Vivekanand Education Society's

Institute of Technology

An Autonomous Institute Affiliated to University of Mumbai,, Approved by AICTE & Recognized by Govt. of Maharashtra

Hashu Advani Memorial Complex, Collector Colony, Chembur East, Mumbai - 400074.

Department of Information Technology

A.Y. 2024-25

Advance DevOps Lab
Experiment 11

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

Roll No.	53
Name	Aryan Deepak Saraf
Class	D15B
Subject	Advance DevOps Lab
LO Mapped	LO1: To understand the fundamentals of Cloud Computing and be fully proficient with Cloud based DevOps solution deployment options to meet your business requirements. LO6: To engineer a composition of nano services using AWS Lambda and Step Functions with the Serverless Framework.
Grade:	

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

THEORY :

AWS Lambda is a serverless computing service by AWS that allows you to run code without provisioning or managing servers. You create functions in supported languages like Python, Java, and Node.js, and these functions are executed in response to specific events such as API calls, file uploads to S3, or data changes in DynamoDB.

Key Features

- Automatic Scaling: Lambda automatically scales the infrastructure to handle incoming requests, reducing operational complexity.
- Cost-Efficiency: You only pay for the compute time you consume, with no upfront costs or server management fees.
- Security: Lambda integrates with AWS Identity and Access Management (IAM) to define roles and policies, ensuring secure execution.
- Fault Tolerance: AWS Lambda is designed to provide high availability and fault tolerance, handling server failures and maintaining continuous operation.

Execution Model

Lambda functions run in stateless containers fully managed by AWS. When an event triggers a function, AWS initiates a container to execute the function. If subsequent requests come in, additional containers are spun up to handle them. AWS may keep containers warm for a short period to reduce cold start latency.

Stateless Functions

Due to the stateless nature of Lambda, each function invocation is independent, running in a fresh environment. Code outside the main handler function runs once per container lifecycle, while the handler itself runs on every invocation.

Common Use Cases

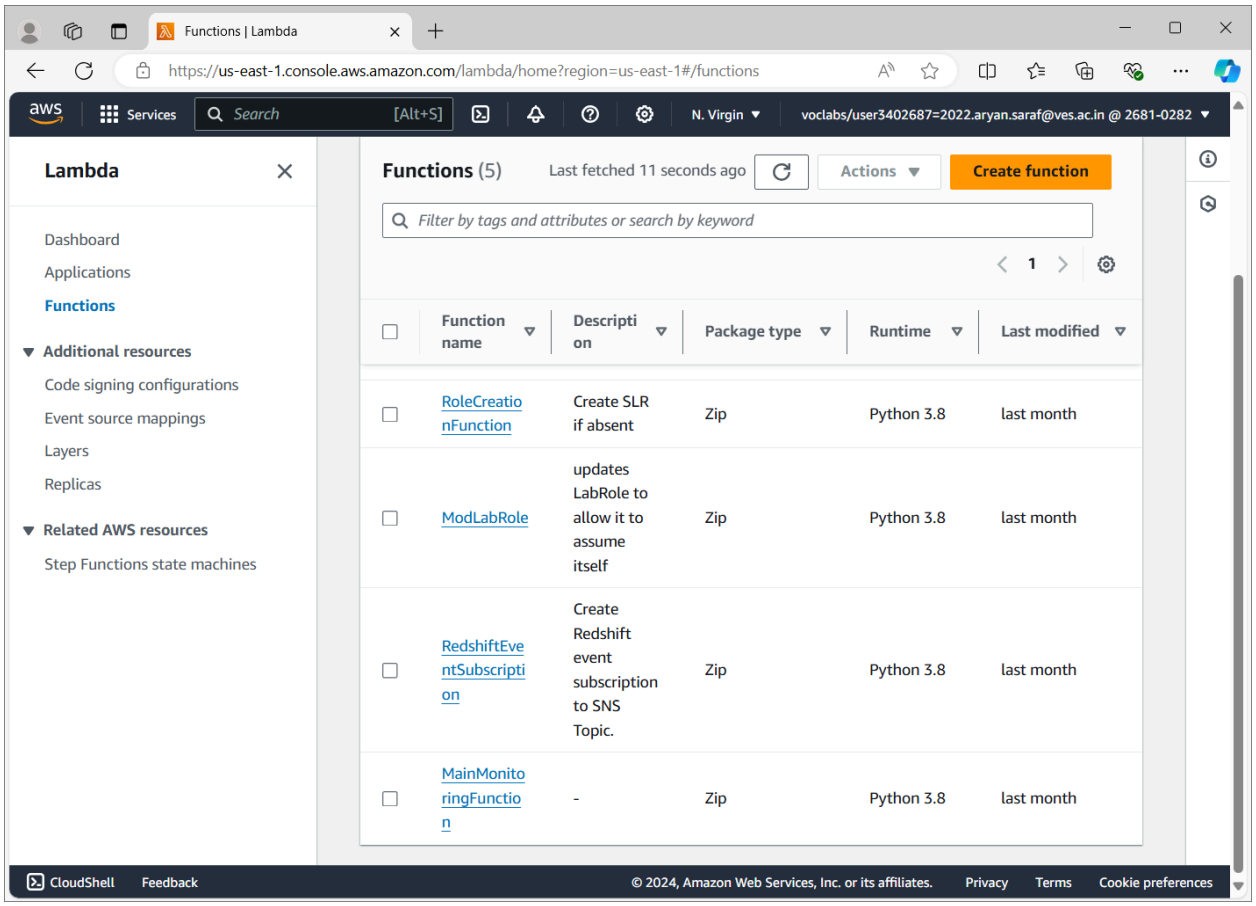
- Scalable APIs: Lambda is ideal for building APIs that need to scale according to demand. Each API request can be routed to a specific Lambda function, and the service automatically adjusts to handle varying workloads.
- Event-Driven Data Processing: Lambda excels in scenarios like real-time data processing, where functions are triggered by events from sources like S3 or DynamoDB, making it suitable for tasks like data transformation, analytics, and notifications.

Packaging and Deployment

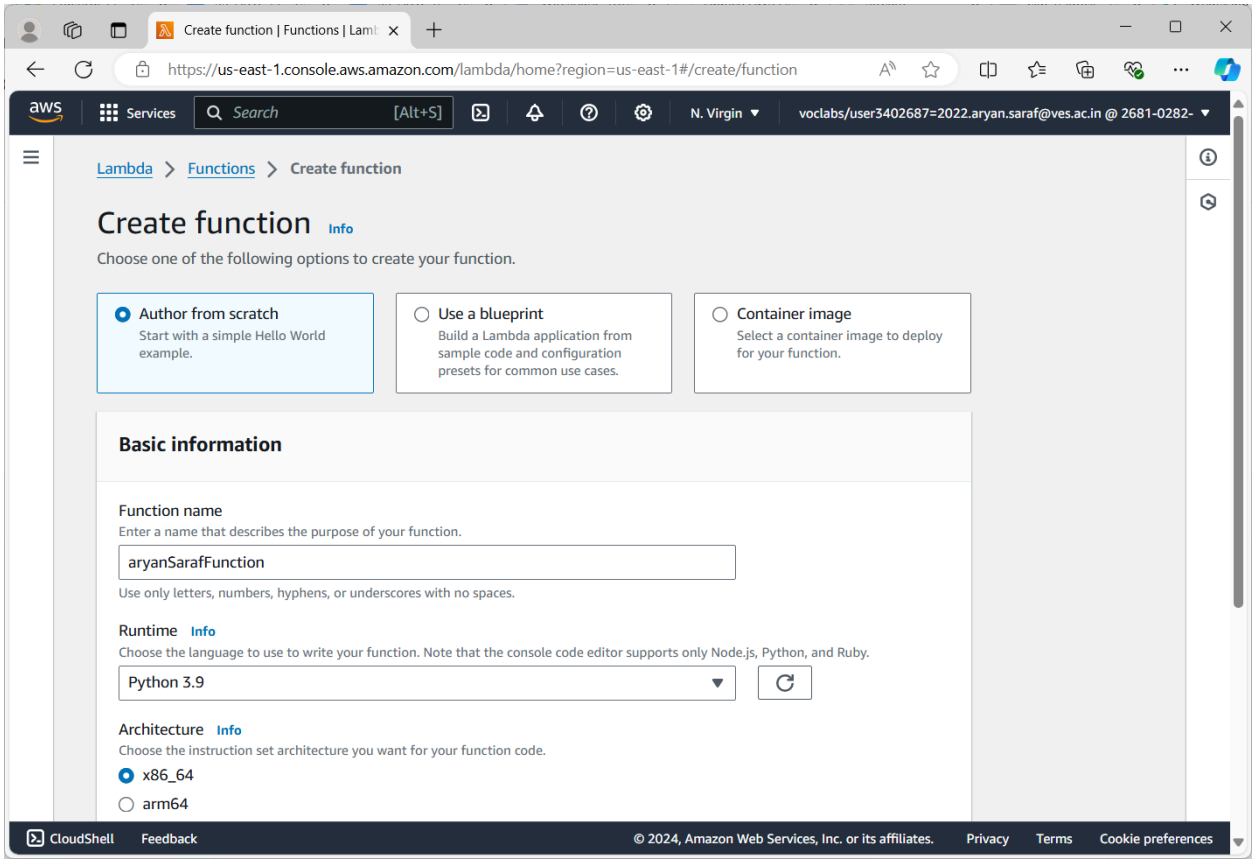
Lambda functions, along with their dependencies, are packaged and uploaded to AWS, often using an S3 bucket. AWS Lambda then uses this package to execute the function when an event occurs. Tools like the Serverless Stack Framework (SST) can simplify this process.

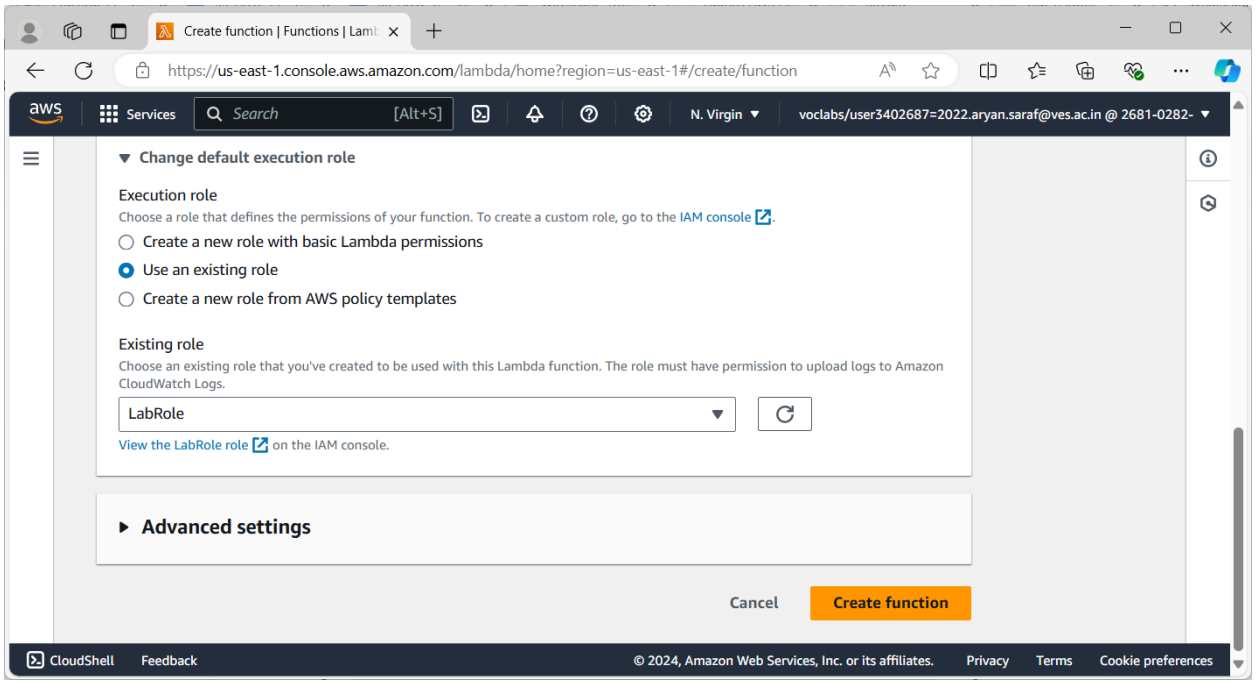
Steps to create an AWS Lambda function

Open up the Lambda Console and click on the Create button.

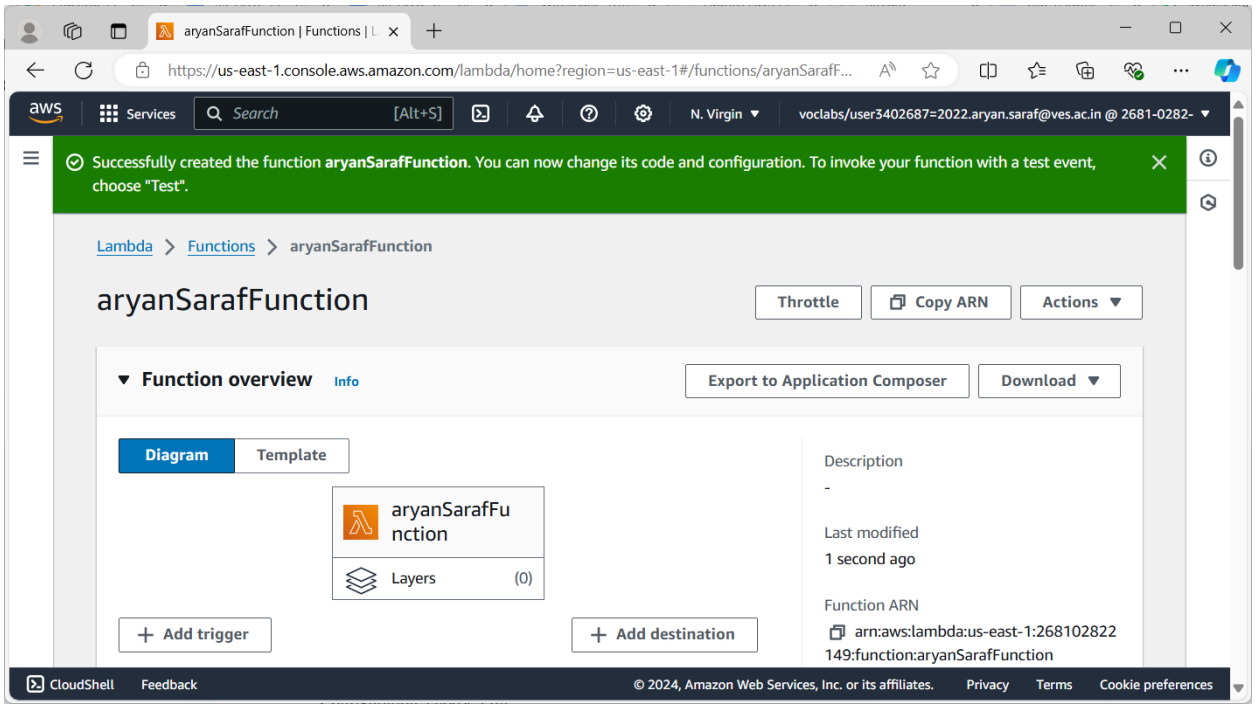


Choose to create a function from scratch or use a blueprint, i.e templates defined by AWS for you with all configuration presets required for the most common use cases.

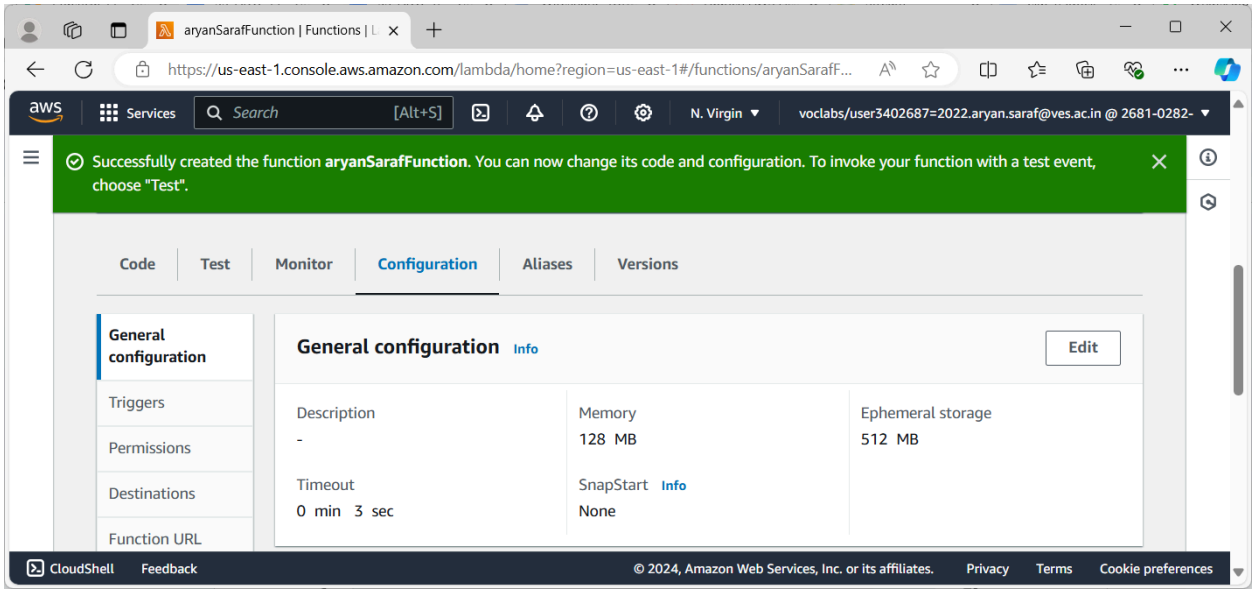


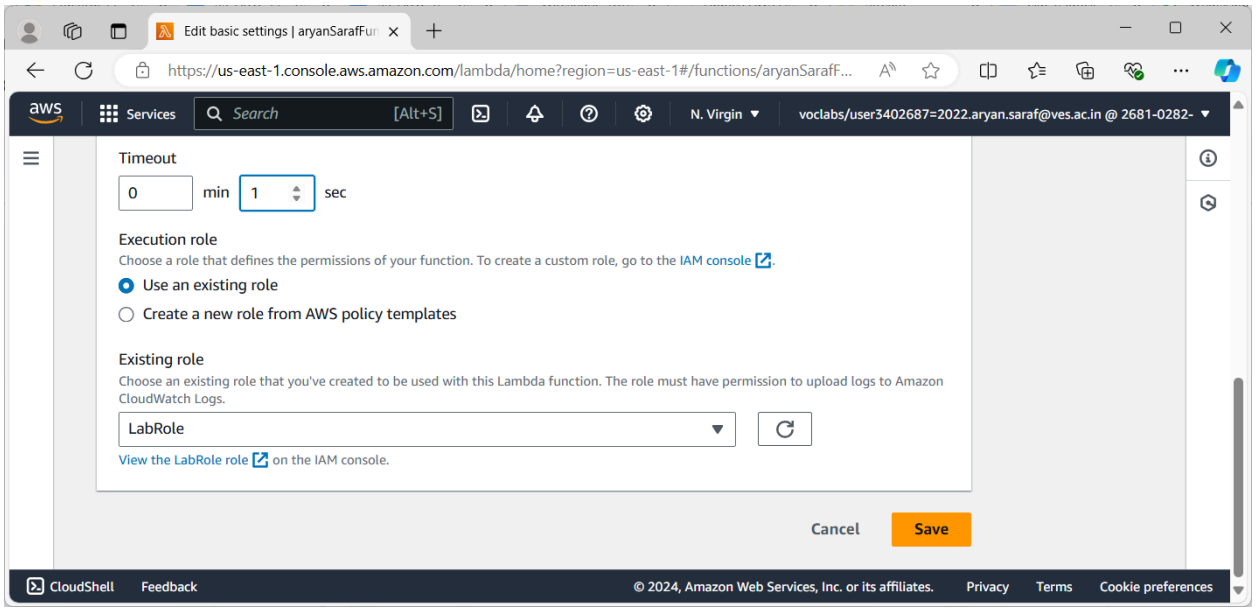


This process will take a while to finish and after that, you'll get a message that your function was successfully created.

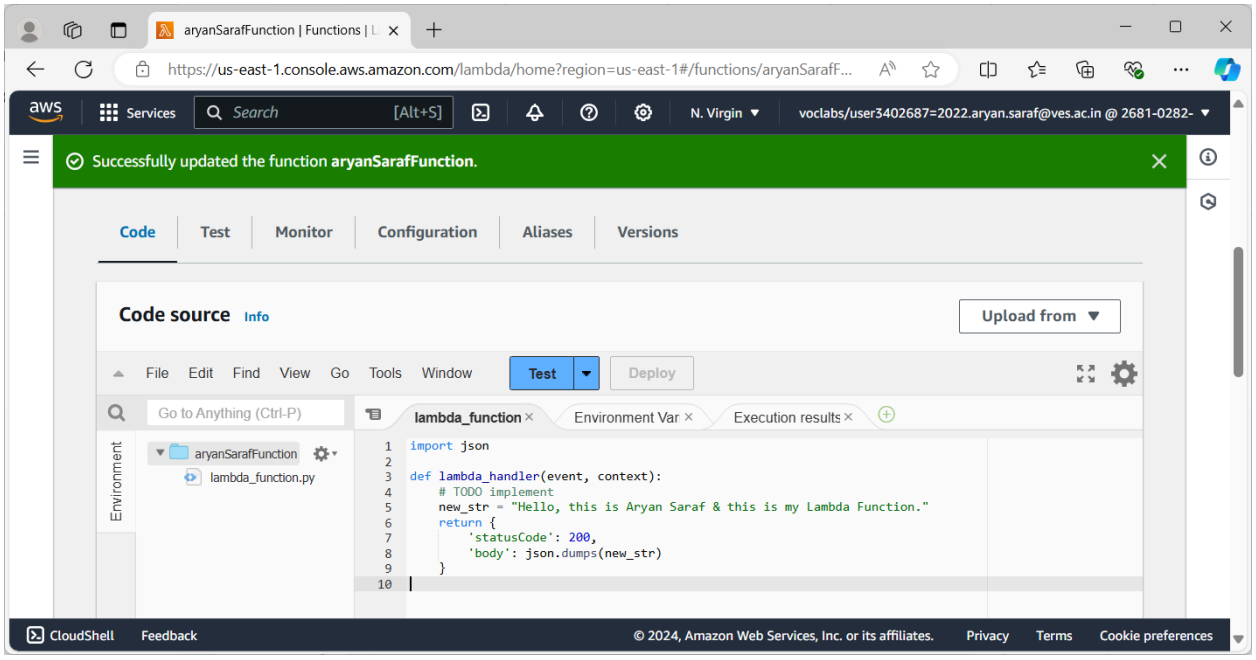


To change the configuration, open up the Configuration tab and under General Configuration, choose Edit.

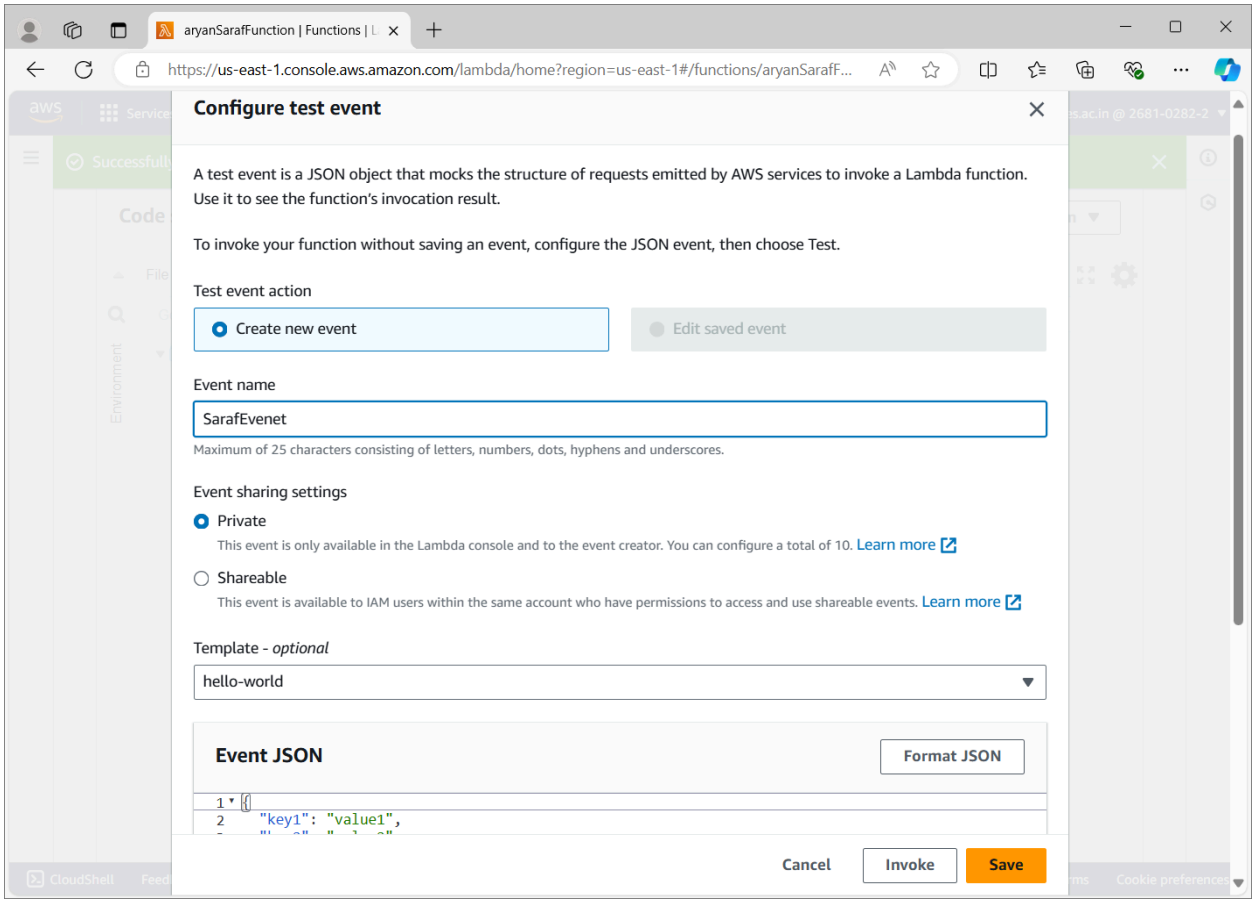




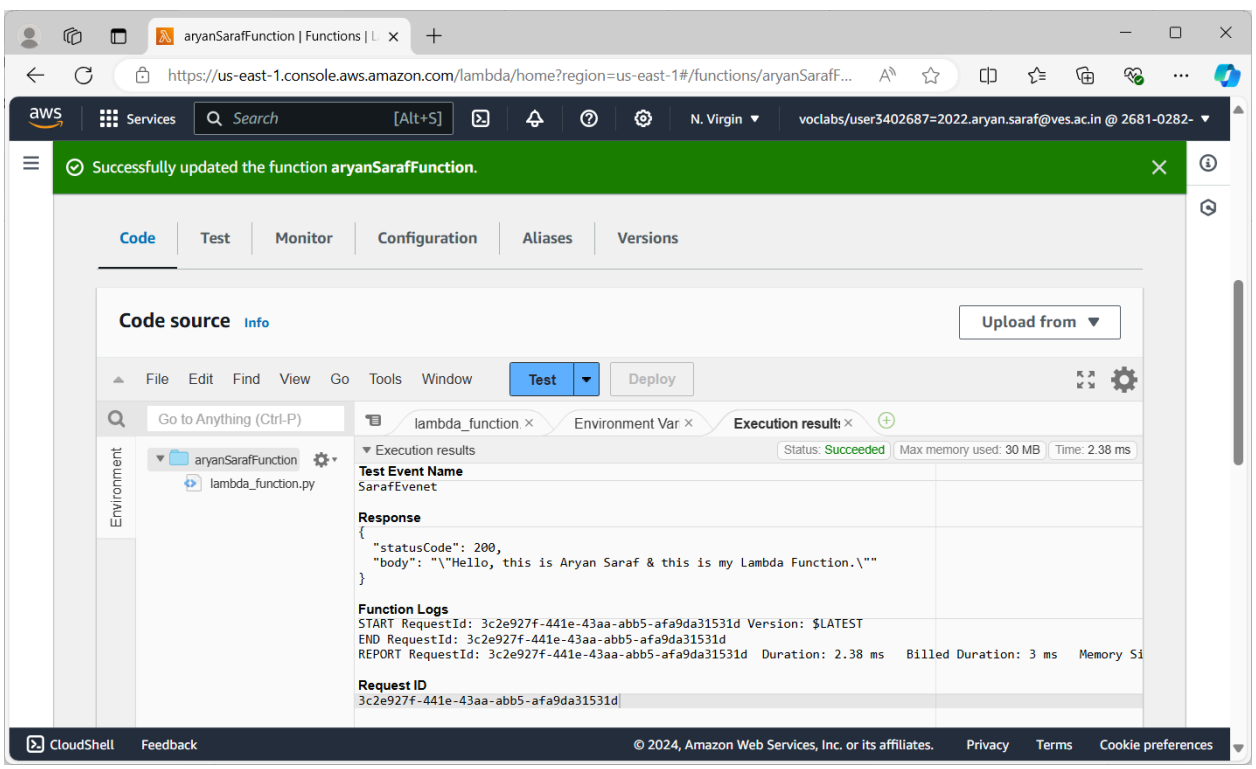
Press Ctrl + S to save the file and click Deploy to deploy the changes.



Click on Test and you can change the configuration, like so.



Now click on Test and you should be able to see the results.



CONCLUSION :

AWS Lambda simplifies the process of running code in the cloud by handling server management, scaling, and security for you. Its flexibility and cost-efficiency make it an ideal choice for a wide range of applications, from scalable APIs to real-time data processing. By leveraging AWS Lambda, you can focus on writing code while AWS takes care of the rest.