# **Name: Abdurrahman Qureshi**

# **Roll No: 242466**

Practical No: 5

Date Of Performance: 06/09/2025

Aim: To create an automated serverless system that processes new files uploaded to an S3 bucket by triggering a Lambda function, which then has the necessary IAM permissions to interact with a DynamoDB table.

1. What is AWS Lambda?
2. What is serverless computing?
3. What languages does AWS Lambda support?
4. What are AWS DynamoDB Table?
5. Explain AWS IAM service.
6. To understand AWS Lambda, create your first Lambda functions using Python / Java / Nodejs. Create AWs Lambda function and configure a trigger for Amazon Simple Storage Service (Amazon S3). The trigger invokes your Lambda function every time that you add an object to your Amazon S3 bucket. Allow AWS Lambda to access Amazon DynamoDB Table. Create IAM role that allows full access to DynamoDB Table

[Terminate the resources after performing the practical]

ANS.1

AWS Lambda is a serverless, event-driven compute service that lets you run code for virtually any type of application or backend service without provisioning or managing servers. You simply upload your code, and Lambda automatically runs and scales it in response to triggers like HTTP requests or file uploads.

ANS.2

Serverless computing is a cloud-native development model where developers can build and run applications without managing servers, as the cloud provider handles the routine work of provisioning, maintaining, and scaling the server infrastructure. You only pay for the compute time your application consumes.

ANS.3

AWS Lambda natively supports several runtimes, including:

* Node.js (JavaScript)
* Python
* Java
* C# (.NET Core)
* Go
* Ruby

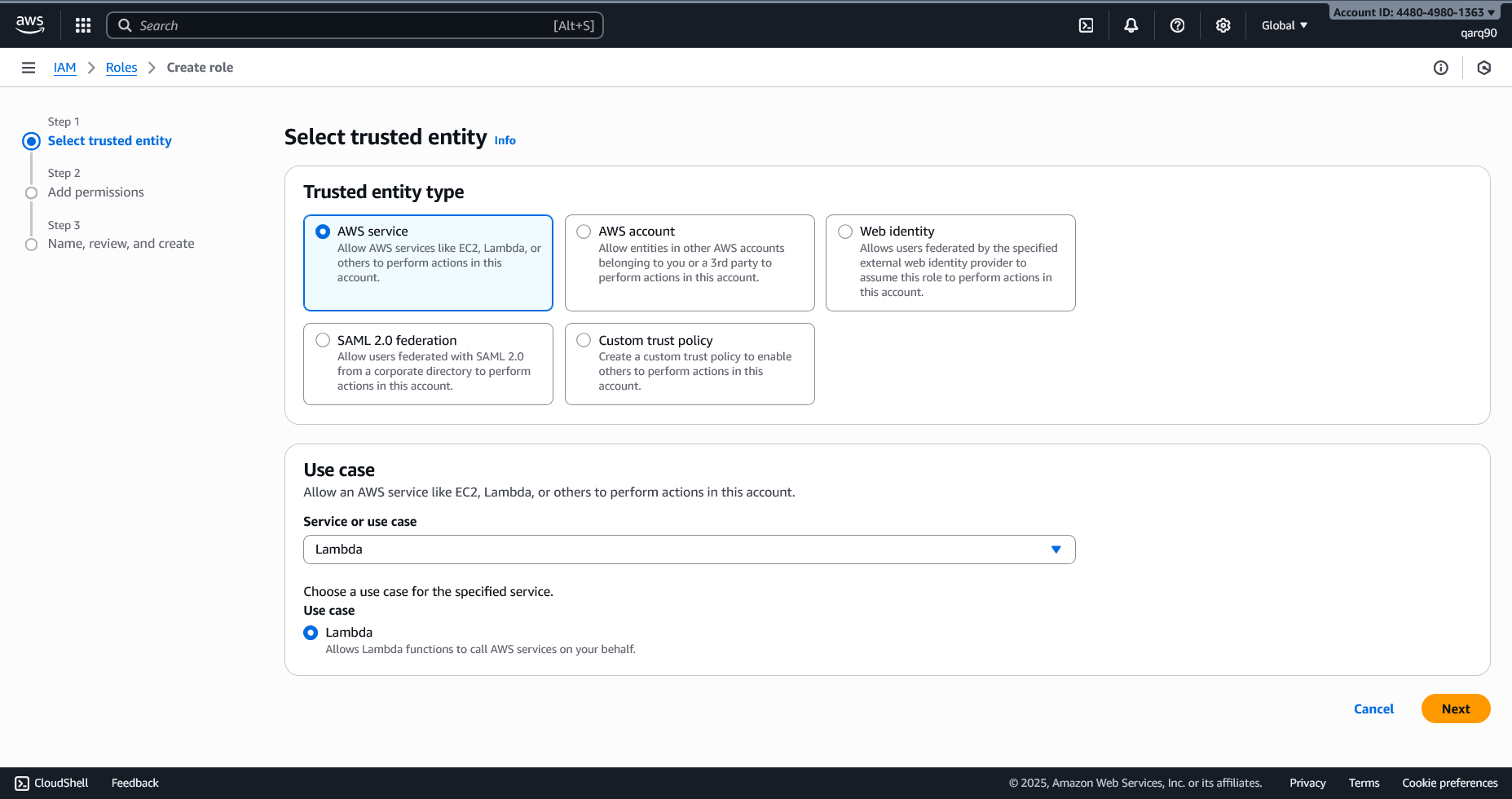
ANS.4

An AWS DynamoDB table is a fully managed, serverless NoSQL database provided by AWS that delivers reliable performance at any scale with seamless scalability and built-in security. Data is stored in items (rows) with attributes (columns), and each item is uniquely identified by a primary key.

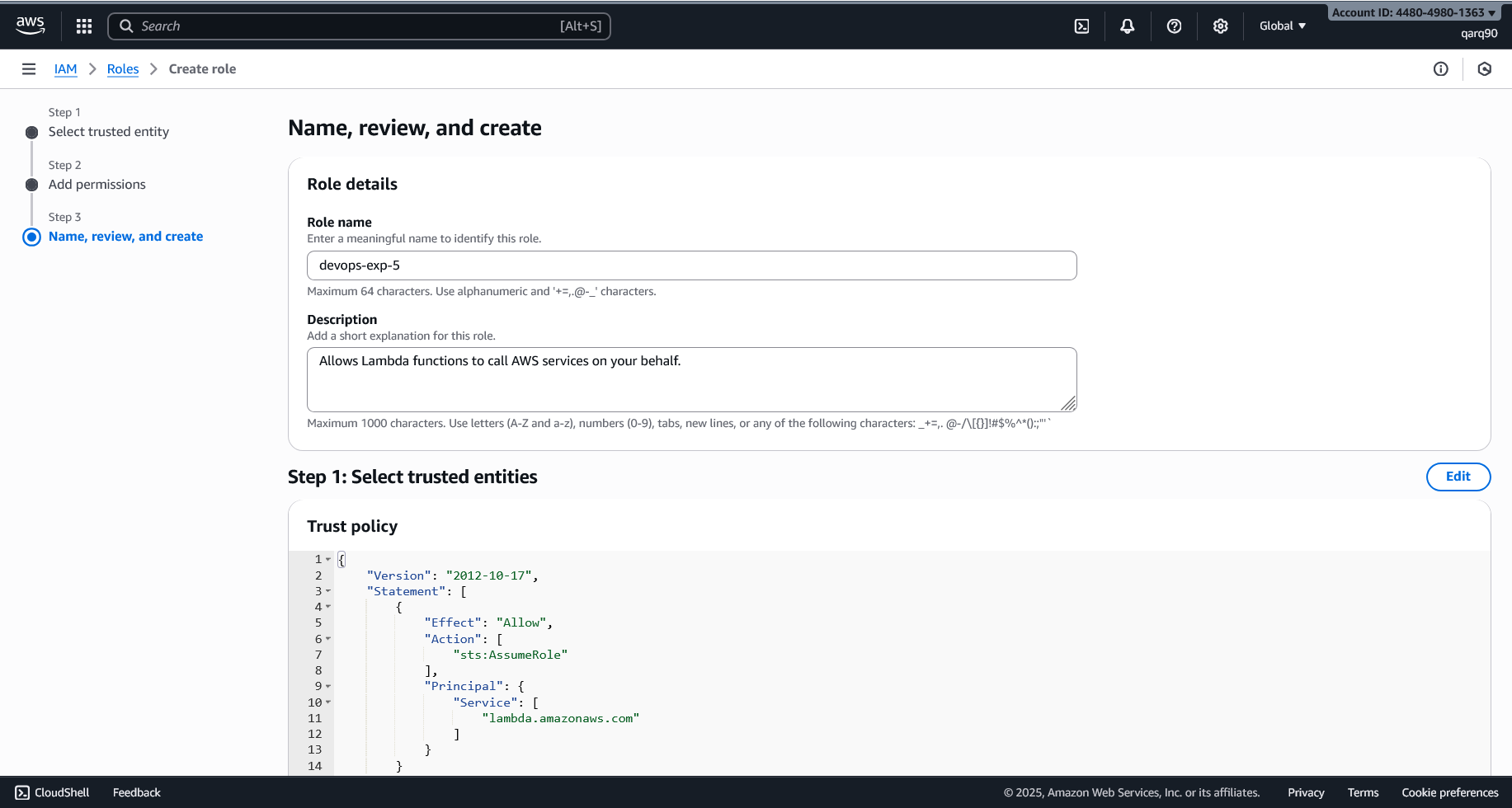
ANS.5

AWS Identity and Access Management (IAM) is a web service that helps you securely control access to AWS resources by enabling you to manage users, groups, roles, and their corresponding permissions. Its core function is to ensure the principle of least privilege, meaning users and services are granted only the permissions they need to perform their specific tasks.

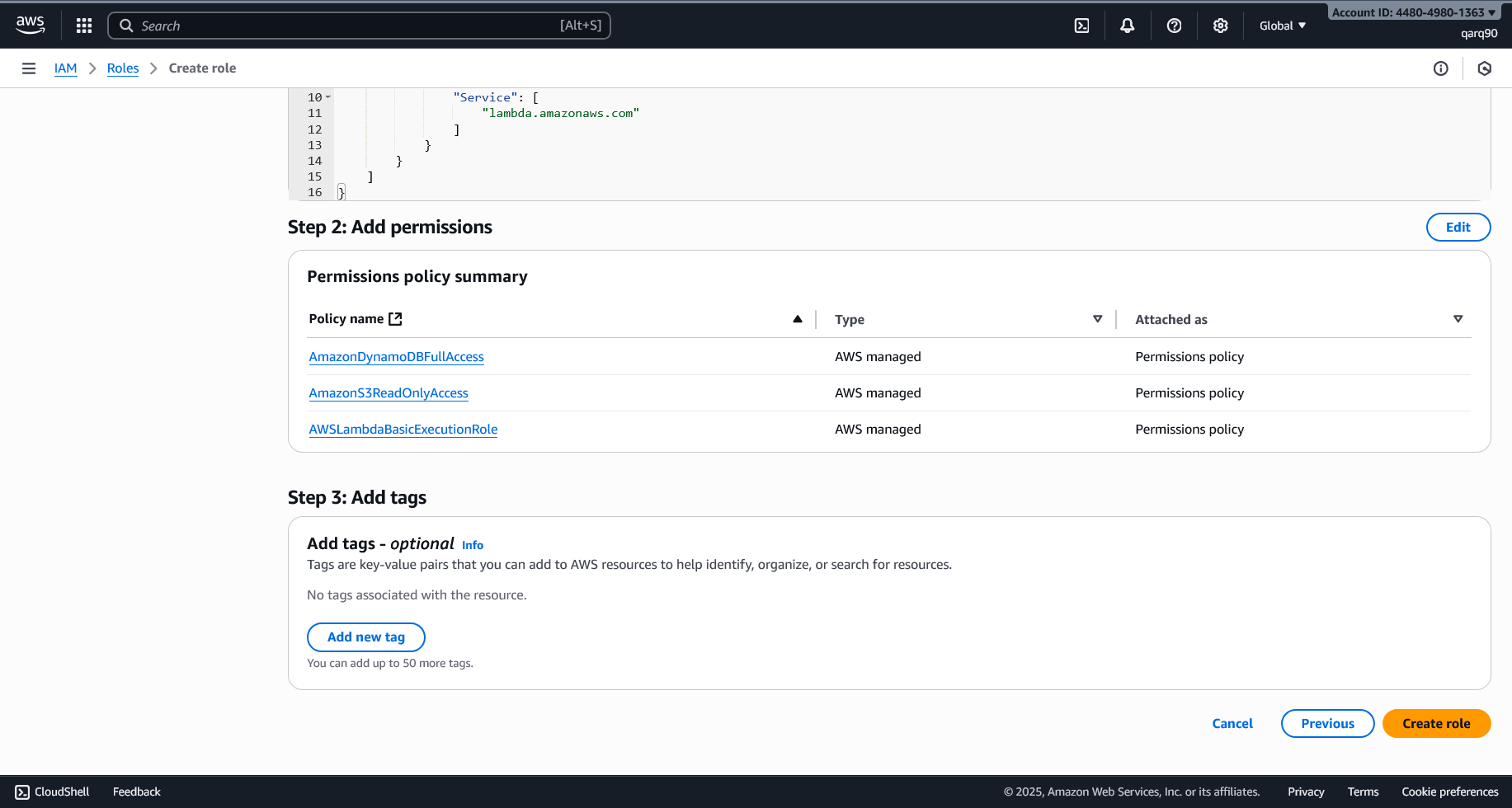
ANS.6



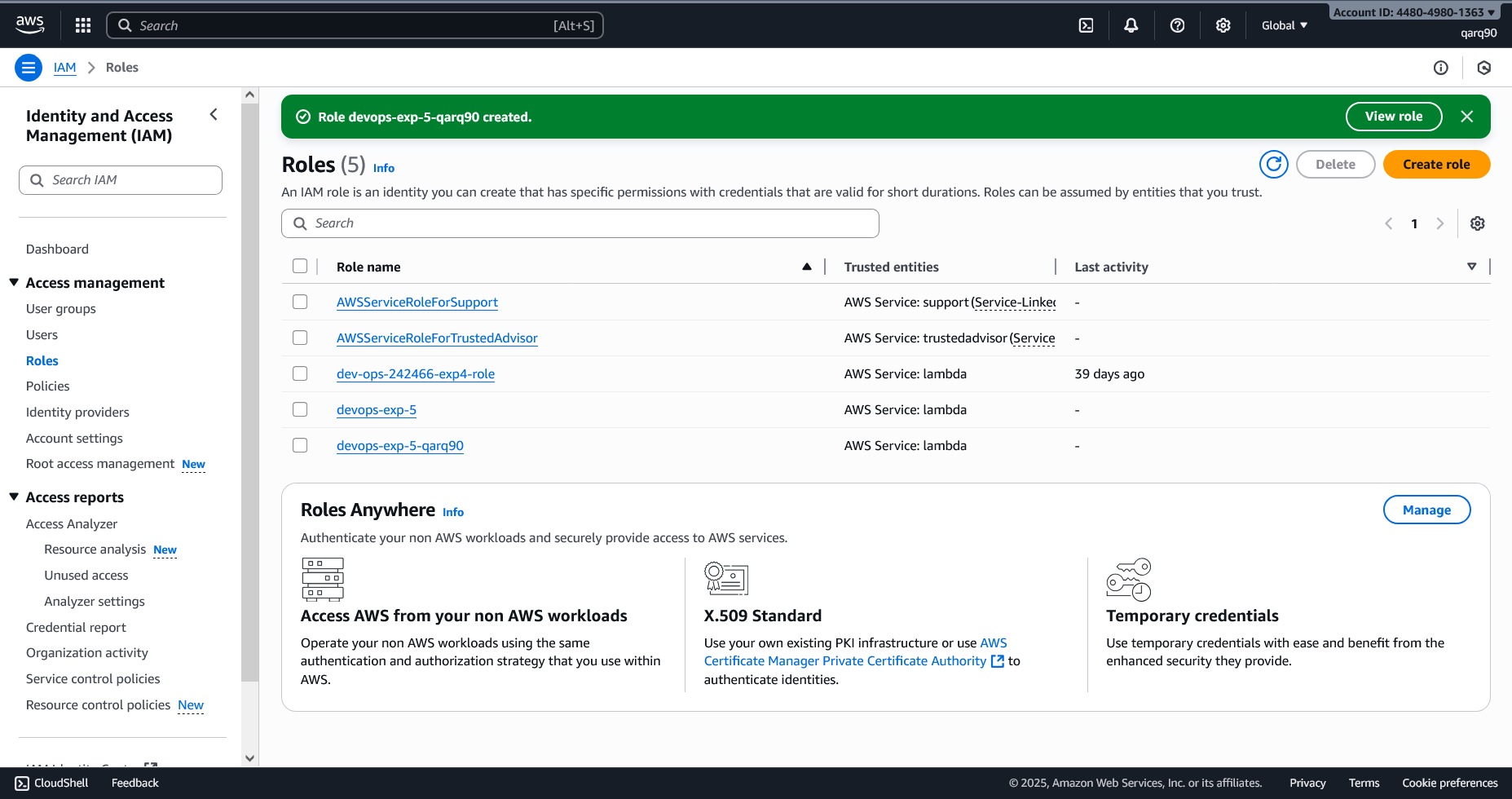
Creating an IAM role



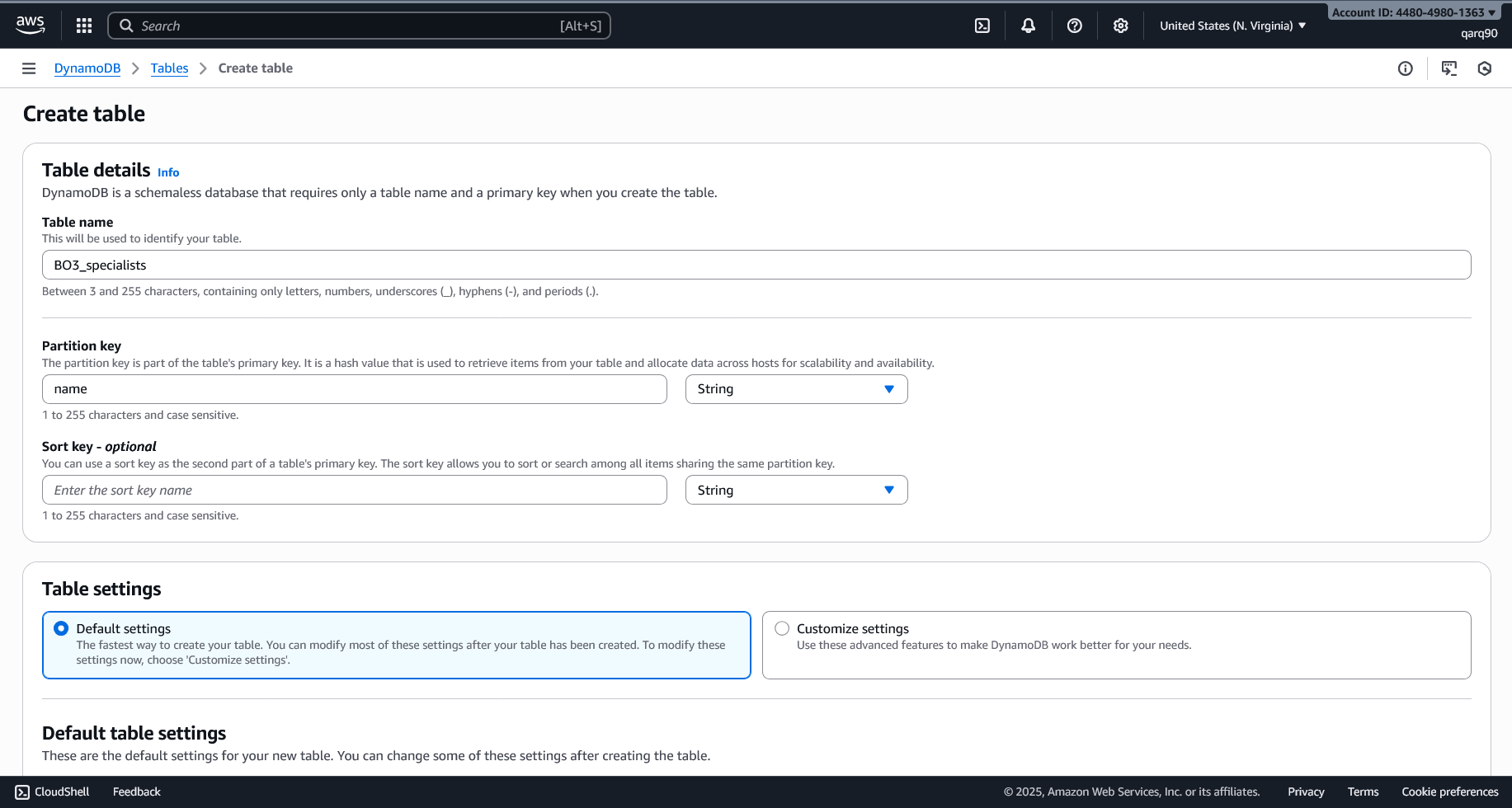
Configuring the role



Adding Permissions



Role Created Successfully



Creating the DynamoDB table

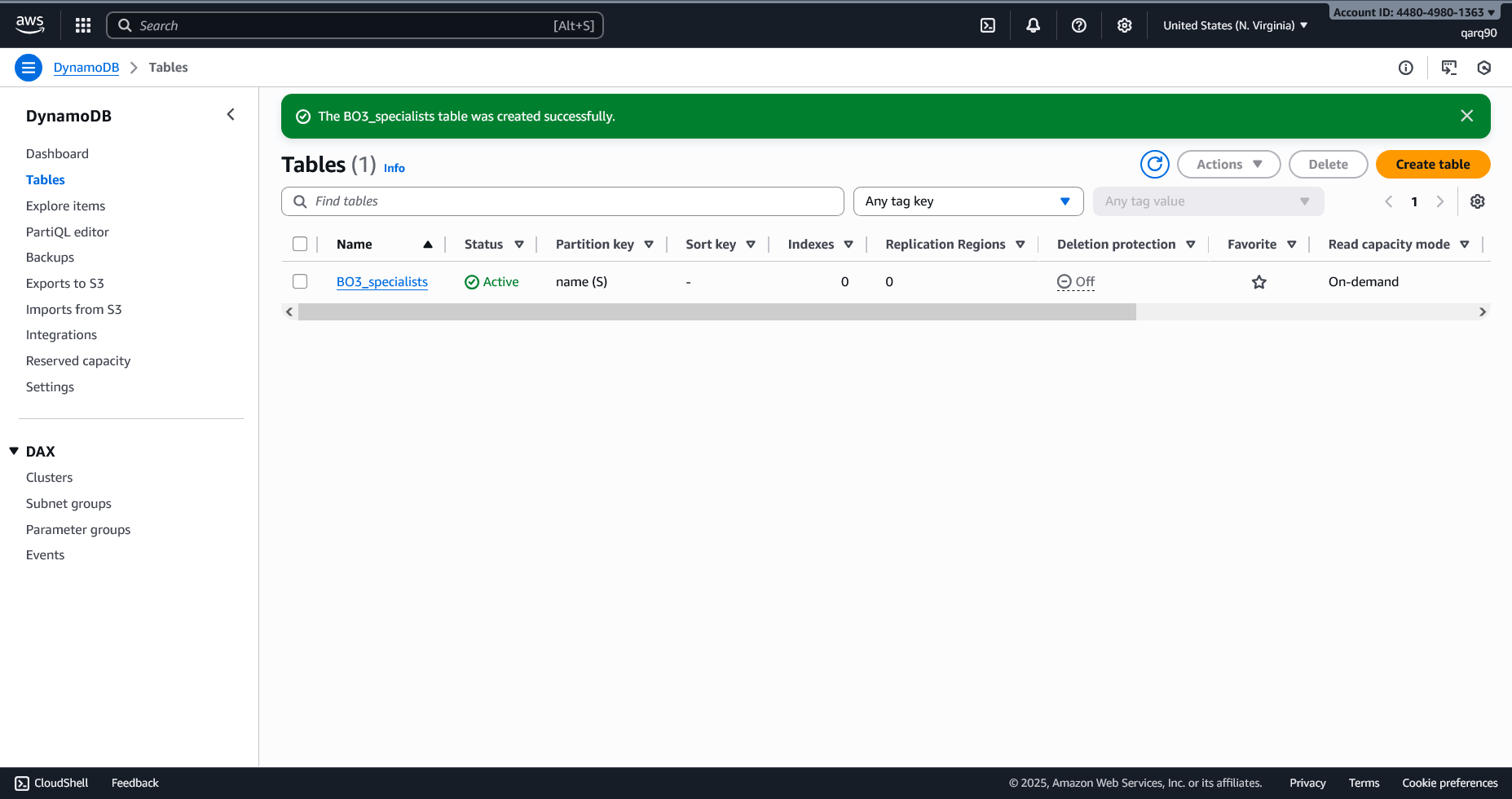
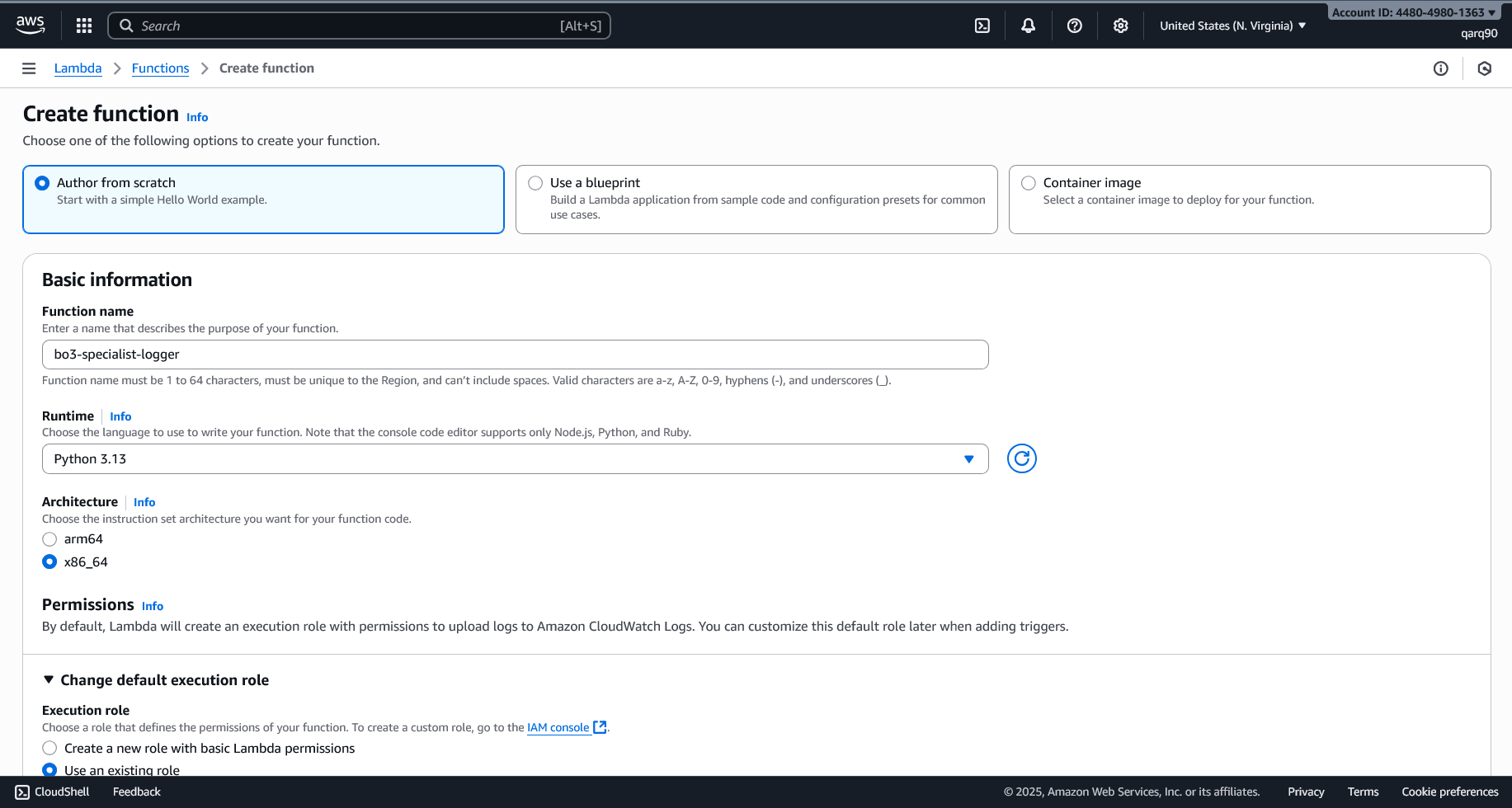
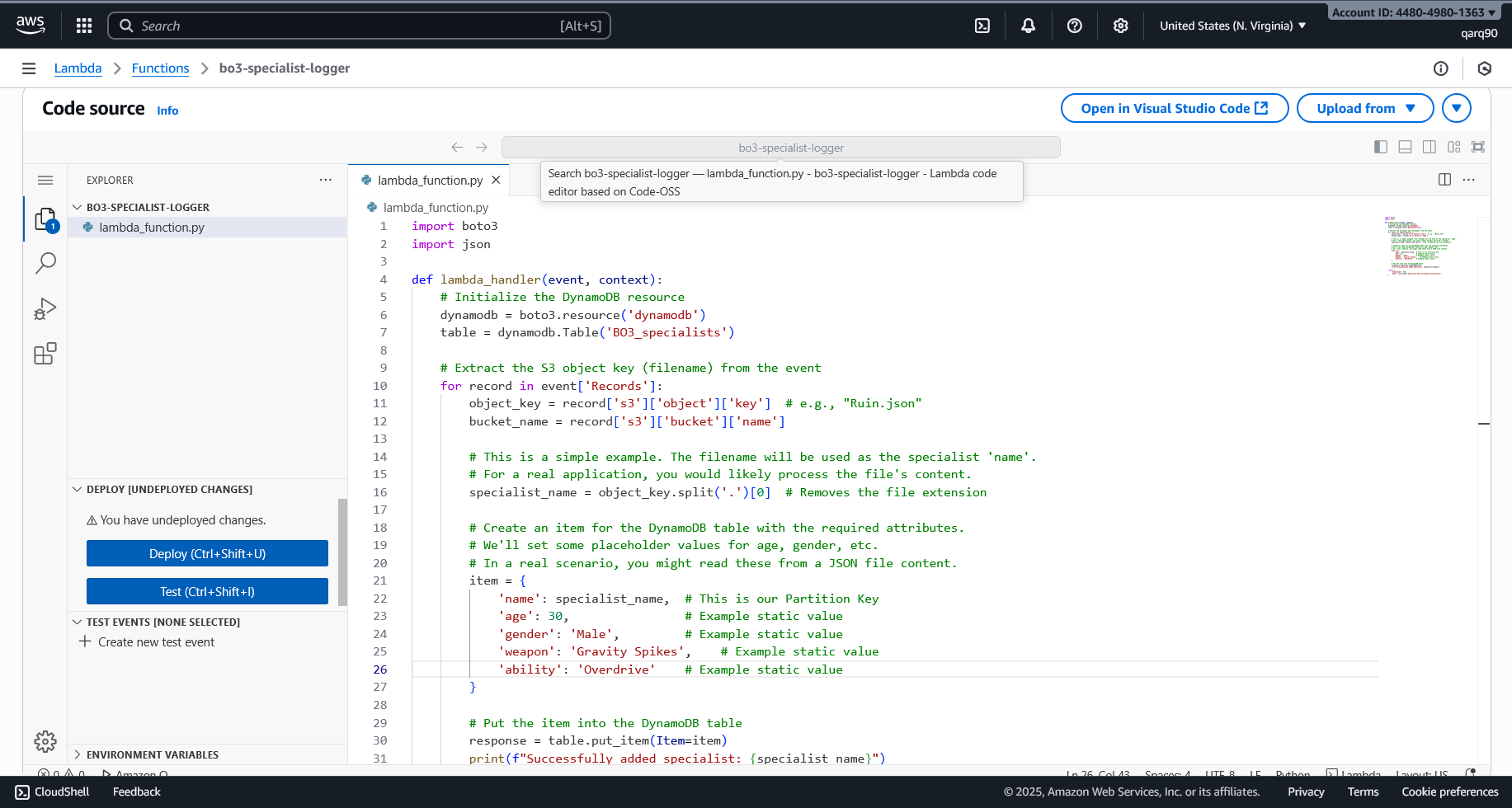


Table created Successfully



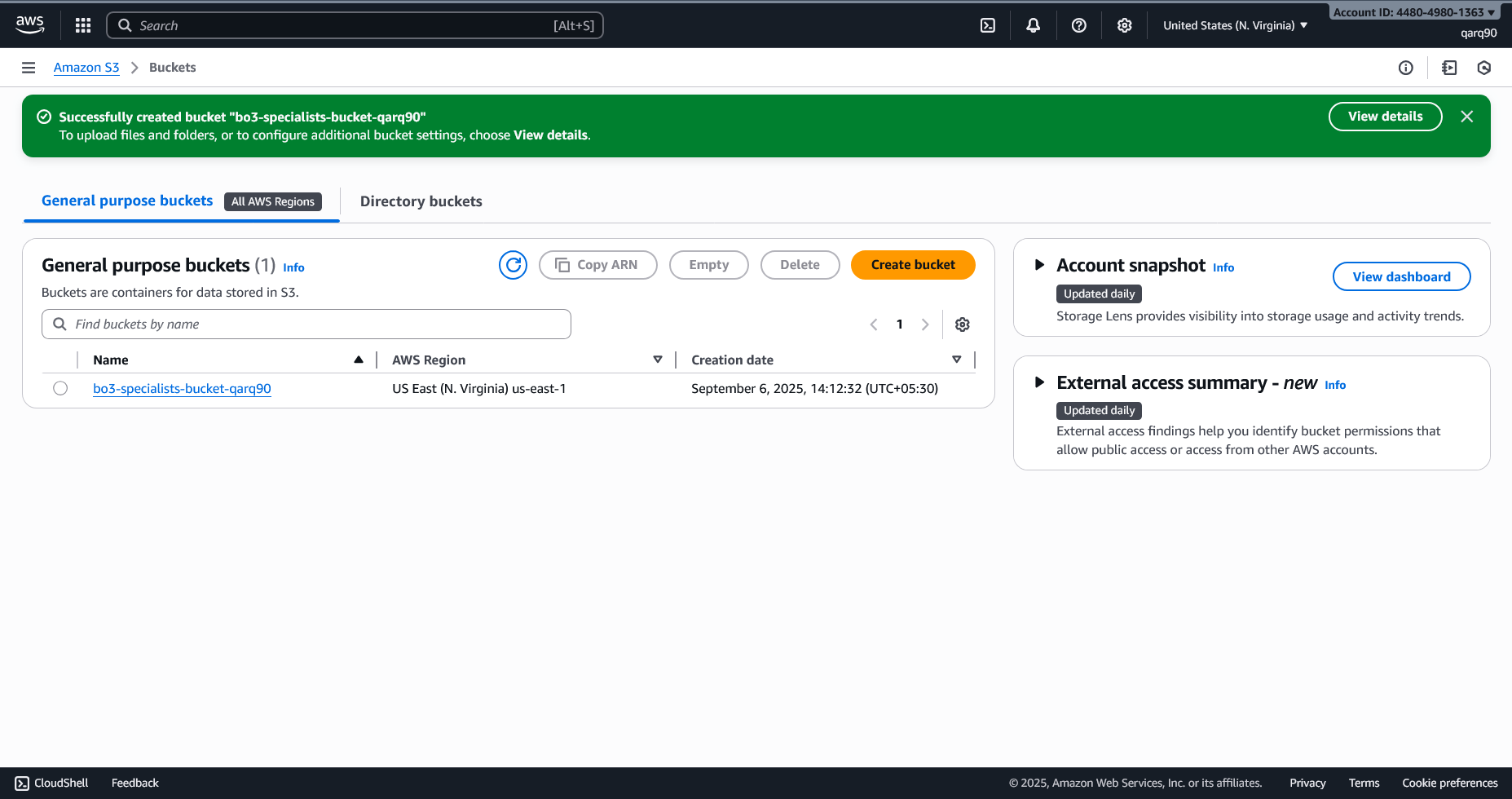
Creating the lambda function



Setting up the lambda function logic



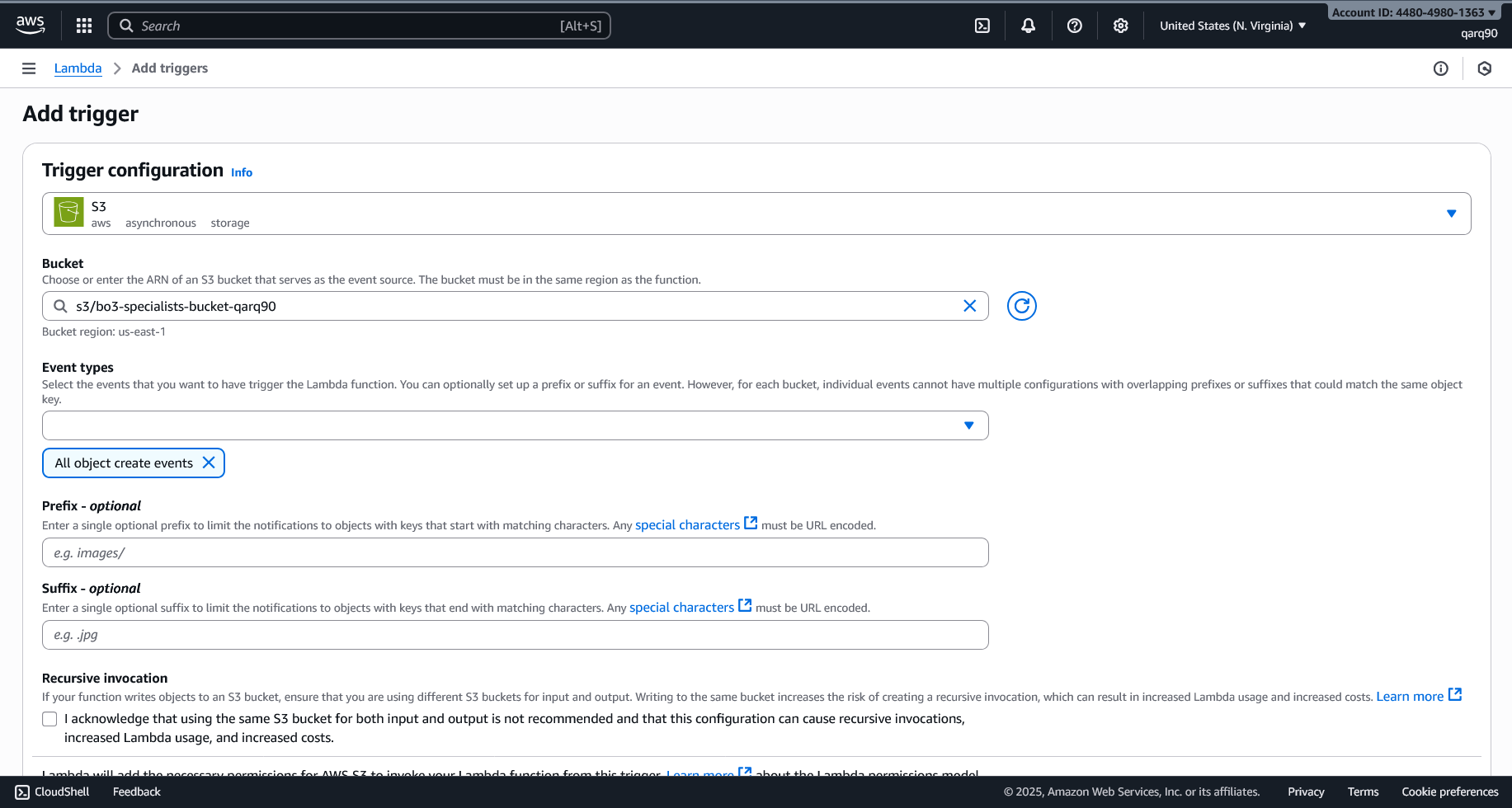
Creating the S3 bucket



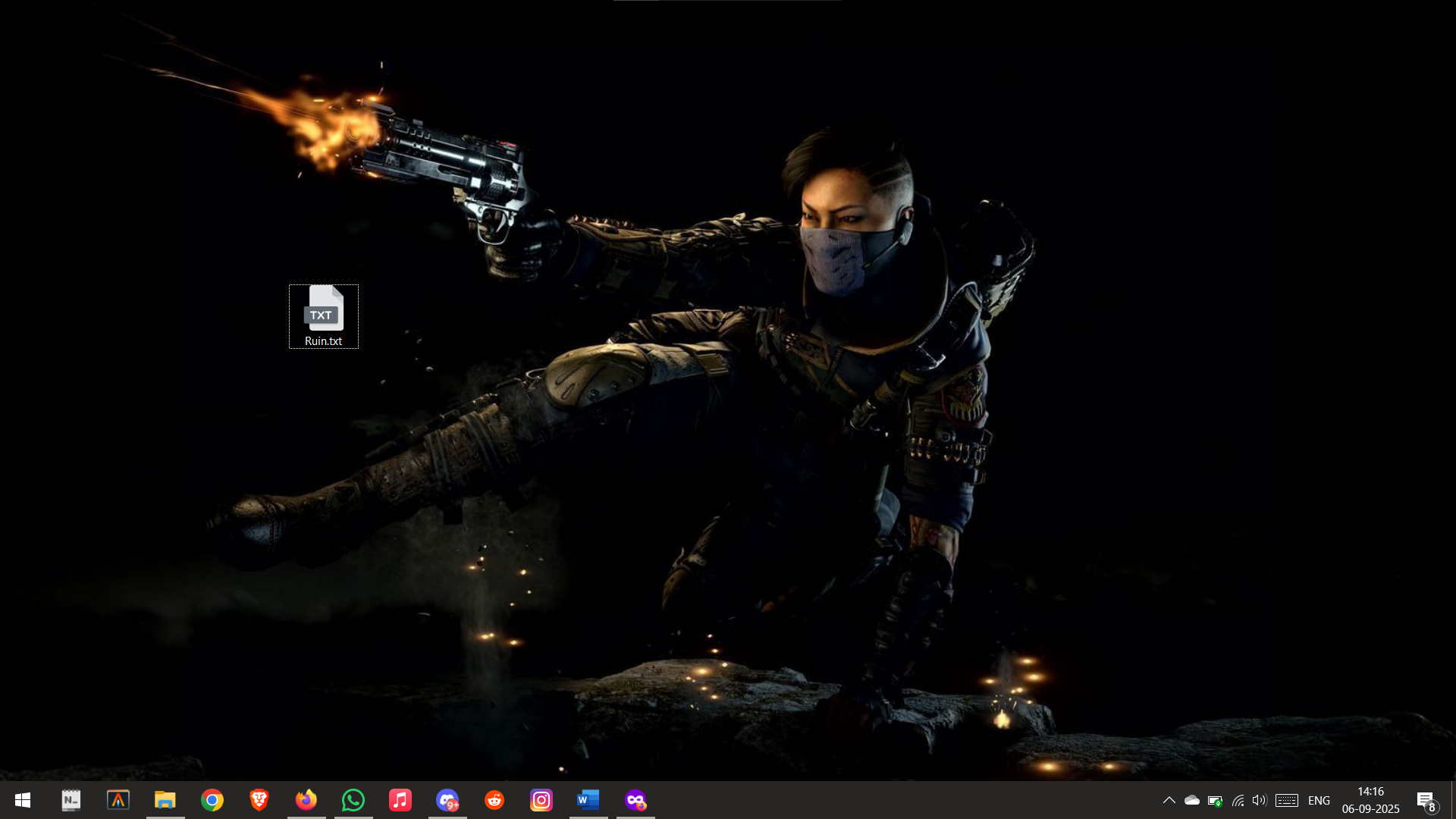
Bucket Created Successfully



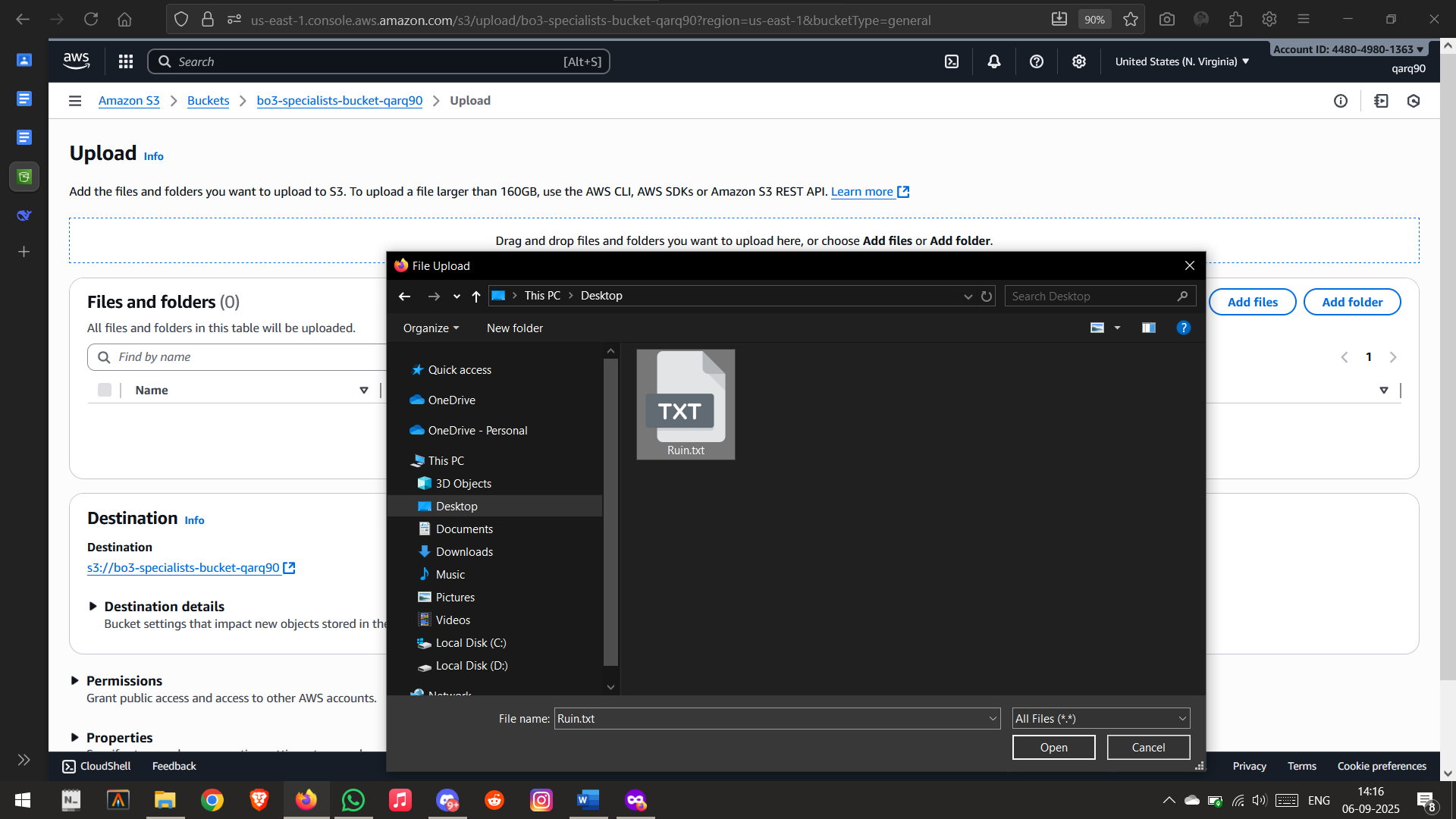
Creating the trigger



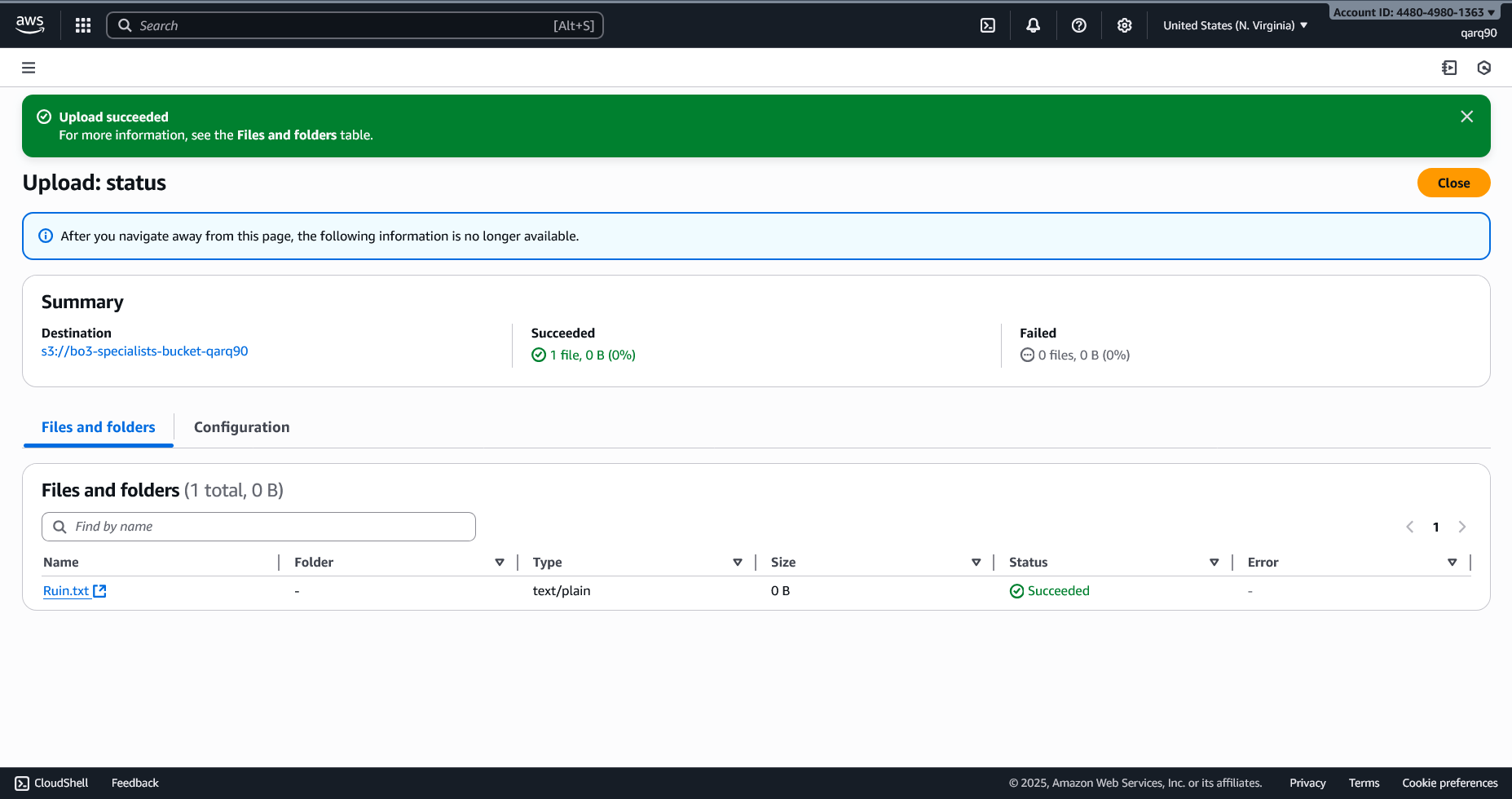
Adding the trigger to the bucket



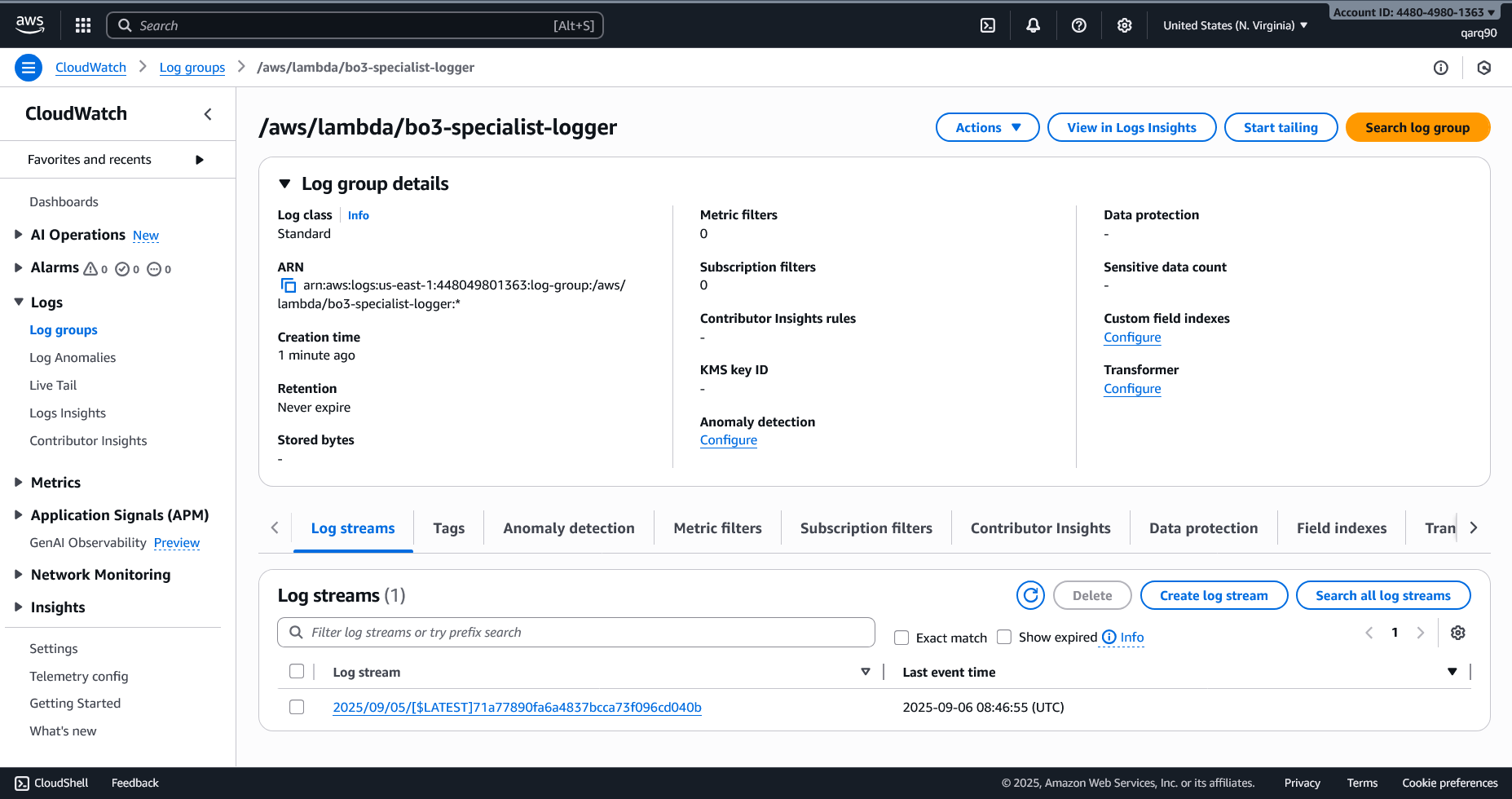
Created a file to trigger the trigger on the bucket



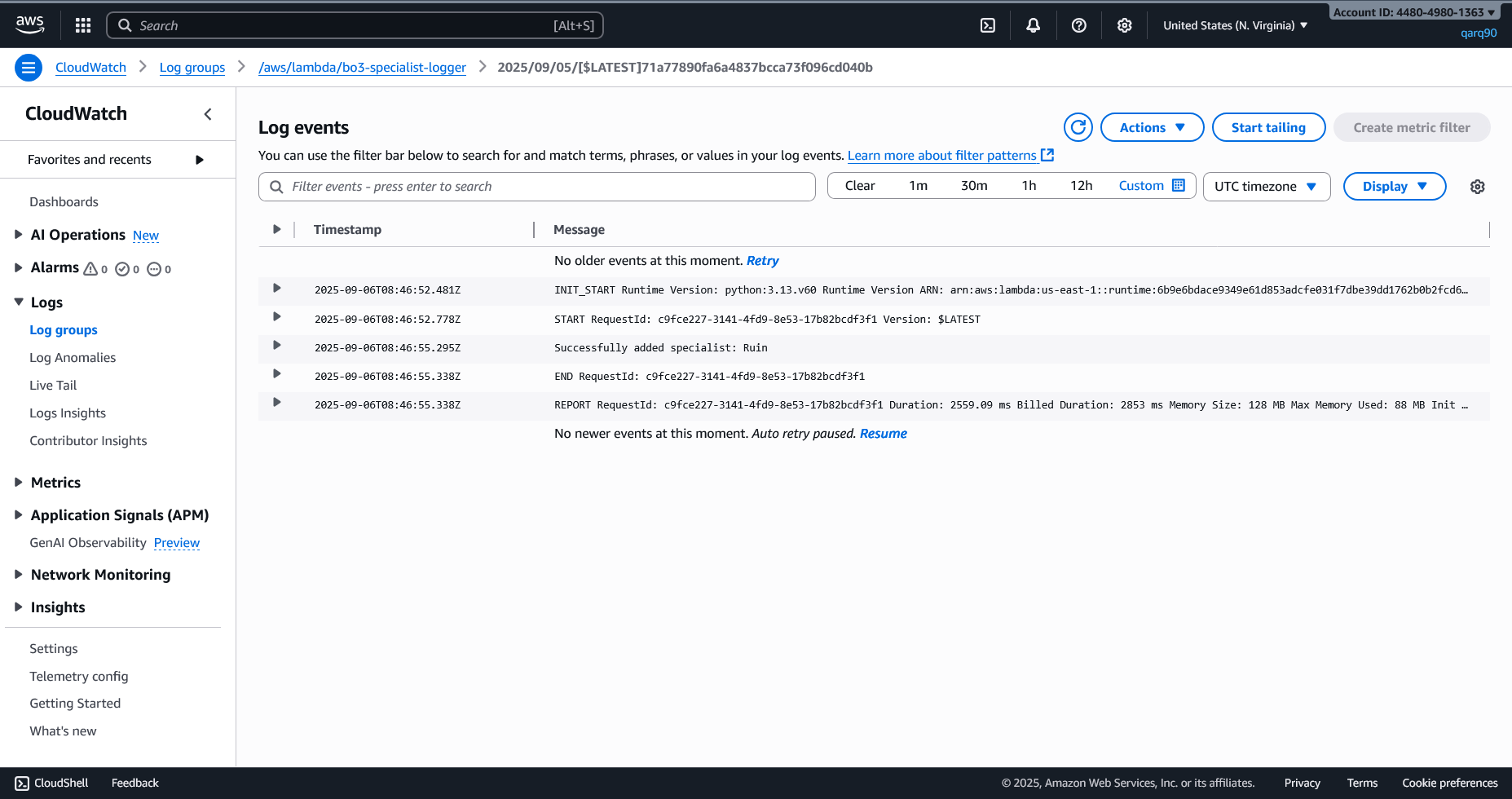
Uploading the file to the bucket



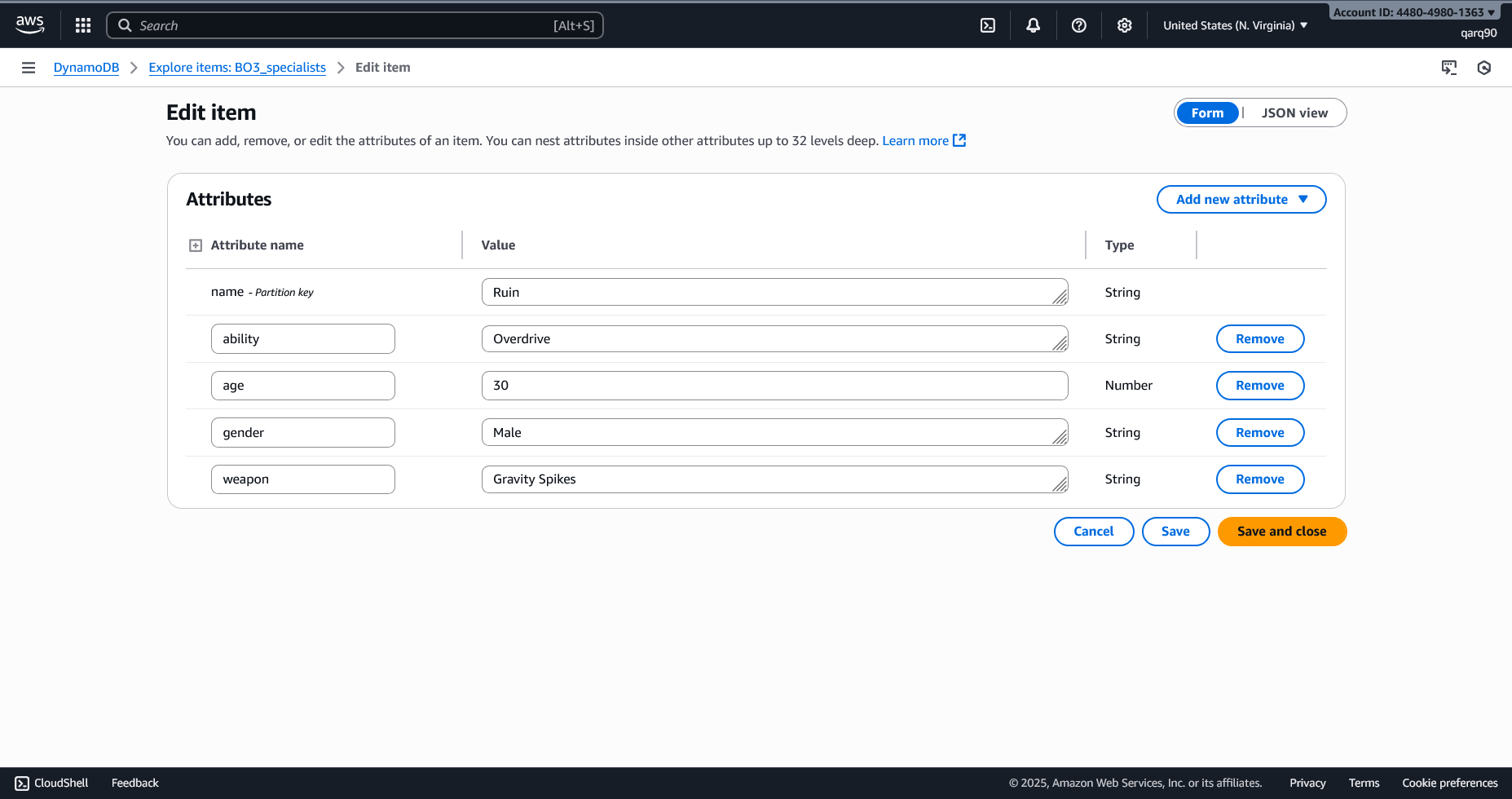
File uploaded successfully



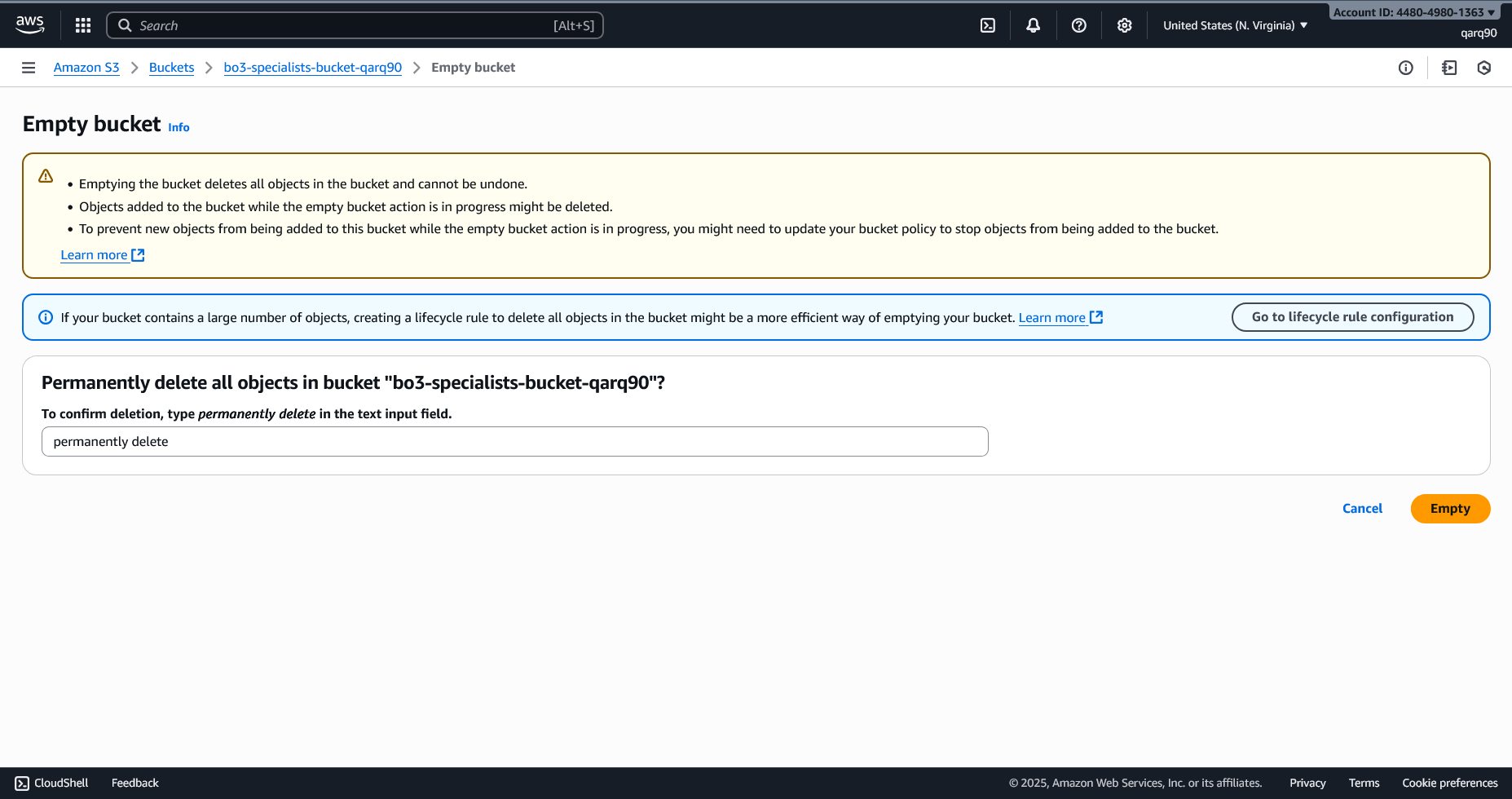
Checking the logs



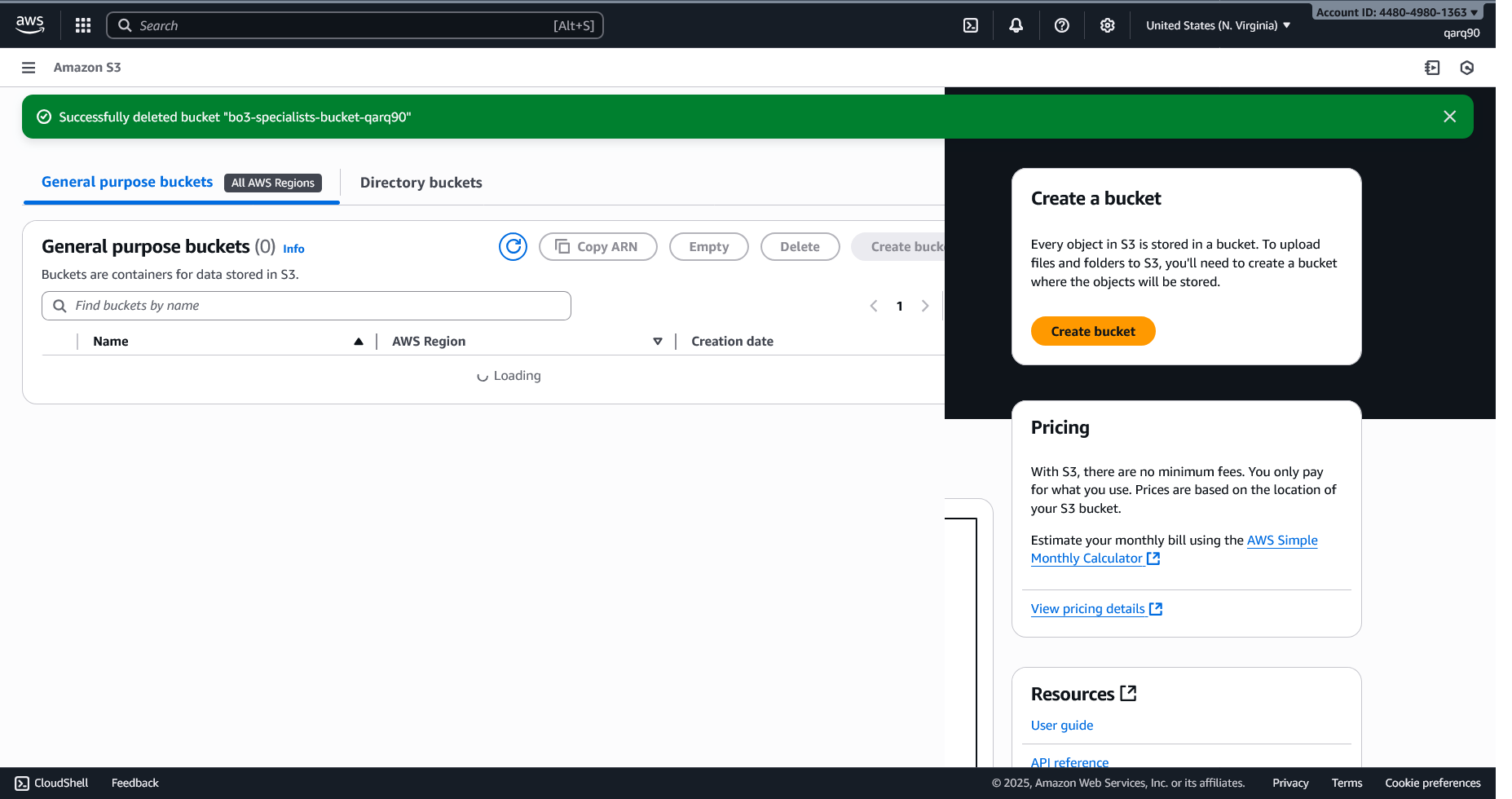
Log created successfully via the trigger



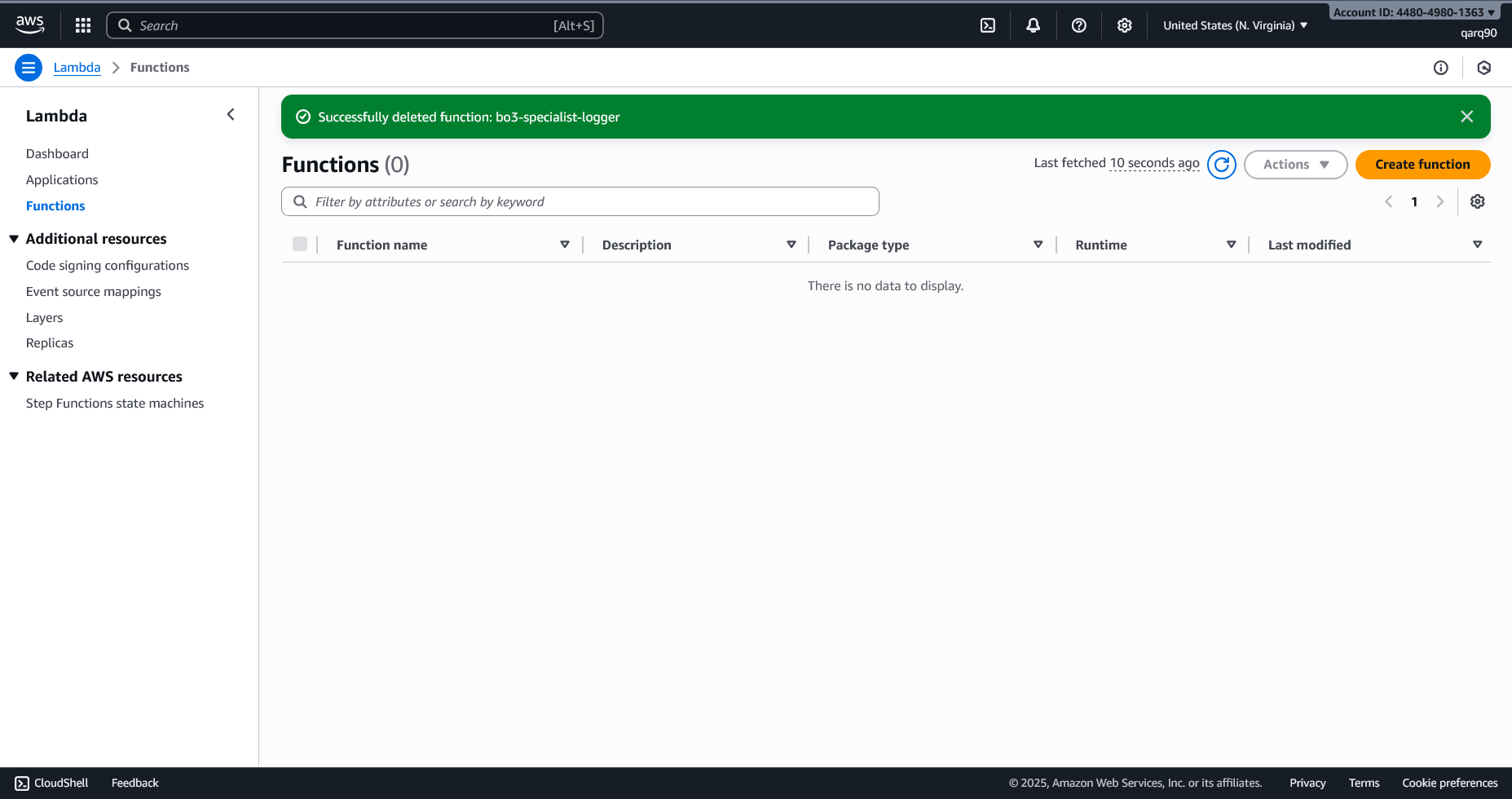
Verified the insertion in the Table



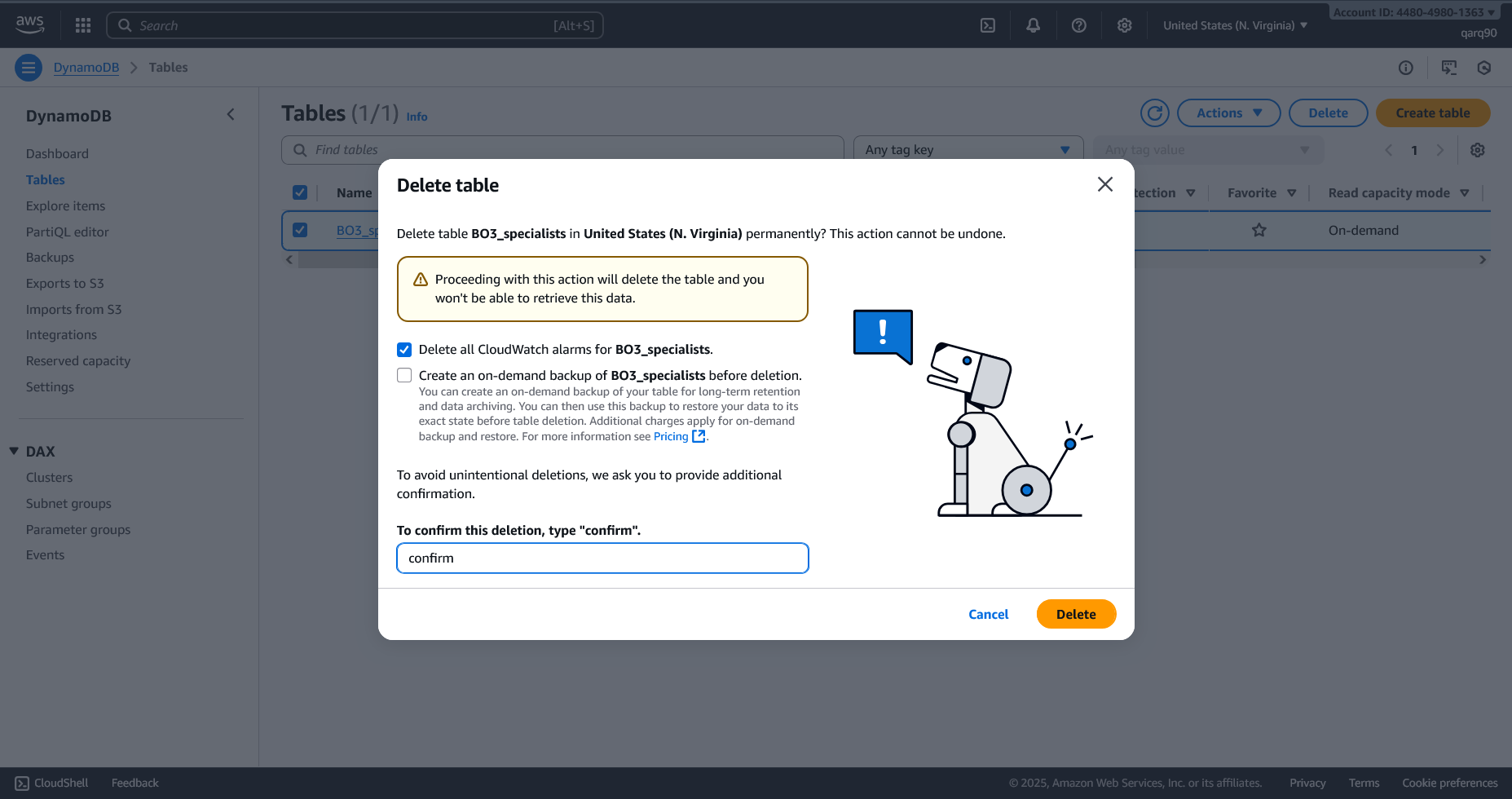
Emptying the bucket



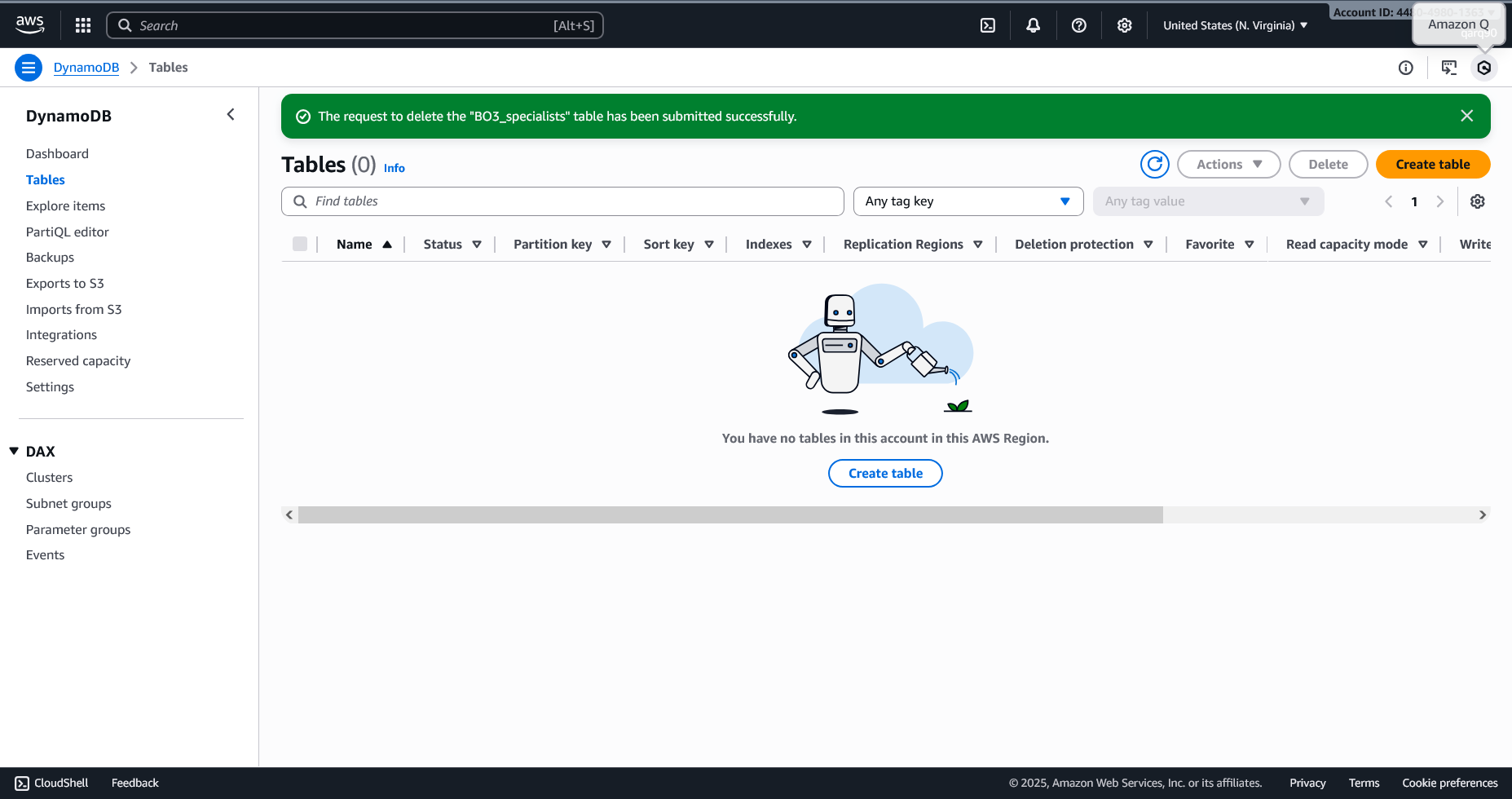
Deleting the bucket



Deleting the lambda function



Emptying the table



Deleted the table