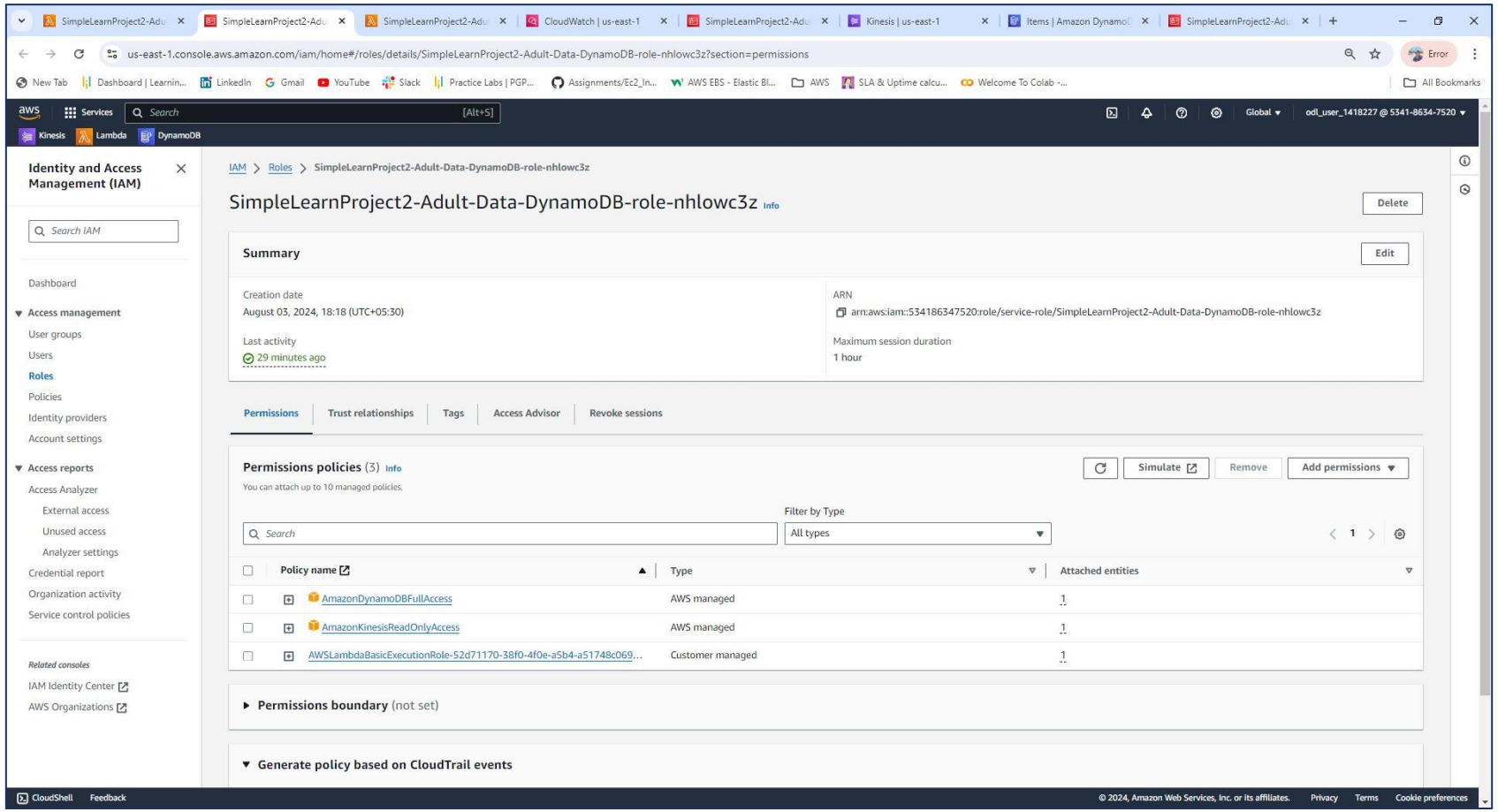
STEP 6: ADD PERMISSIONS TO LAMBDA FUNCTIONS

- Assigned necessary permissions for both Lambda functions, Below for Ingestion AWS Kinesis



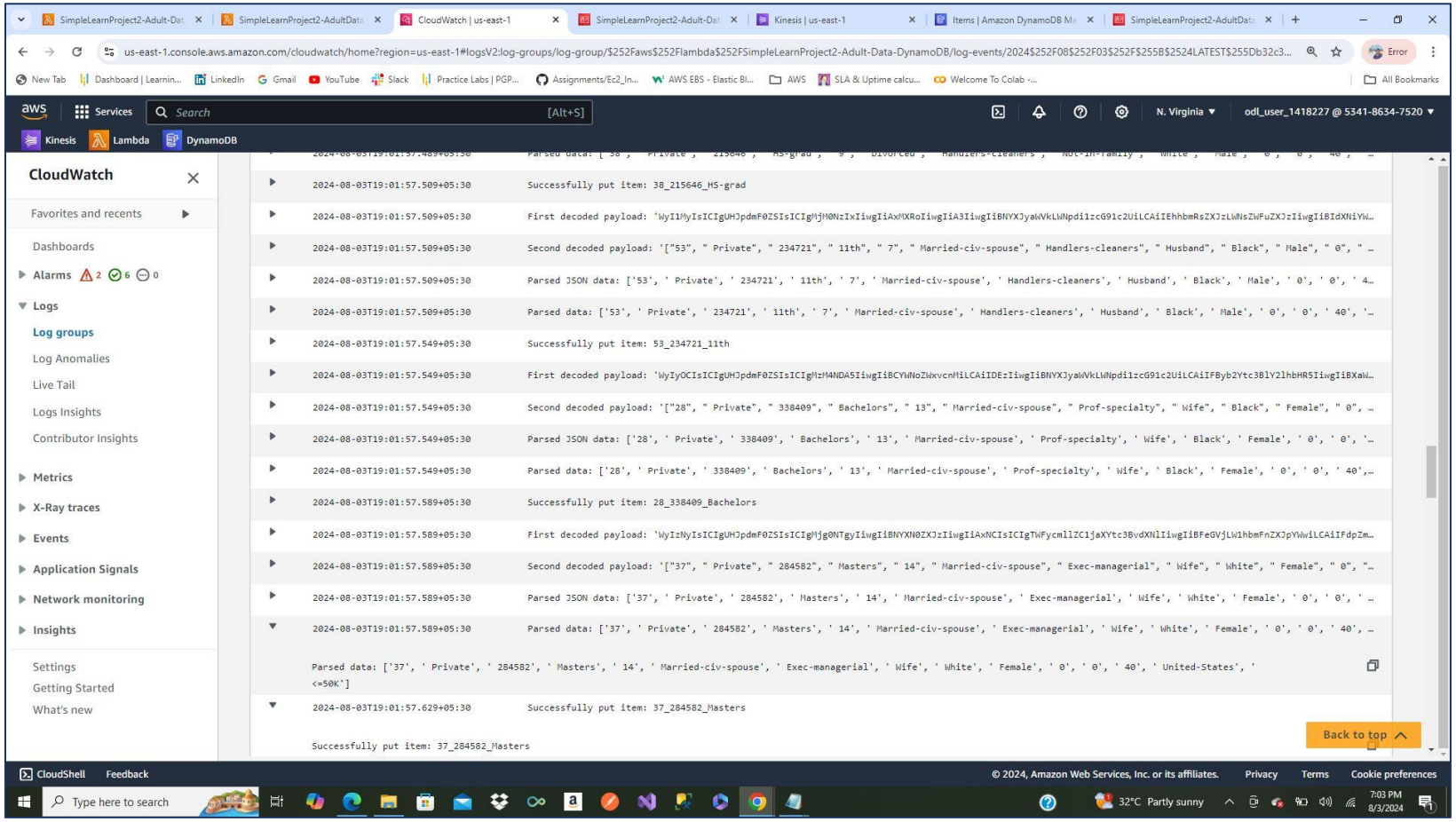
STEP 6: ADD PERMISSIONS TO LAMBDA FUNCTIONS

- Assigned necessary permissions for both Lambda functions, Below for Ingestion AWS DynamoDB



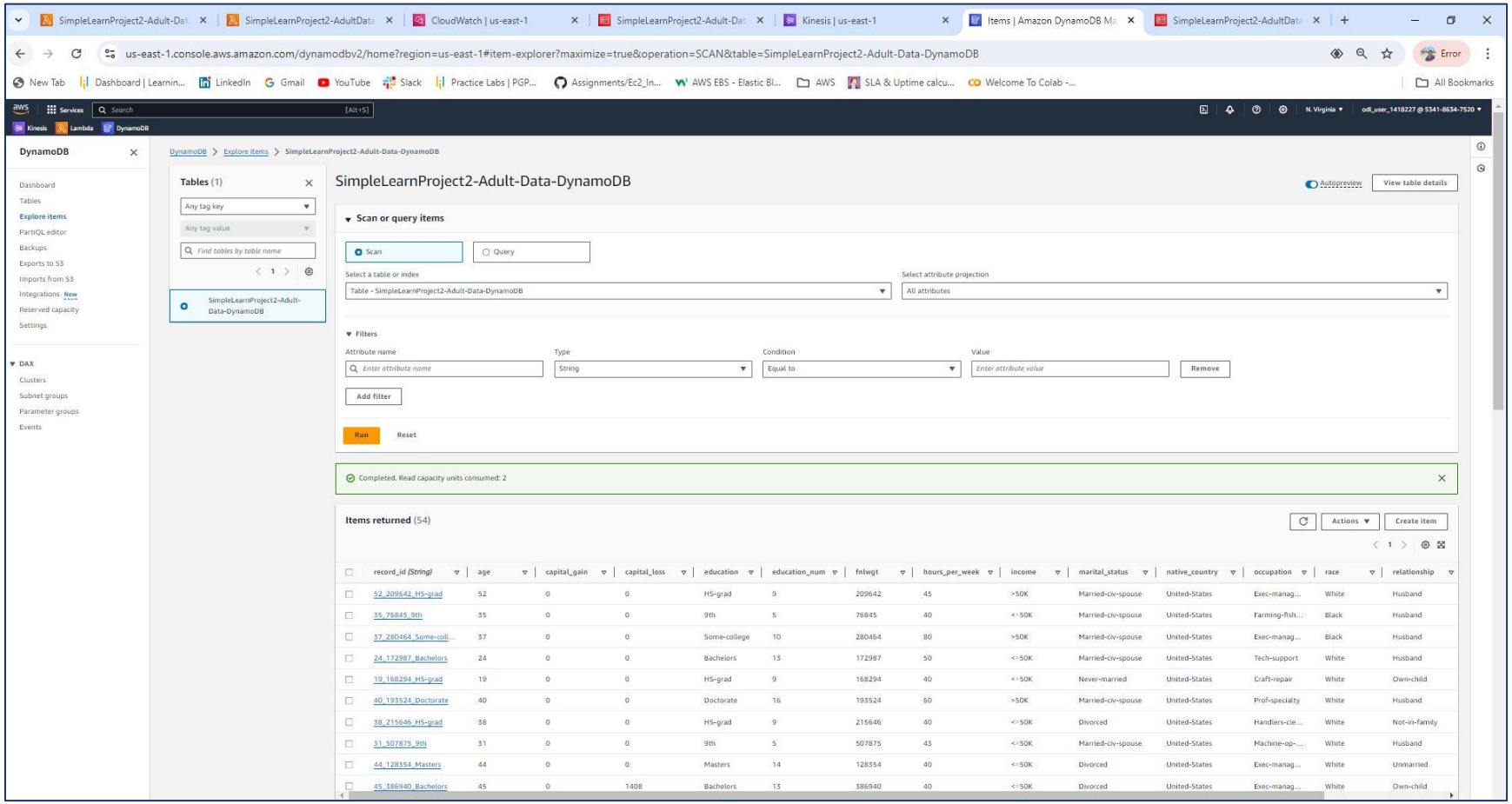
STEP 7: CHECK AWS CLOUDWATCH LOGS

- Monitored CloudWatch logs for SimpleLearnProject2-Adult-Data-DynamoDB-LambdaFunction to identify and debug errors



STEP 8: VERIFY DATA IN DYNAMODB TABLE

- Verified data was successfully loaded into DynamoDB table SimpleLearnProject2-Adult-Data-DynamoDB



CONCLUSION

- Successfully deployed a real-time data management solution on AWS.

Achievements:

- Streamed data using AWS Kinesis
- Processed and loaded data into DynamoDB using Lambda functions Verified data flow and integrity

THANK YOU

12ASHU12@GMAIL.COM

WWW.LINKEDIN.COM/IN/ASHUTOSH-SRIVASTAVA-12ASHU12