A close up of a logo

Description automatically generated

**Course-End Project**

**Infrastructure Deployment for Real-time Data Management Requirements on the AWS Cloud**



Note:

The availability zone is constant throughout all instances and volumes.

**Steps to be followed:**

1. Create an AWS Kinesis Data Streams
2. Create an AWS Lambda Function
3. Create an AWS DynamoDB Database
4. Perform Scan Operation

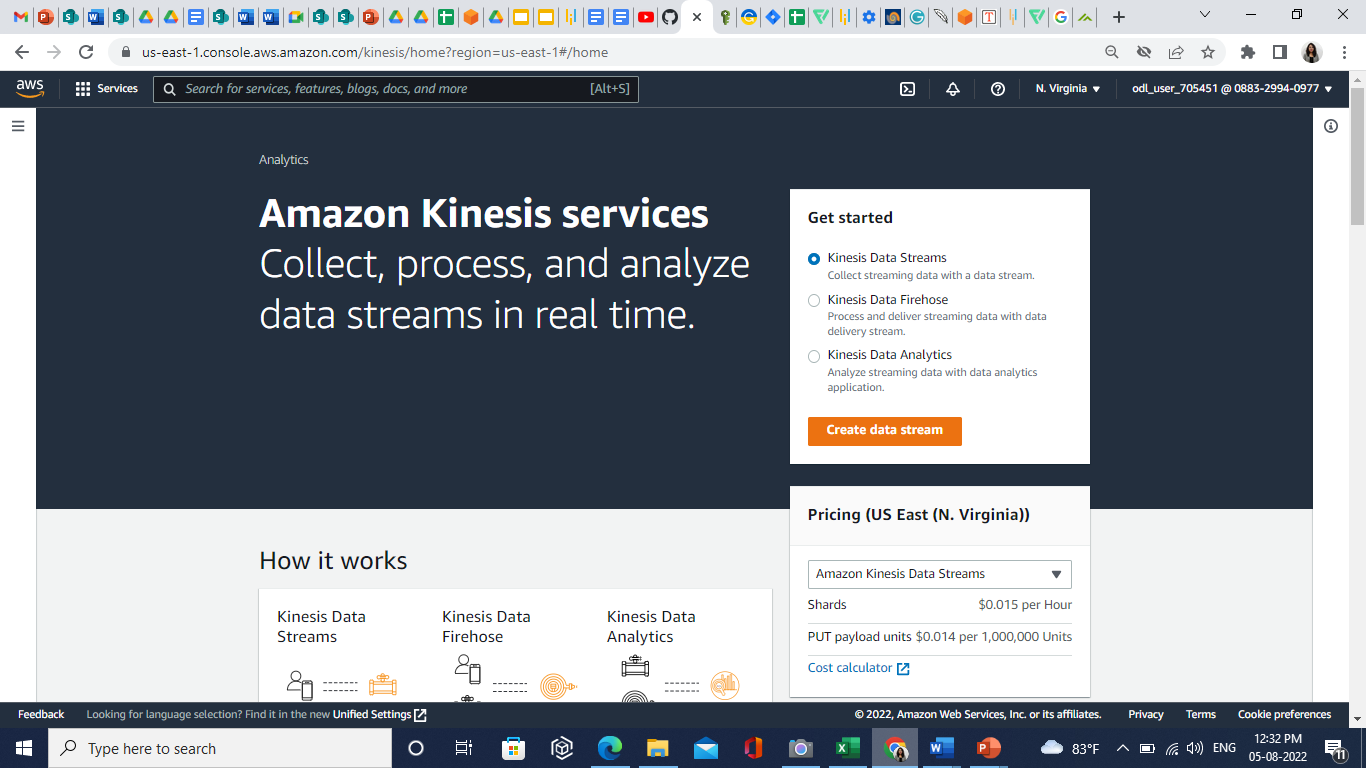
**Step1: Create an AWS Kinesis Data Streams**

* 1. Open the AWS dashboard, search and select the **Kinesis**.

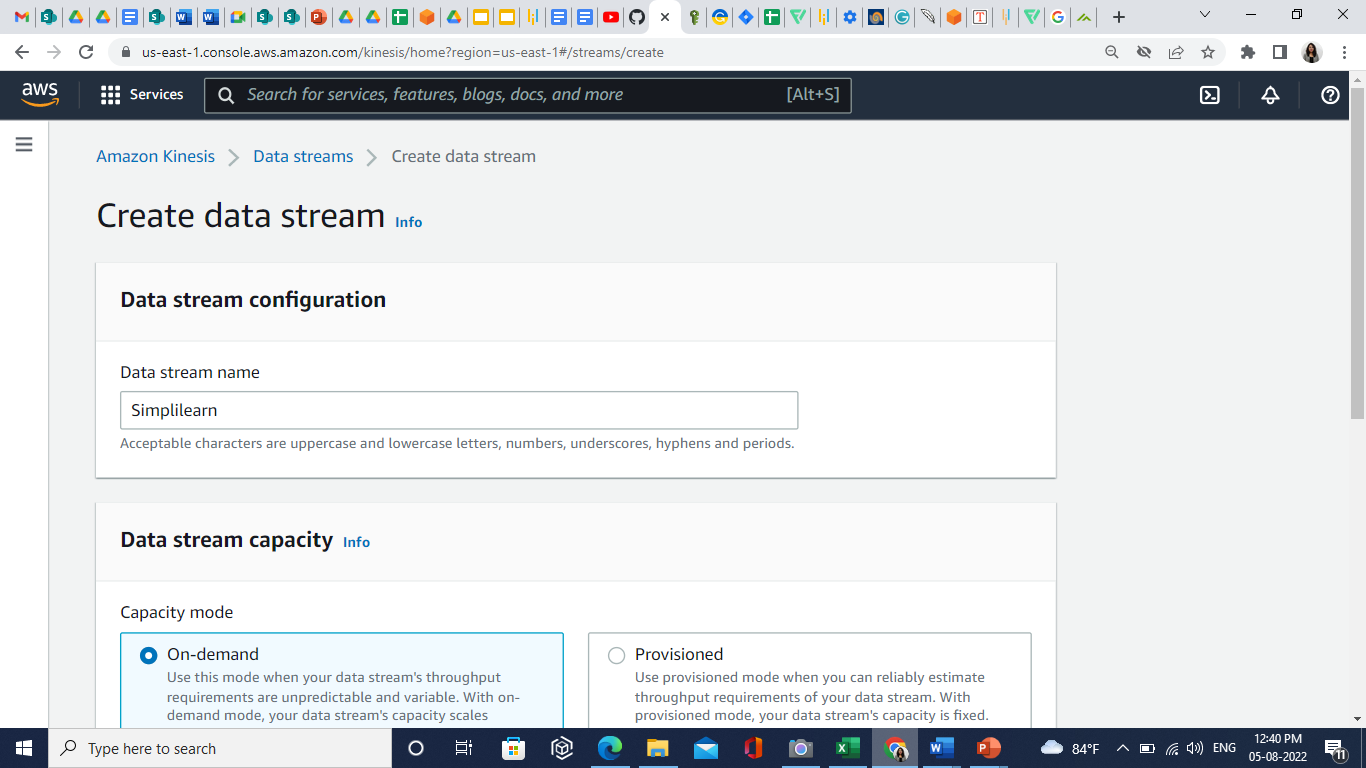
A screenshot of a computer

Description automatically generated

* 1. Click on the **Create data stream** button.



* 1. In the **Create data stream** section, enter the **Data stream name.**



* 1. In the **Data stream capacity** section, select the **On-demand** option.

Graphical user interface, text, application, email

Description automatically generated

* 1. Click on the **Create data stream** button.

Graphical user interface, text, application

Description automatically generated

**Step 2: Create an AWS Lambda Function**

* 1. In the dashboard, search and select **Lambda**.

A screenshot of a computer

Description automatically generated

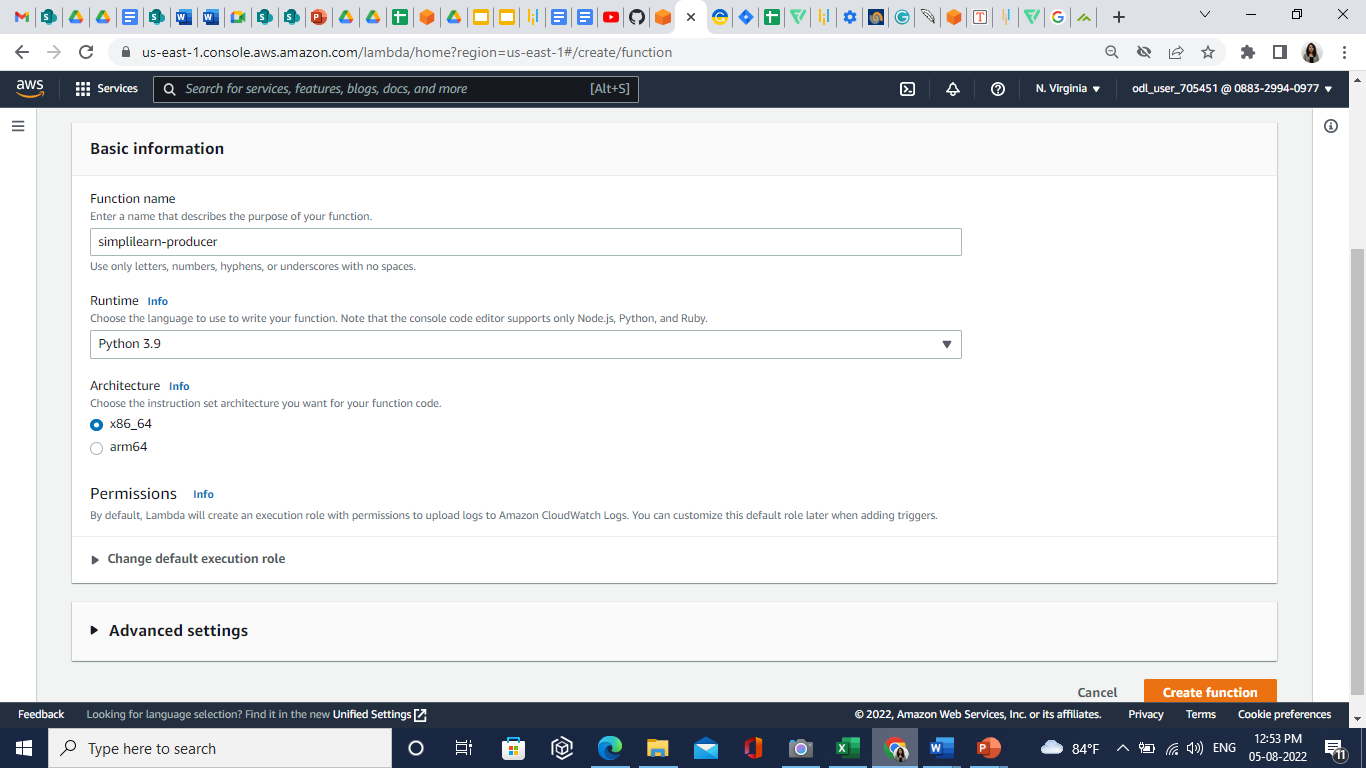
* 1. Click on the **Create function**.

A screenshot of a computer

Description automatically generated

* 1. In the **Basic information** section, specify the following values:

1. Function name: simplilearn-producer
2. Runtime: Python 3.9
3. Architecture: x86\_64



* 1. In the **Change default execution role** section, select **Create a new role with basic Lambda permissions** and click on the **Create function.**

A screenshot of a computer

Description automatically generated

* 1. In the dashboard, search and select the **IAM**.

A screenshot of a computer

Description automatically generated

* 1. Click on the **Roles**.

A screenshot of a computer

Description automatically generated

* 1. Click on the **simplilearn-producer-role-dmretkw4** (created previously).

A screenshot of a computer

Description automatically generated

* 1. Select the **Permissions** tab, and in the **Add** **Permissions** click on the **Attach policies.**

A screenshot of a computer

Description automatically generated

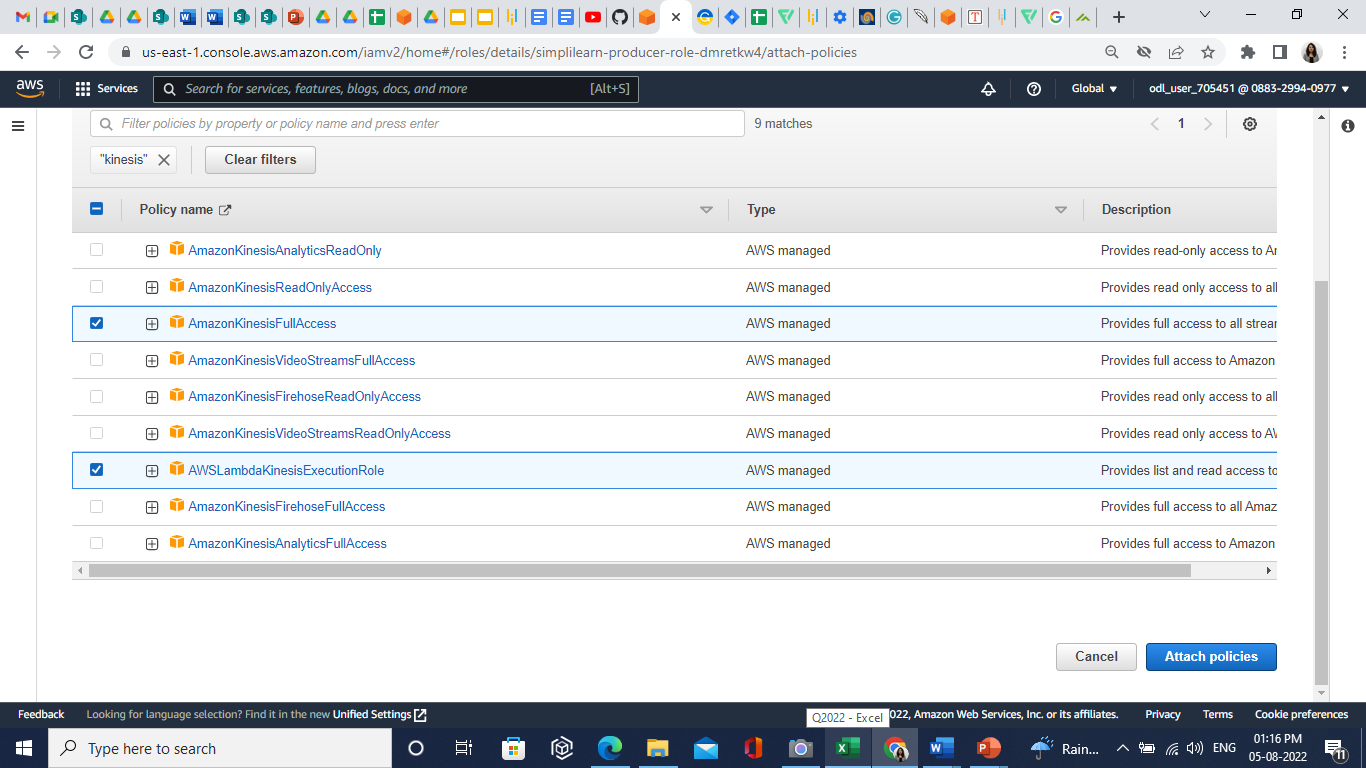
* 1. In the **Other permissions policies**,search **Kinesis** and select the below policies:

1. **AmazonKinesisFullAccess**
2. **AWSLambdaKinesisExecutionRole**

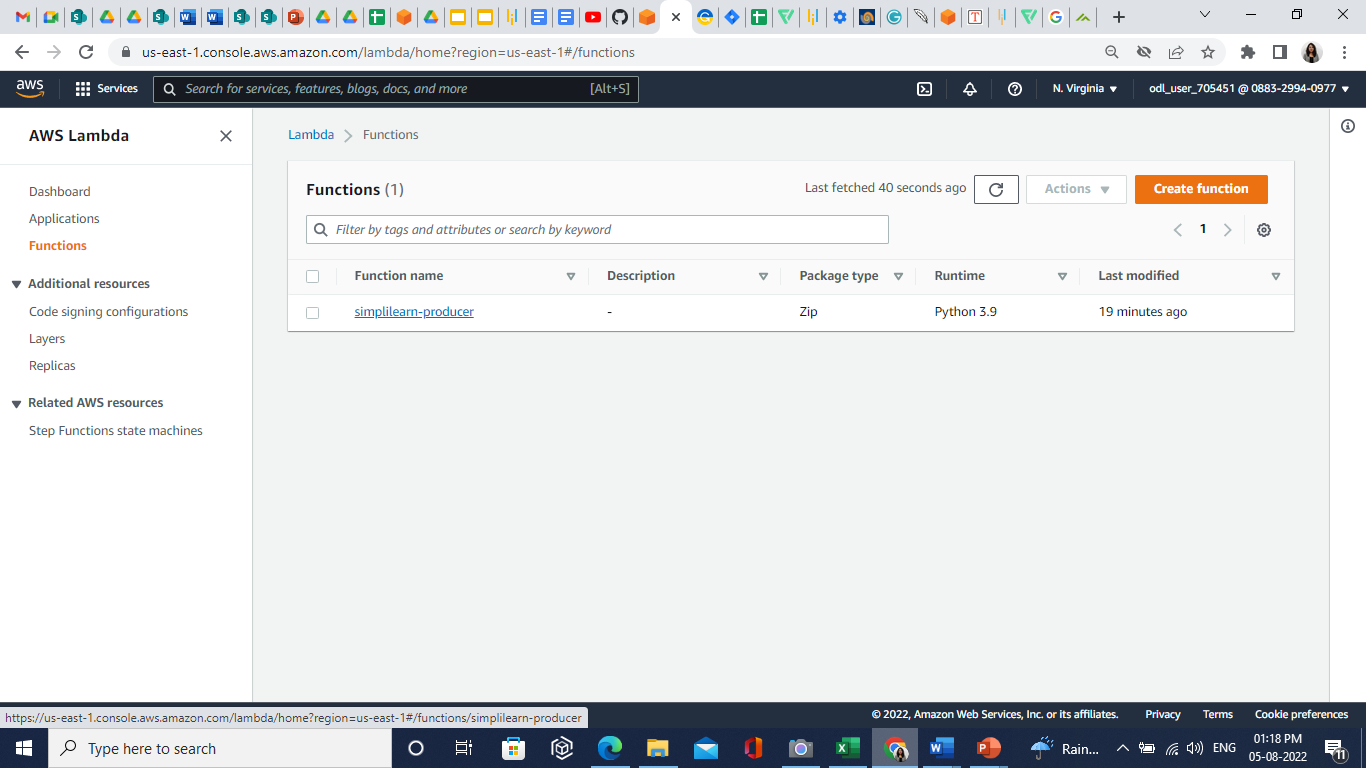
A screenshot of a computer

Description automatically generated

* 1. Click on the **Attach policies**.



* 1. Go back to the **Lambda** andselect the created function.



* 1. In the code tab, enter the following code:

**import json**

**import boto3**

**import uuid**

**def lambda\_handler(event, context):**

**client = boto3.client('Simplilearn')**

**response = client.put\_record(**

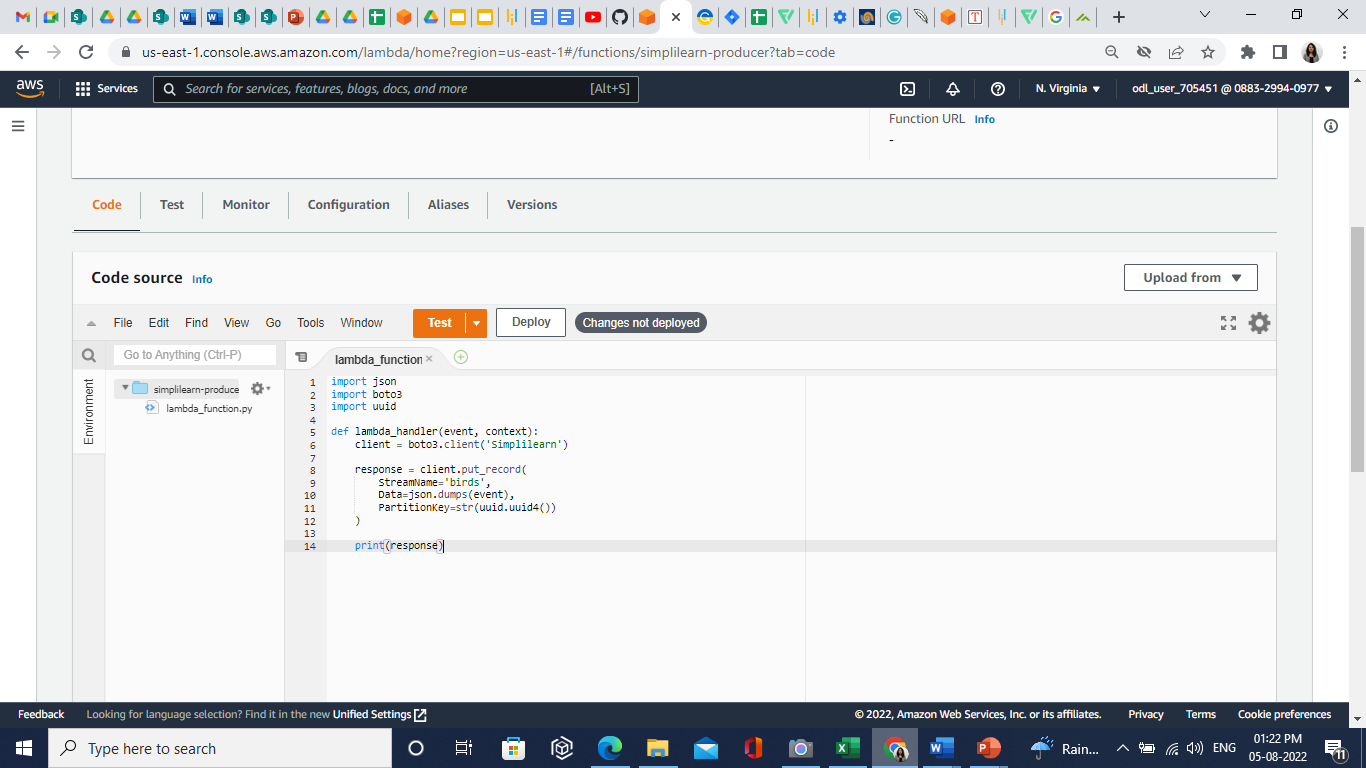
**StreamName='birds',**

**Data=json.dumps(event),**

**PartitionKey=str(uuid.uuid4())**

**)**

**print(response)**





**Note:**

Replace Streamname with data stream name.

* 1. Click on the **Deploy** button.



* 1. Click on the **Test** button.
  2. Enter the **Event name**.

Graphical user interface, text, application, email

Description automatically generated

* 1. In the **Even JSON,** enter the following script:

**{**

**"id": "b2bfe236-c738-4916-9015-ce47d49d666d",**

**"canFly": true,**

**"name": "blue jay",**

**"colors": [**

**"black",**

**"white",**

**"blur",**

**"grey"**

**]**

**}**

A screenshot of a computer

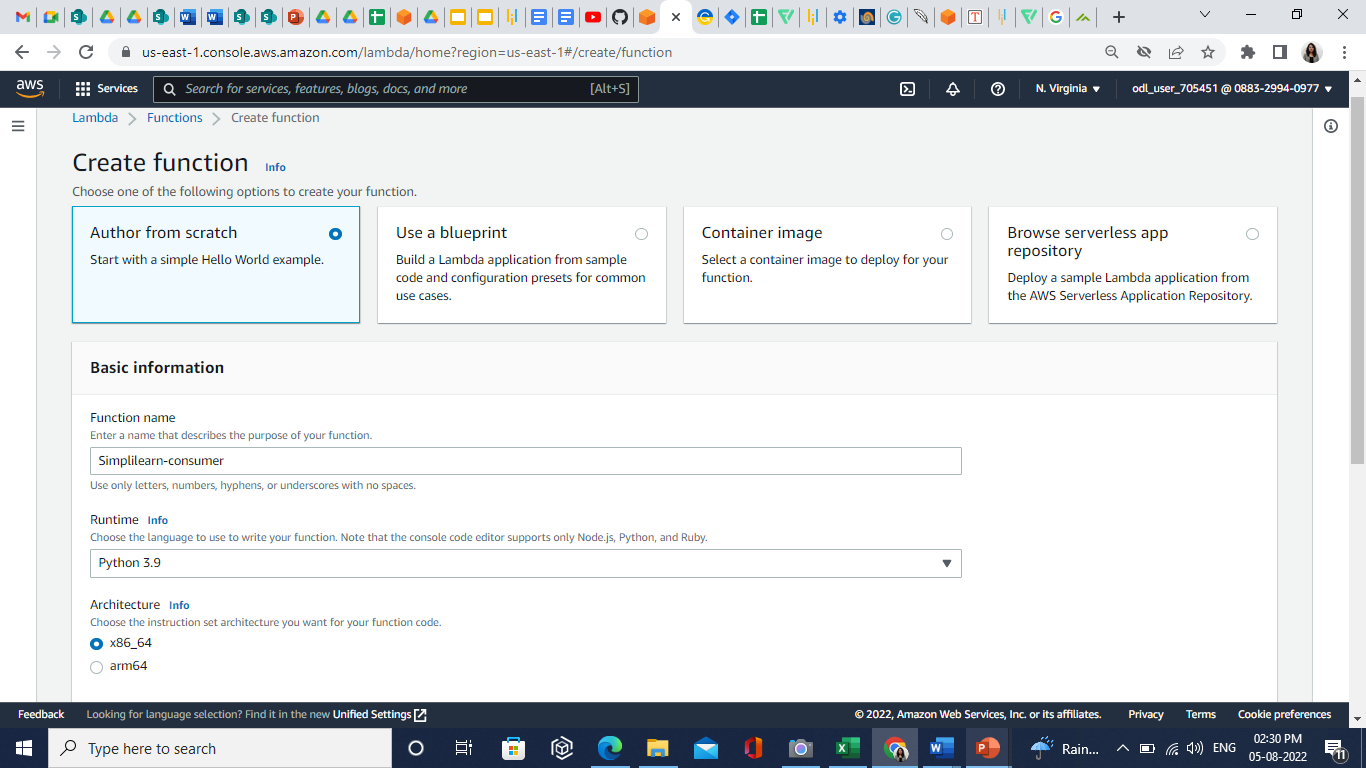
Description automatically generated

* 1. Click on the **Save** button.
  2. Click on the **Test** button again.

A screenshot of a computer

Description automatically generated

* 1. Create one more Lambda function named **Simplilearn-producer**, using the previous procedure.



A screenshot of a computer

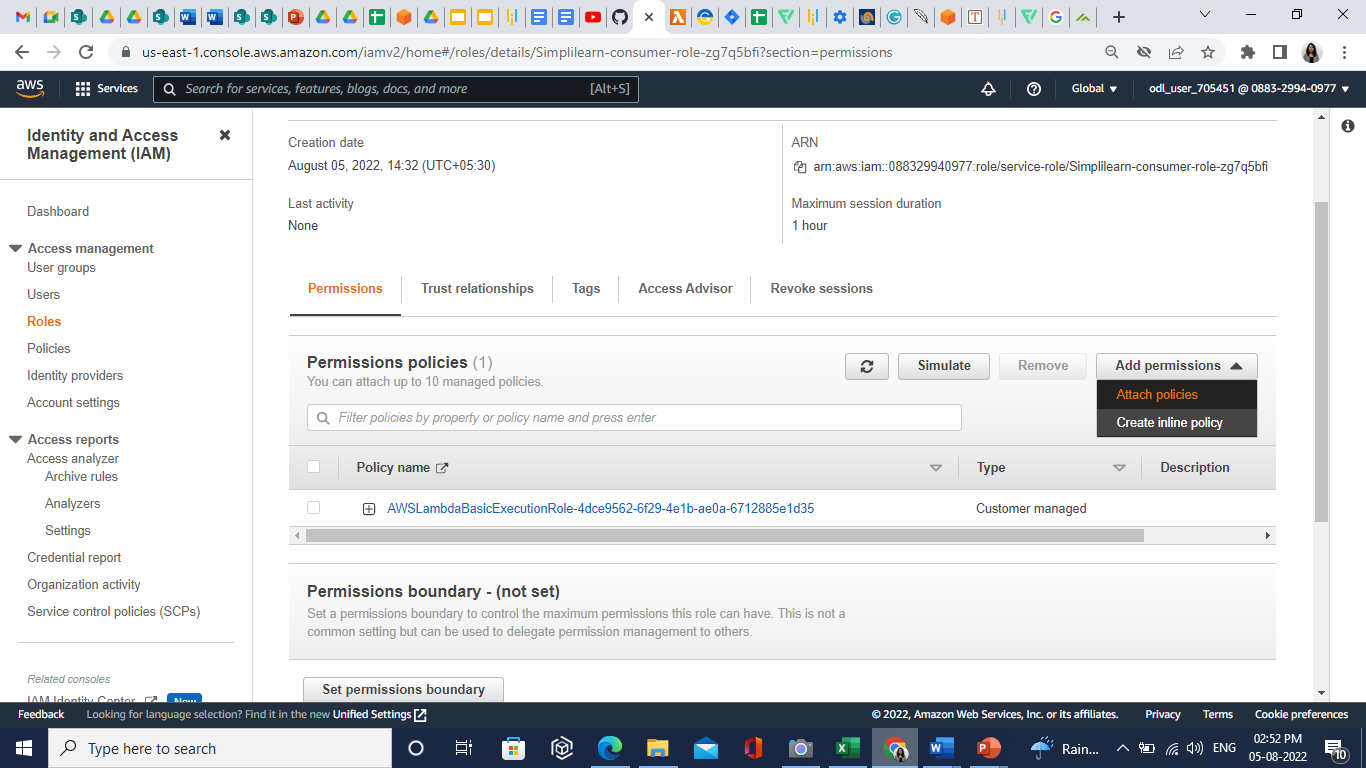
Description automatically generated

* 1. Go to **IAM Roles** andselect **Simplilearn-consumer-role-zg7q5bfi.**

A screenshot of a computer

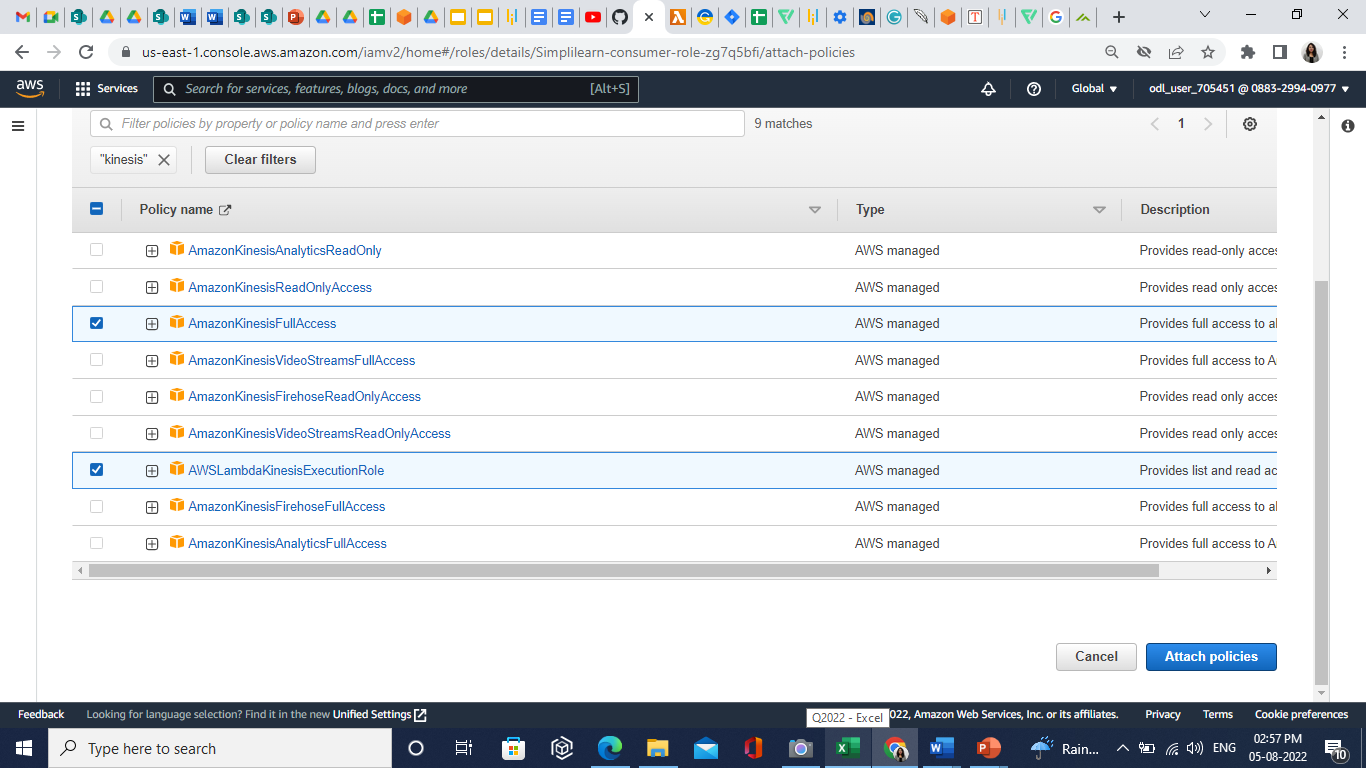
Description automatically generated

* 1. Select the **Add permissions** and click on the **Attach policies.**



* 1. Attach the following policies:

1. **AmazonKinesisFullAccess**
2. **AWSLambdaKinesisExecutionRole**
3. **AmazonDynamoDBFullAccess**



A screenshot of a computer

Description automatically generated

* 1. Click on the **Attach policies**.

**Step 3: Create an AWS DynamoDB Database**

* 1. In the dashboard, search and select the **DynamoDB**.

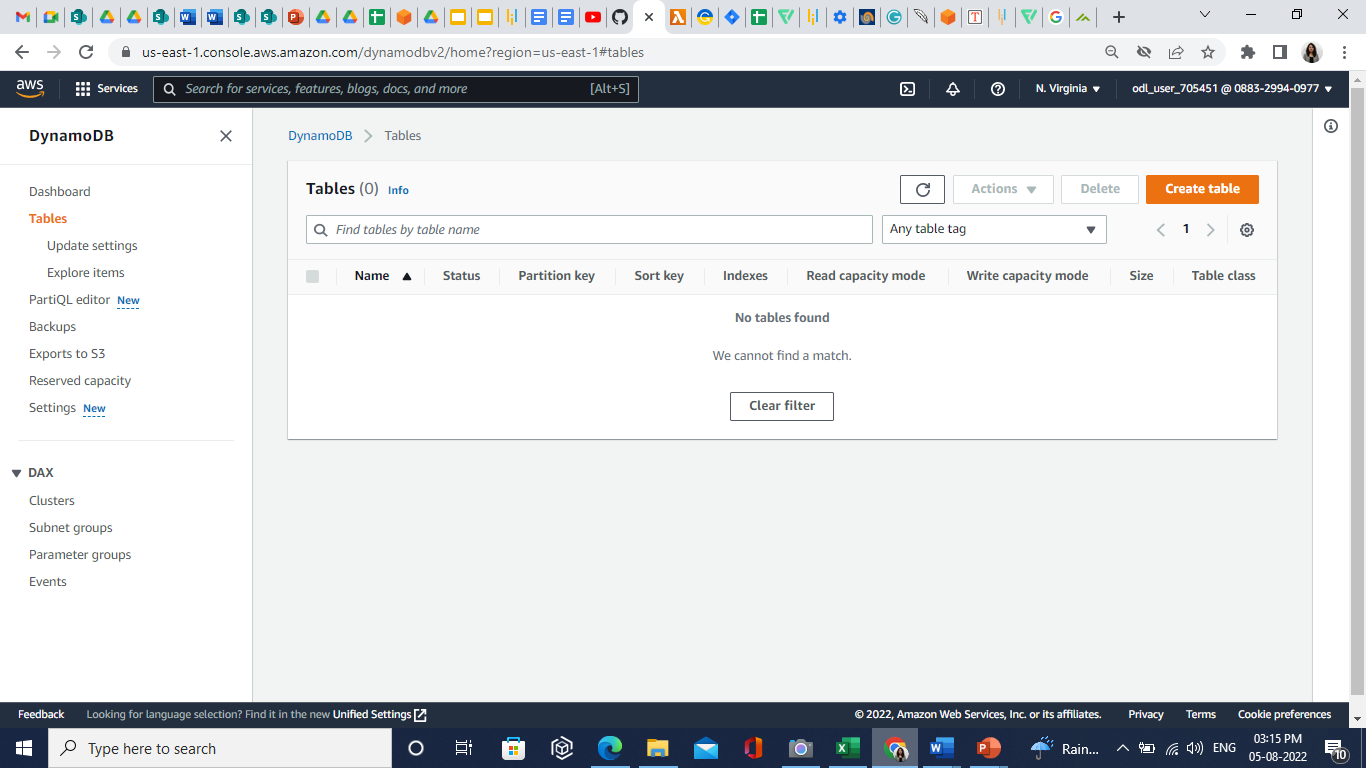
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* 1. Click on the **Create table**.



* 1. In the **Table details**, enter the following values:

1. Table name: Simplilearn-table
2. Partition key: id

Graphical user interface, text, application, Word

Description automatically generated

* 1. Click on the **Create table**.
  2. Go to the **Lambda function** and select the **Simplilearn-consumer**.
  3. In the **Code** section, enter the following code:

**import base64**

**import json**

**import boto3**

**def lambda\_handler(event, context):**

**for record in event['Records']:**

**#Kinesis data is base64 encoded so decode here**

**payload=base64.b64decode(record["kinesis"]["data"])**

**res = json.loads(payload)**

**write\_to\_db(res)**

**print("Object successfully stored in DB.")**

**def write\_to\_db(data):**

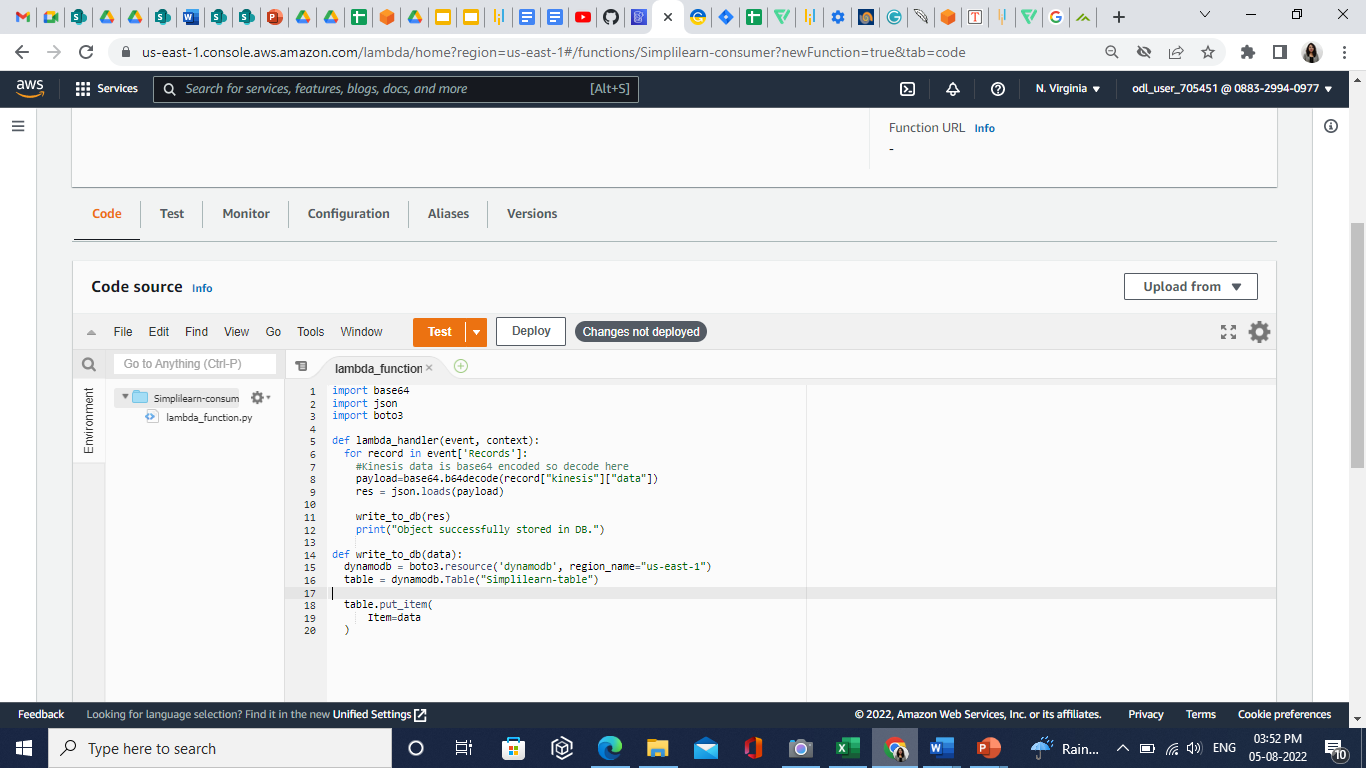
**dynamodb = boto3.resource('dynamodb', region\_name="eu-central-1")**

**table = dynamodb.Table("endres-test")**

**table.put\_item(**

**Item=data**

**)**





**Note:**

Replace **table = dynamodb.Table("endres-test")** table name with dynamodb table name. Replace the region name with current region.

* 1. Click on the **Deploy**.
  2. In the **Simplilearn-consumer,** click on the **Add trigger**.

A screenshot of a computer

Description automatically generated

* 1. Search and select the **Kinesis.**

Graphical user interface, text, application, Word

Description automatically generated

* 1. In the **Kinesis stream** select the **Simplilearn**.

Graphical user interface, text, application, Word

Description automatically generated

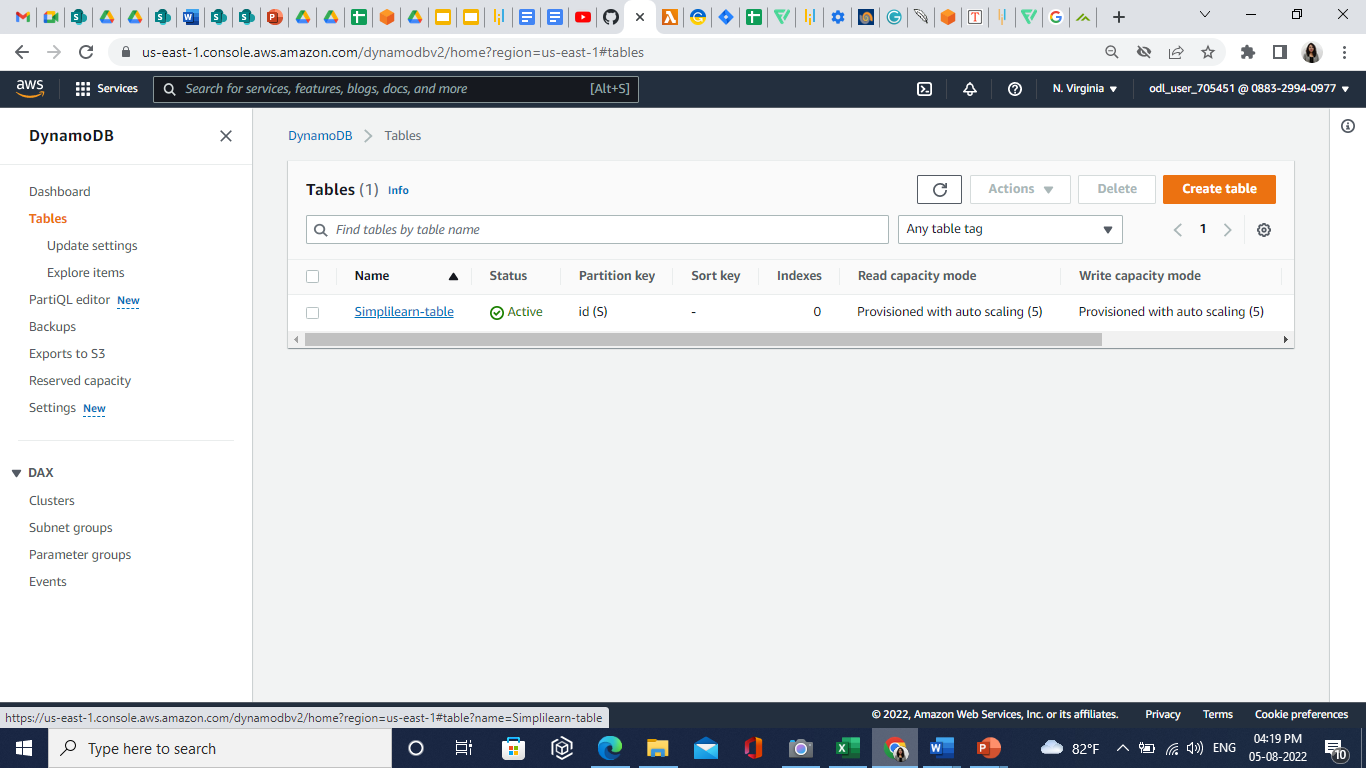
* 1. Click on **Add** option.

A screenshot of a computer

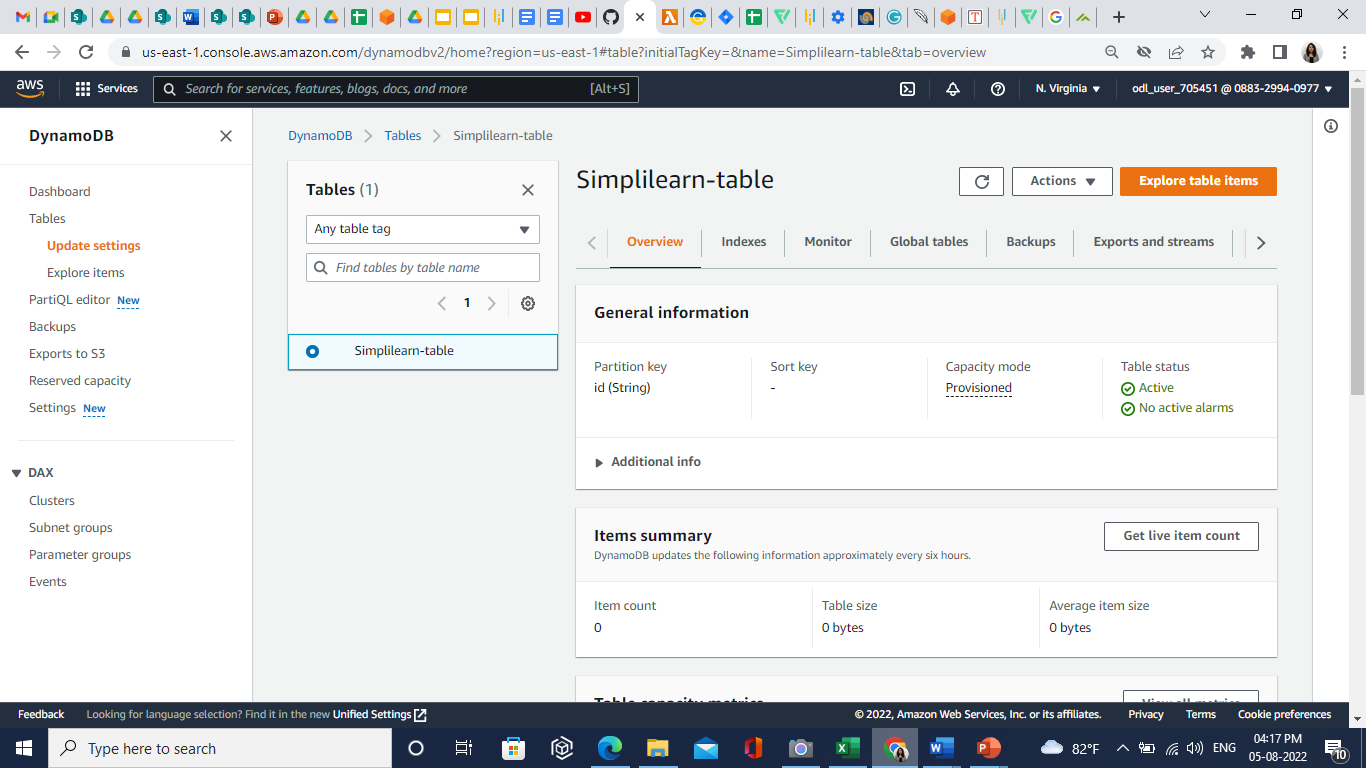
Description automatically generated

**Step 4: Perform Scan Operation**

* 1. Go to the **DynamoDB** andclick on the **Simplilearn-table**.



* 1. Click on the **Explore table items**.

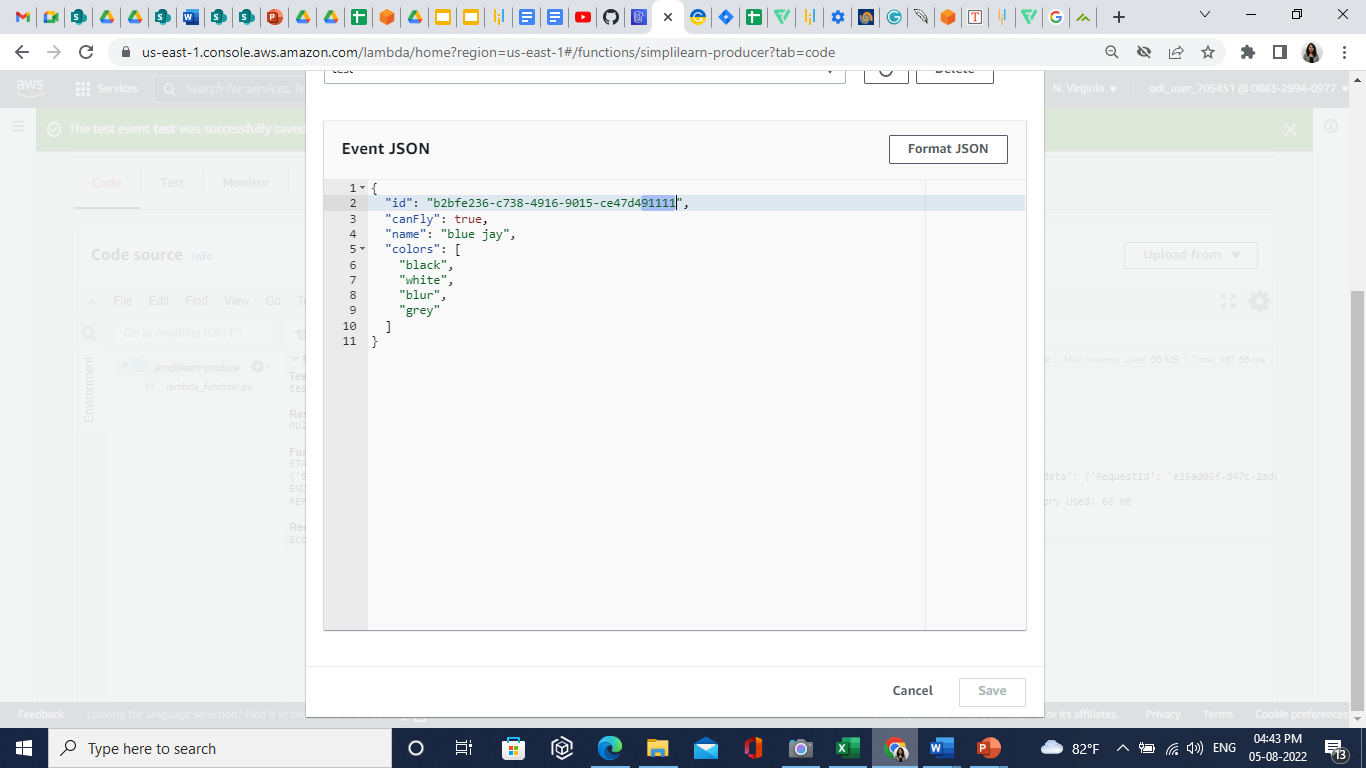


* 1. Go back to the **simplilearn-producer** Lambda function, click on **Test,** and select **Configure test event**.

A screenshot of a computer

Description automatically generated

* 1. In the **id**, change last 4-5 digits and click on the **Save** button.



* 1. Go back to **DynamoDB,** click on the table, and click on the **Explore table items.**

A screenshot of a computer

Description automatically generated

* 1. Refresh the page.
  2. In the **Items returned** section, you can see the **simplilearn-producer** JSON data.

A screenshot of a computer

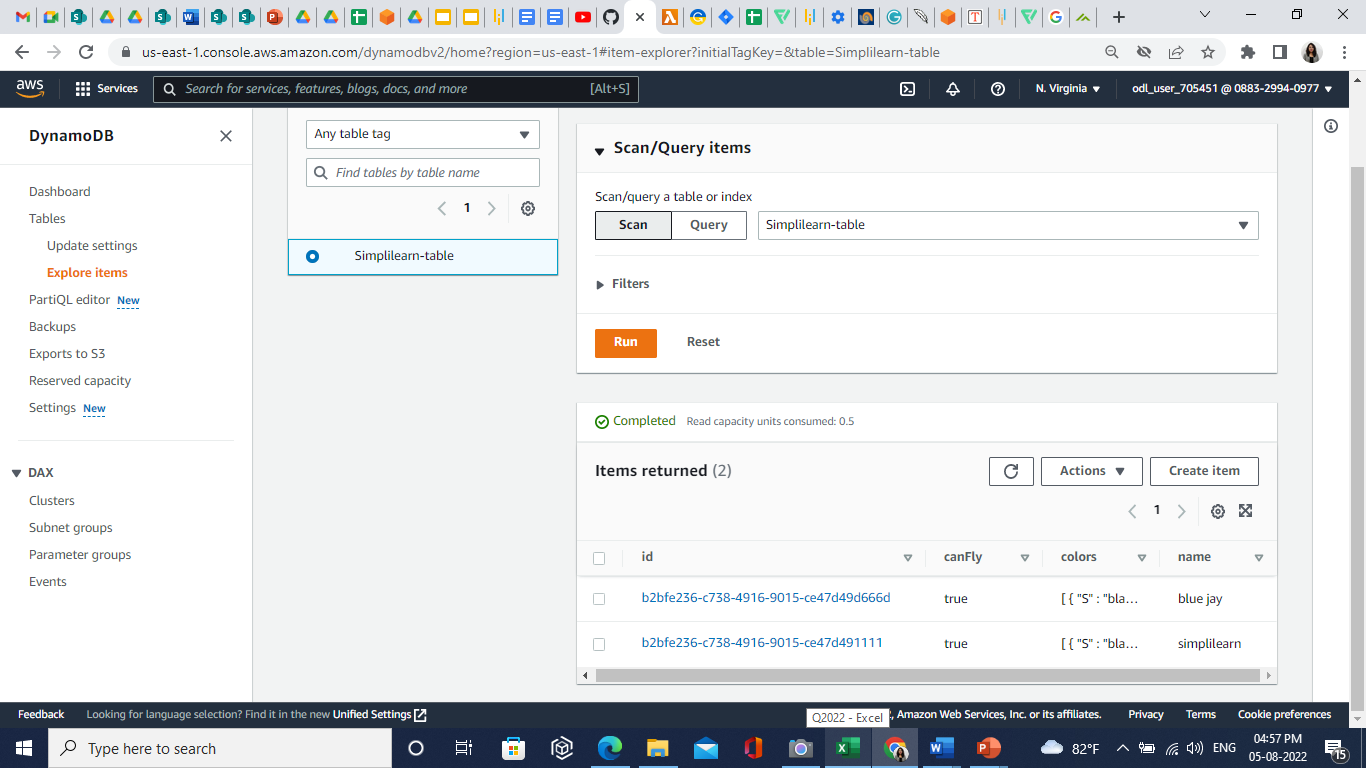
Description automatically generated

* 1. Again, go back to **simplilearn-producer**, change the **name value,** and click on the **Save** button. Then, click on the **Test**.

Graphical user interface, text, application, email

Description automatically generated

* 1. Refresh the below page. Changes done in simplilearn-producer JSON will reflect here:



* 1. In the **Scan/Query items** section, click on the **Filters,** enter the following values:

1. Attribute name: name
2. Value: simplilearn

A screenshot of a computer

Description automatically generated

* 1. Click on the **Run** button. You can see the filtered one JSON data.

A screenshot of a computer

Description automatically generated