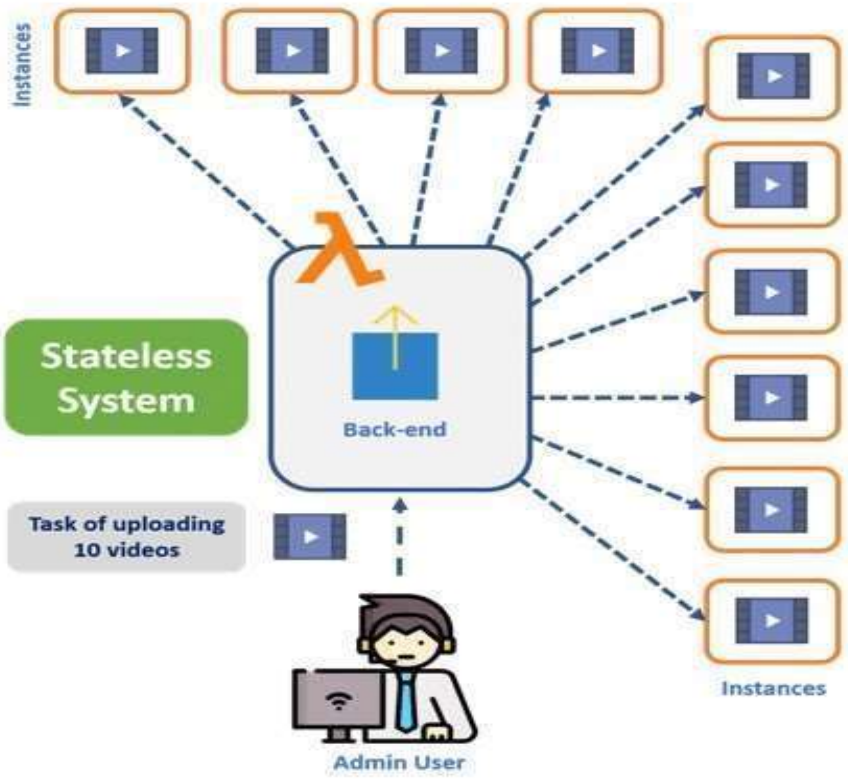
EXPERIMENT NO. 11 DATE –

Aim -

To understand AWS Lambda, its workflow, various functions and create your first Lambda functions using Python / Java / Nodejs.

Theory -

AWS Lambda is one of the computing services provided by AWS, which is event- driven and serverless. It is a stateless serverless system that helps us run our background tasks in the most efficient manner possible.



Here, being server less doesn't mean that servers are nowhere in the play. Instead, it means that we don't have to worry about the provisioning or management of our servers or instances; it just helps us focus on our main goal, i.e., coding. We just have to put our code in AWS Lambda, and we're good to go! Whatever resources are required for our code in response to our events, AWS Lambda automatically provides us. The best feature of it is that we just have to pay for every single request made during the time.

Events that Trigger AWS Lambda

Here, are Events which will be triggered when you use AWS Lambda.

- Insert, updating and deleting data Dynamo DB table
- To include push notifications in SNS
- To search for log history in CloudTrail
- Entry into an S3 object
- DynamoDB can trigger AWS Lambda whenever there is data added, modified, and deleted in the table.
- Helps you to schedule the event to carry out the task at regular time pattern. • Modifications to objects in S3 buckets
- Notifications sent from Amazon SNS.
- AWS Lambda can be used to process the CloudTrail logs
- API Gateway allows you to trigger AWS Lambda on GET/POST methods.

AWS Lambda Concepts

Function:
A function is a program or a script which runs in AWS Lambda. Lambda passes invocation events into your function, which processes an event and returns its response.

Runtimes:
Runtime allows functions in various languages which runs on the same base execution environment. This helps you to configure your function in runtime. It also matches your selected programming language.

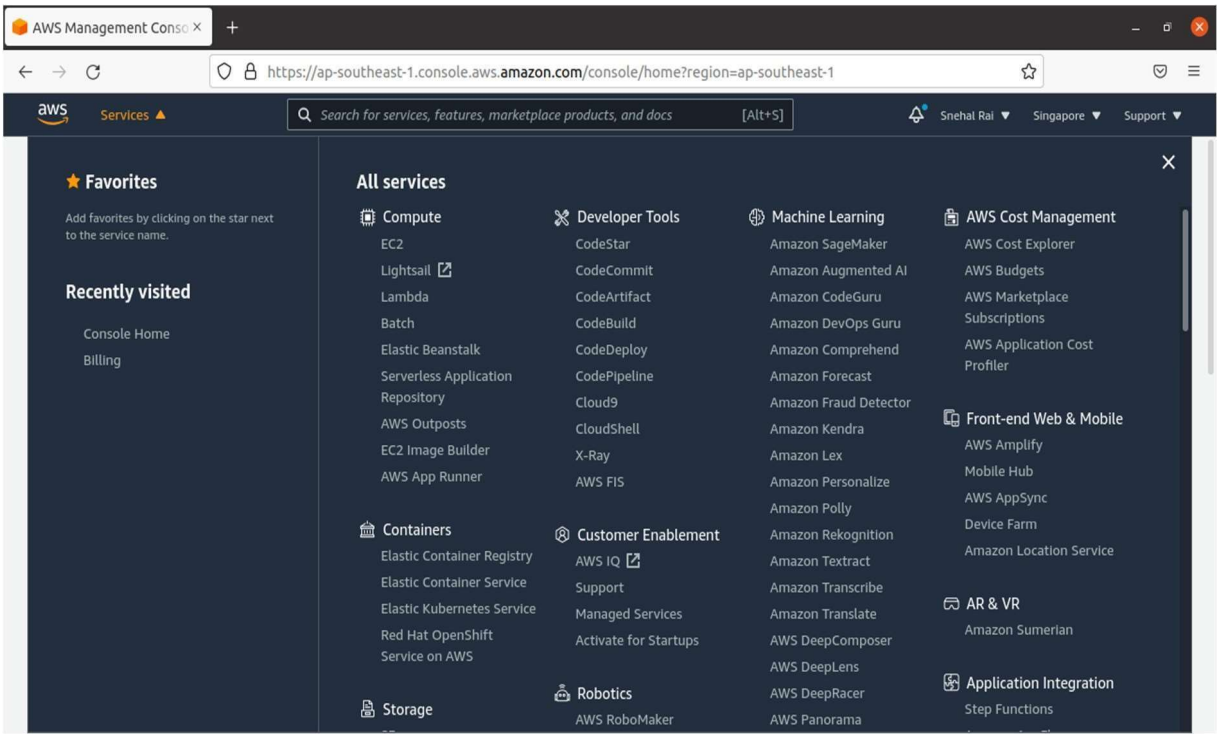
Event source:
An event source is an AWS service, such as Amazon SNS, or a custom service. This triggers function helps you to execute its logic.

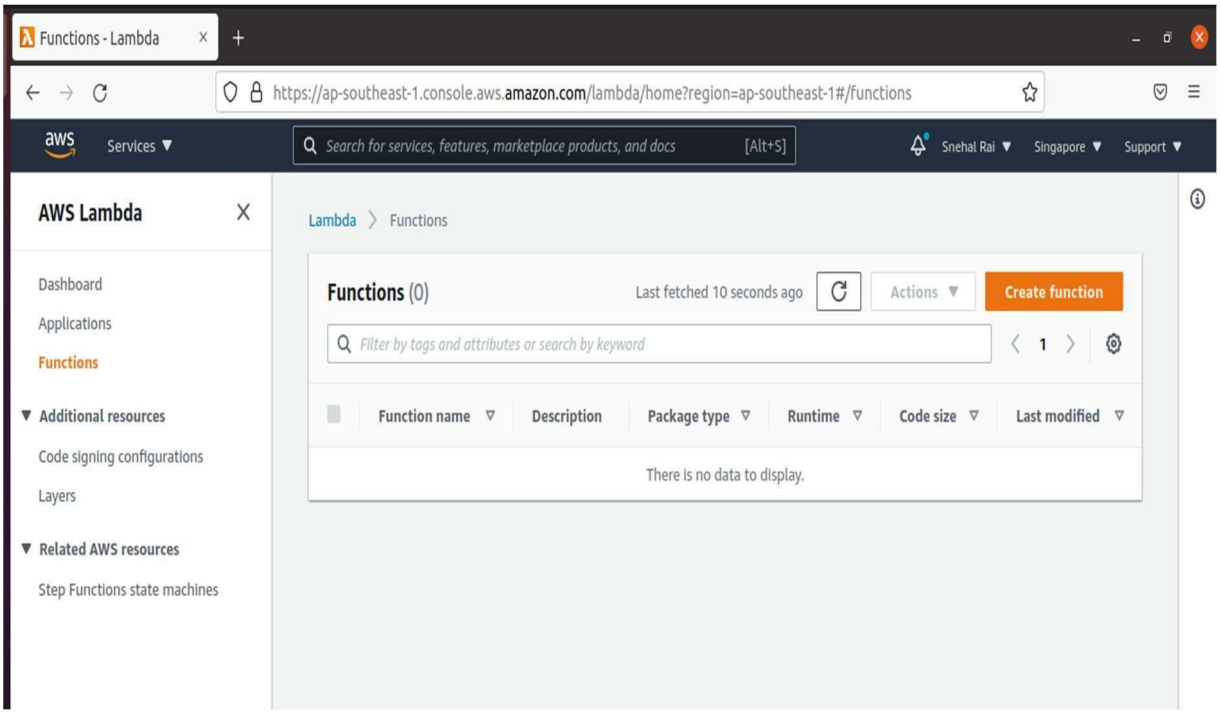
Lambda Layers:
Lambda layers are an important distribution mechanism for libraries, custom runtimes, and other important function dependencies. This AWS component also helps you to manage your development function code separately from the unchanging code and resources that it uses.

Log streams:
Log stream allows you to annotate your function code with custom logging statements which helps you to analyze the execution flow and performance of your AWS Lambda functions.

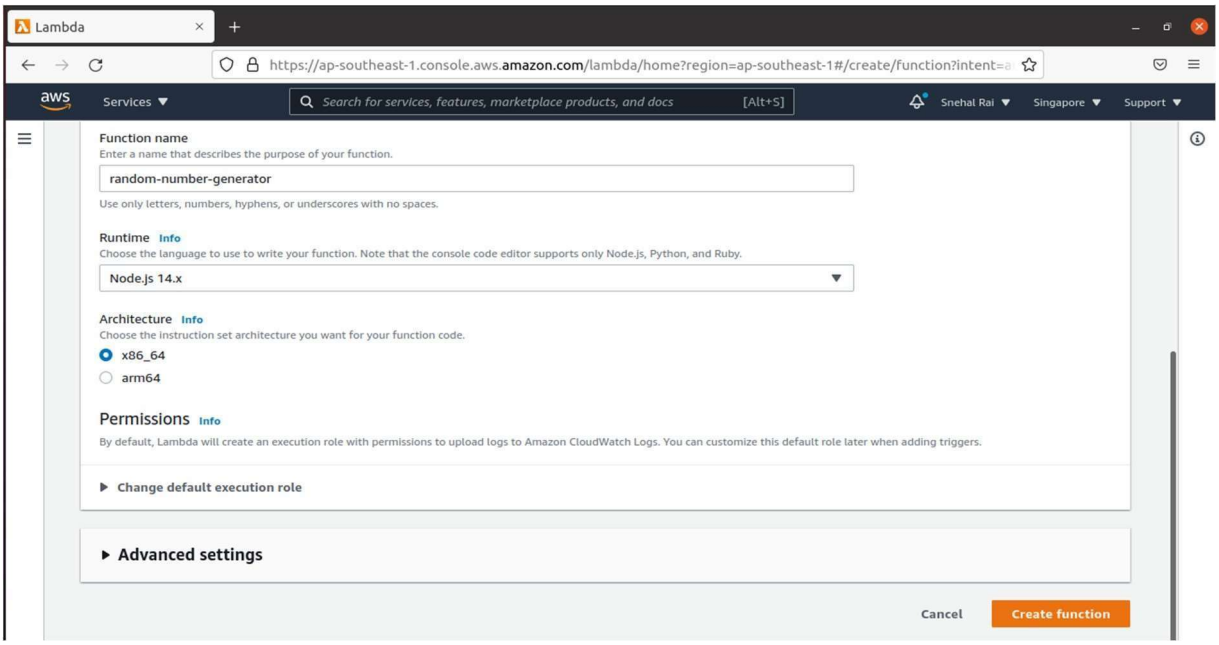
Steps:

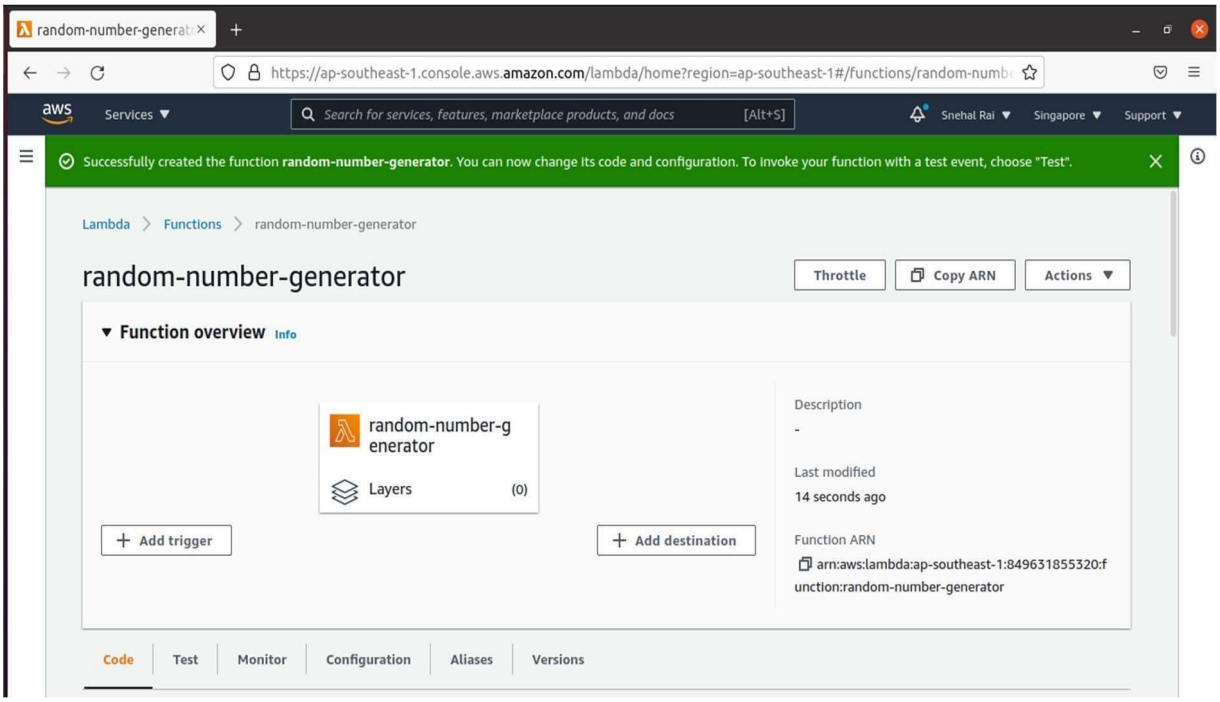
Step 1 - Open AWS Lambda URL in AWS





Step 2 - Create an account Next, Create an account or sign in with your existing account

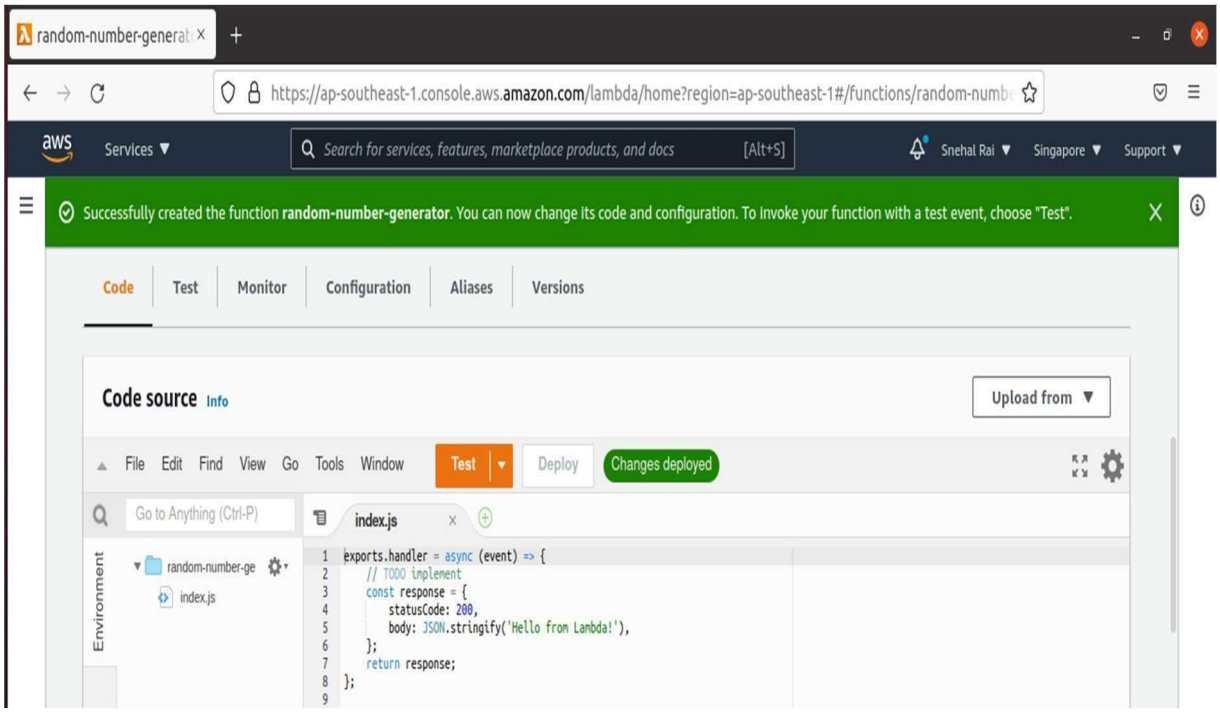


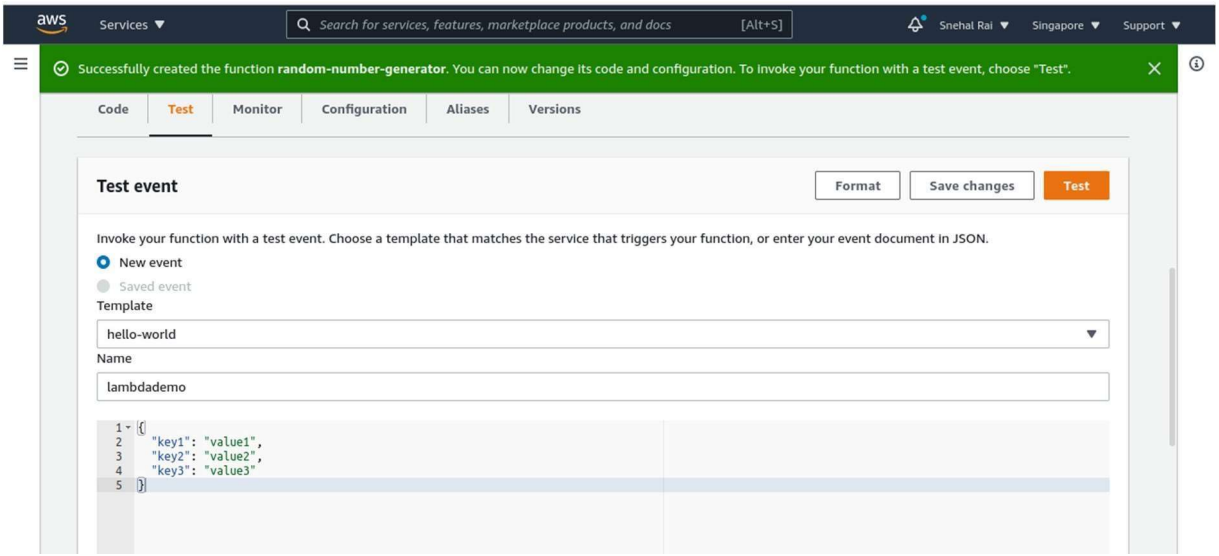


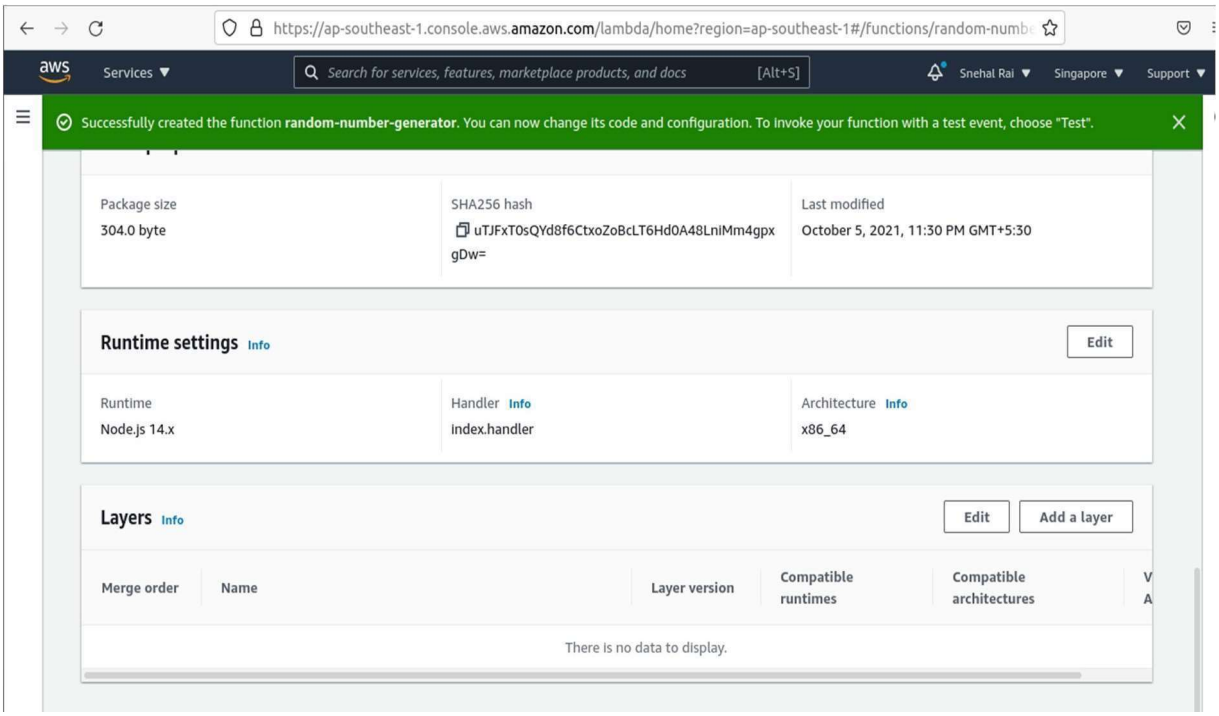
Step 3-

Edit the code & Click Run, In the next Lambda page,

1. Edit the code
2. Click Run

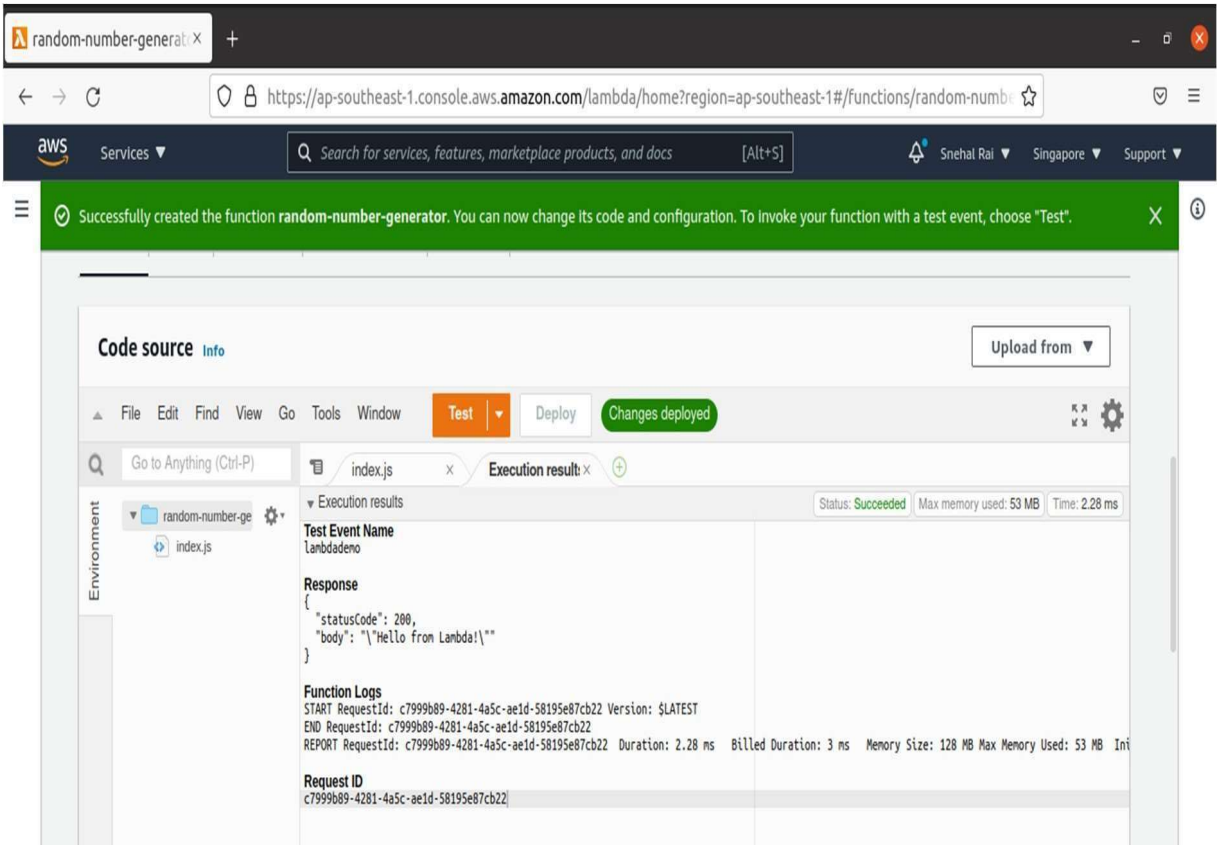


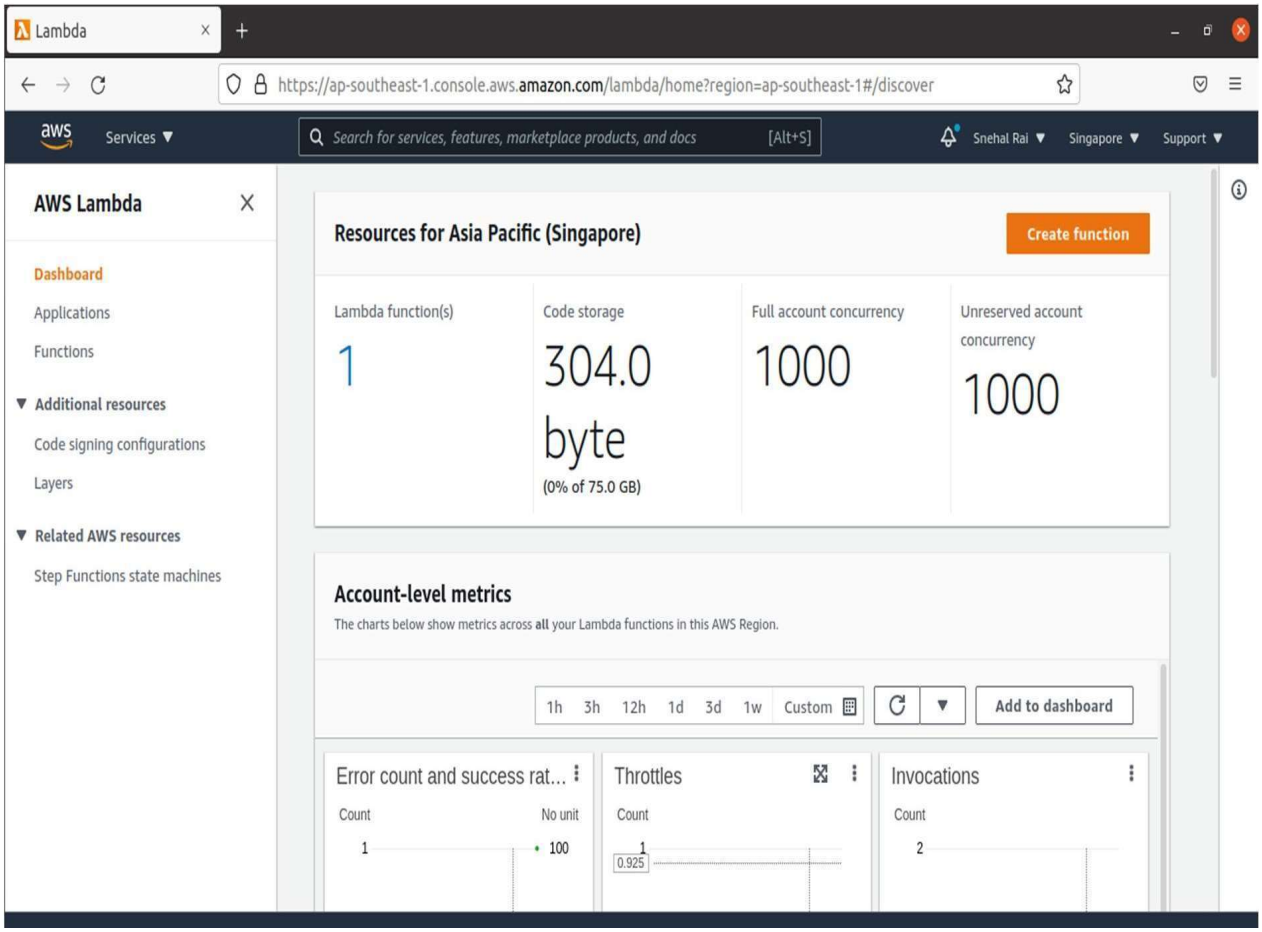




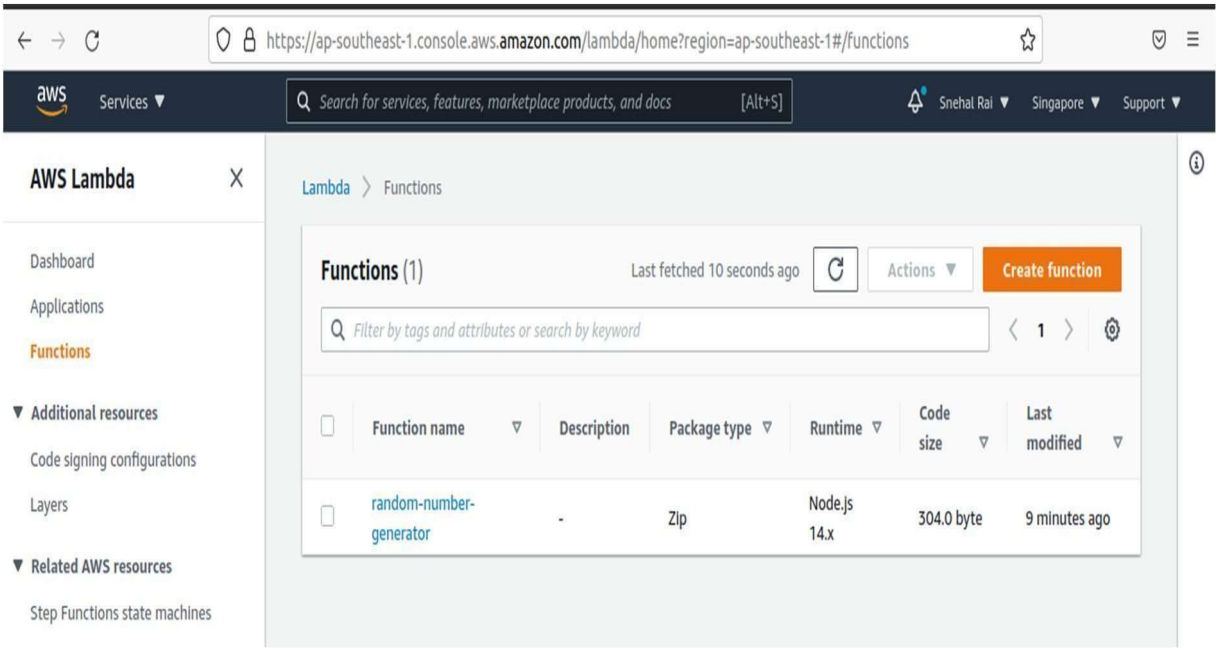
Step 4 - Check output

You will see output





Step 5- Delete Function.



Conclusion -

Lambda enables the creation of new micro services to access the data stream by decoupling the product engineering efforts from the platform analytics pipeline, eliminating the need to be bundled with the main analytics applications.

From this experiment, I got a detailed understanding of AWS Lambda, its workflow, various functions, how to create Lambda functions, its need, and various use cases.