NAME: Bhushan Malpani **DIV:D15C** **ROLL NO:33**

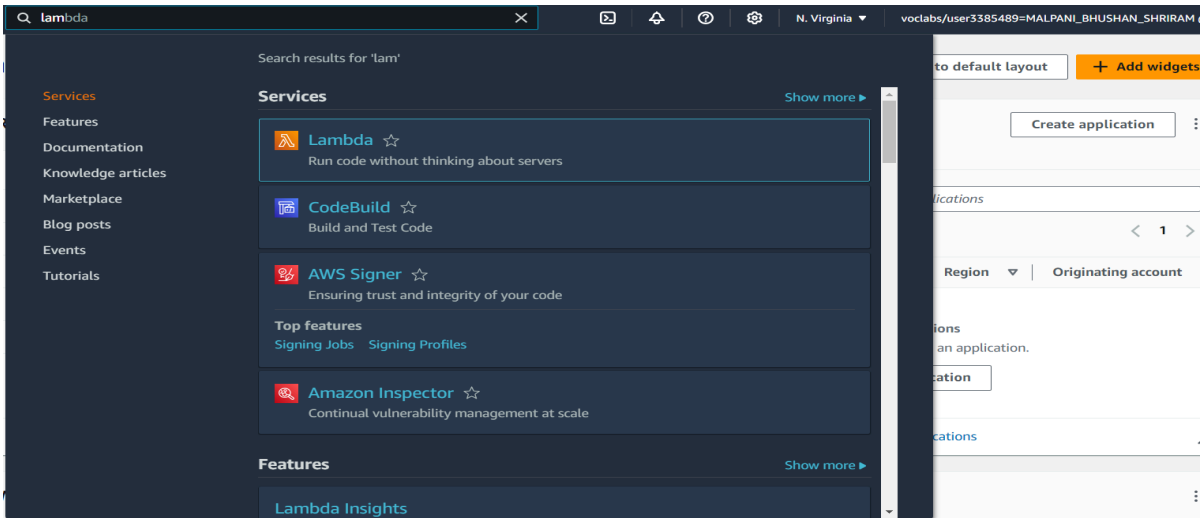
Aim: To understand AWS Lambda, its workflow, various functions and create your first Lambda functions using Python / Java / Nodejs.

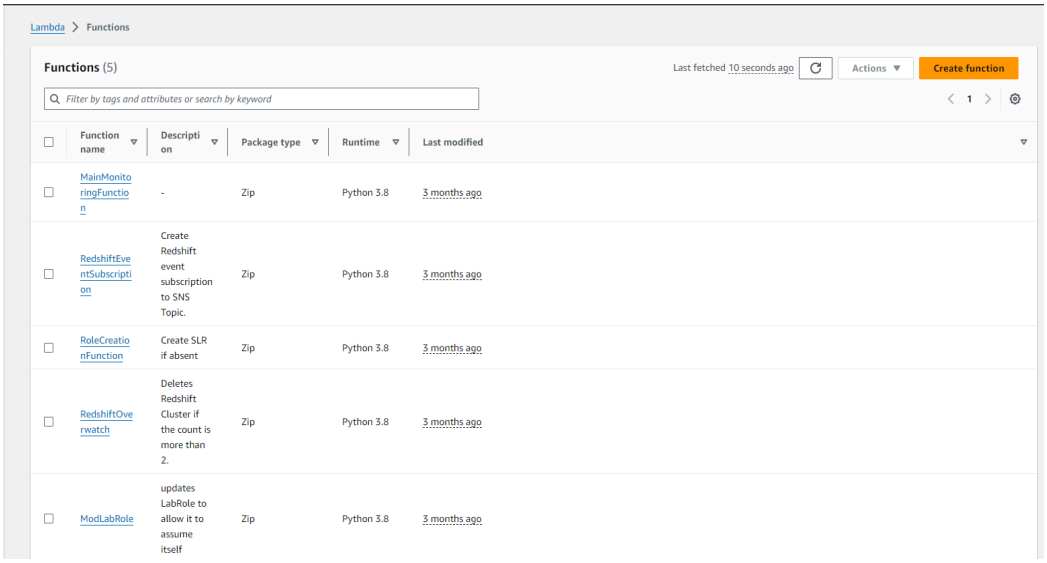
Prerequisites:

1) AWS account (academy recommended)

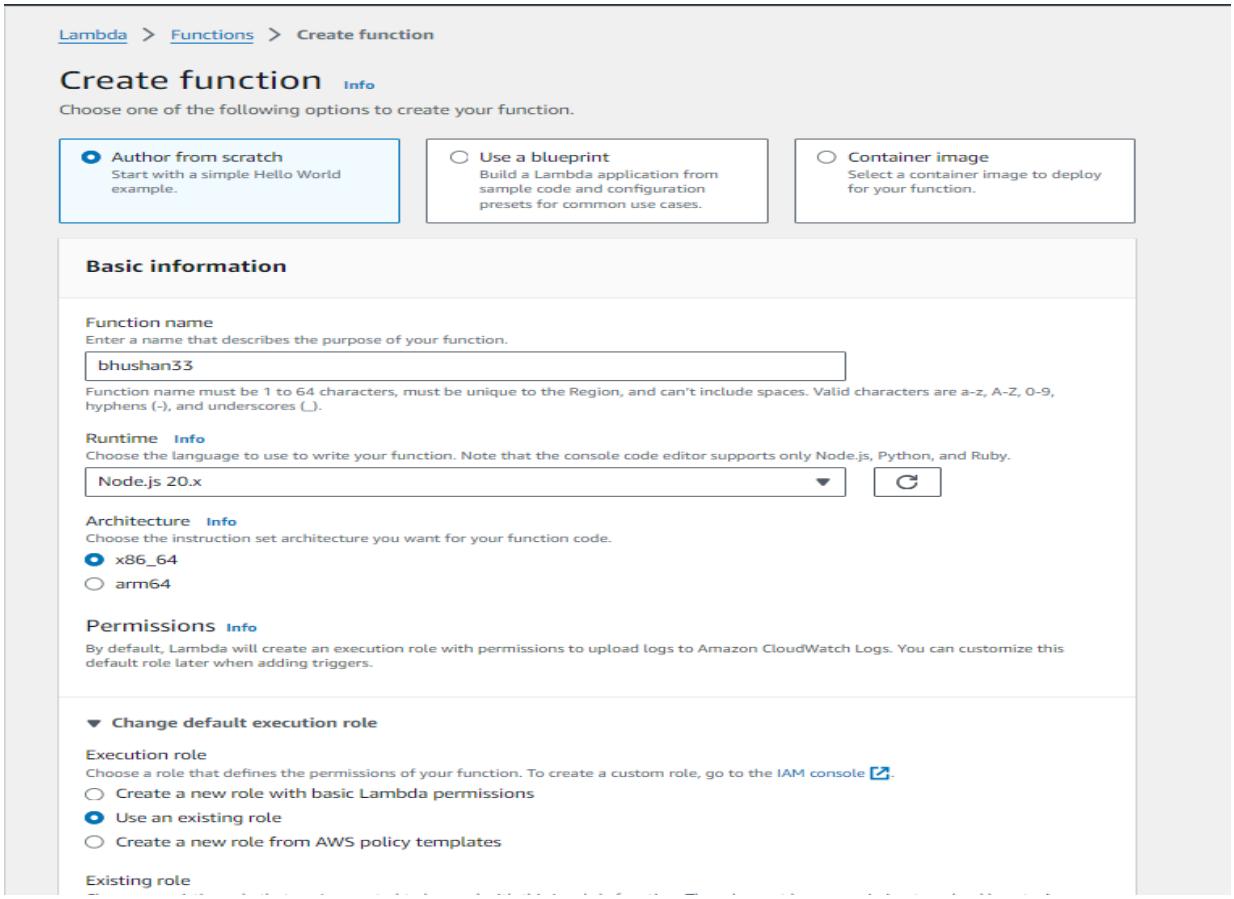
Step 1: Set up AWS Lambda Function

1) Search for Lambda in the services tab. Click on it once found.

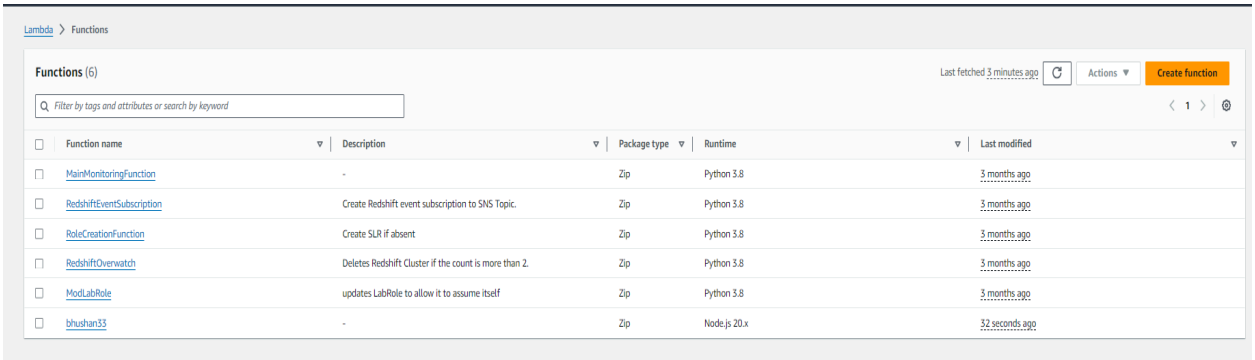




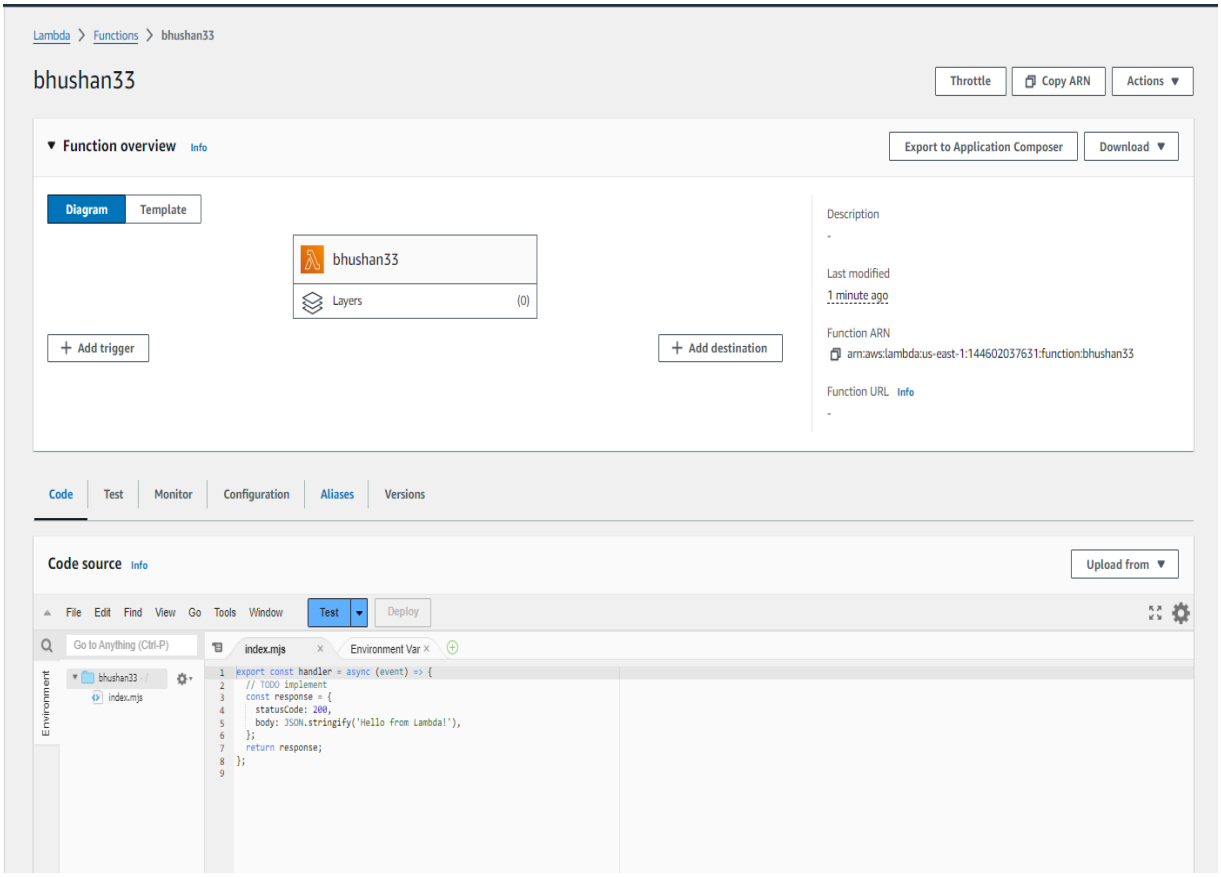
2) Click on create functions.
3) Give a name to your Lambda function. Select the runtime as Node.js 20.x (You can also use python). Select the architecture as x86_64. Set the default execution role as LabRole if you are doing this on your academy account. (Use an existing role → LabRole)



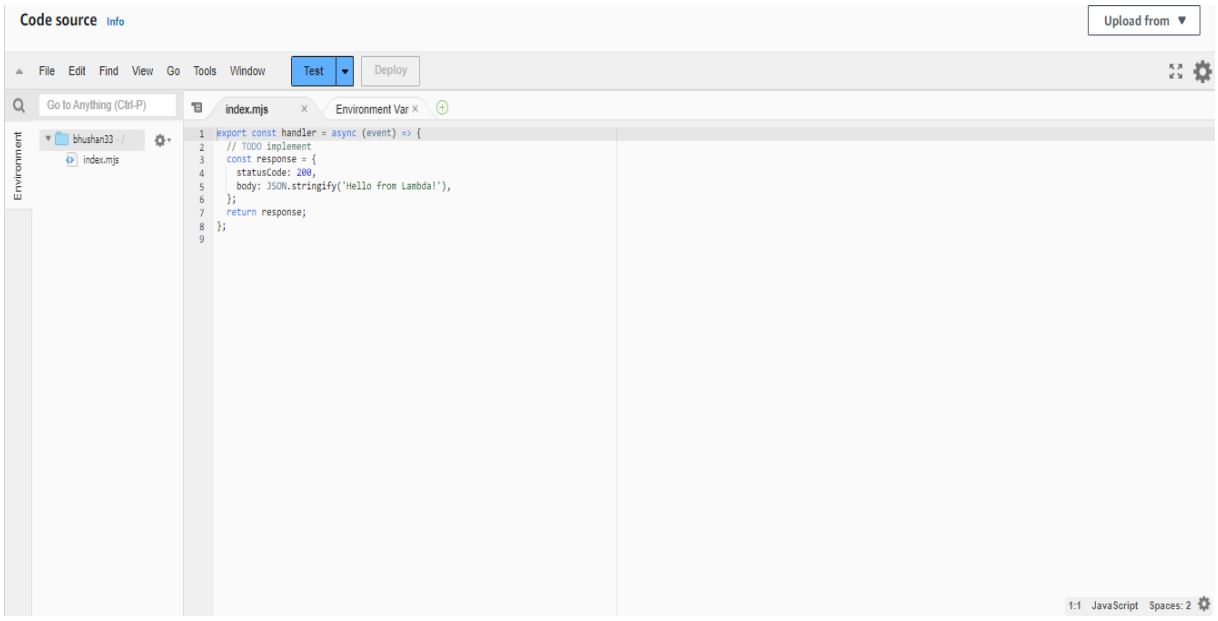
4) Once the function is created, click on the name of the function (myLambda27 in my case).



5) This is the dashboard of our lambda function.

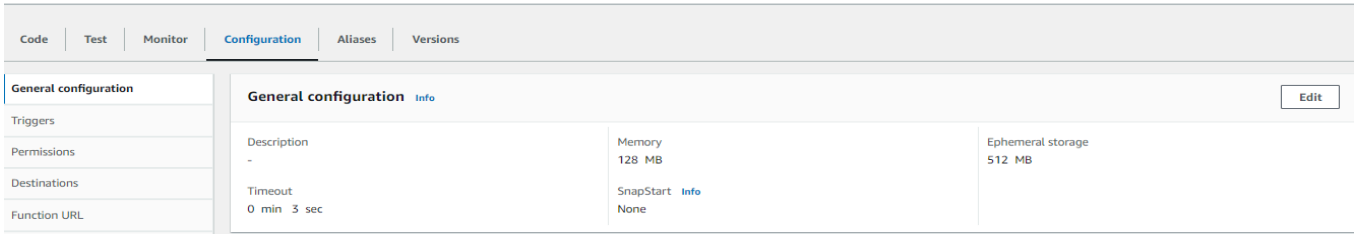


6) This function has the following default code, which is used to print “Hello form Lambda!”.

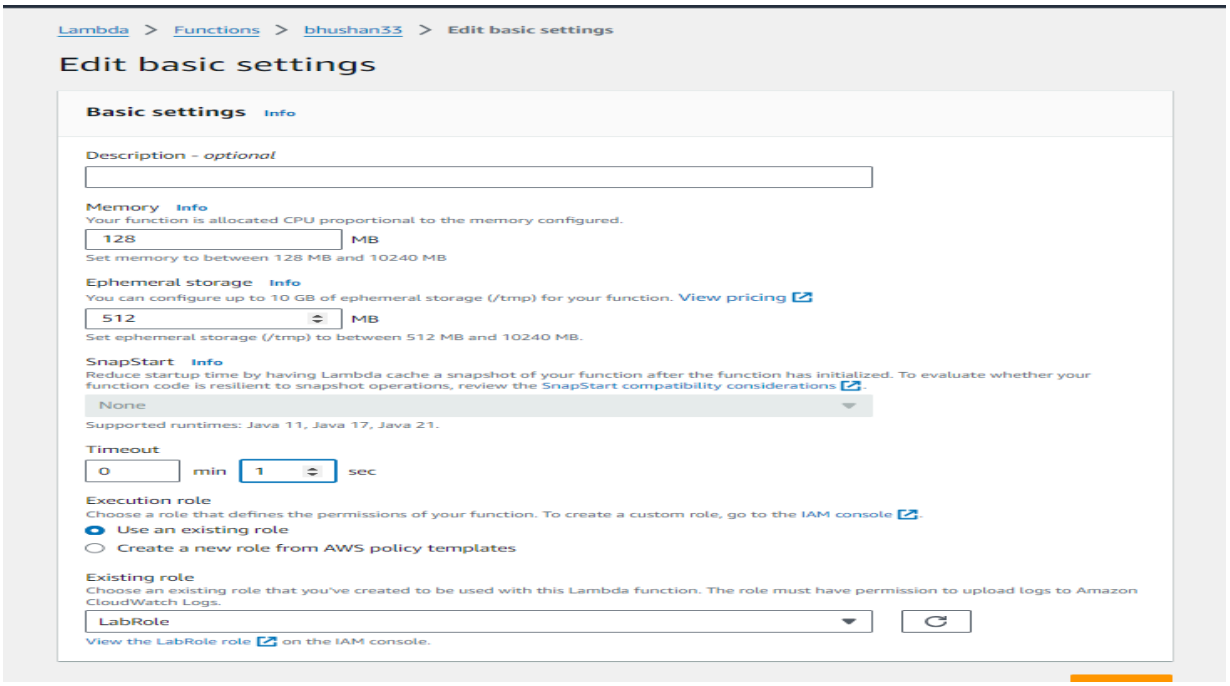


Step 3: Set up configurations and test events

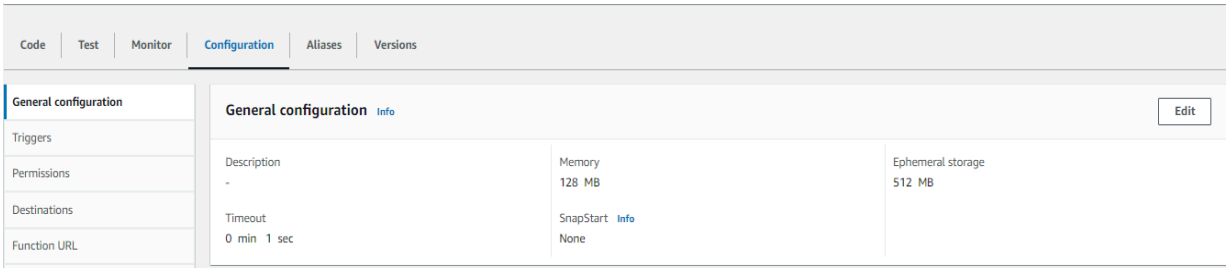
1) Just above the test code, you would find Configuration, click on it. Then click on Edit.



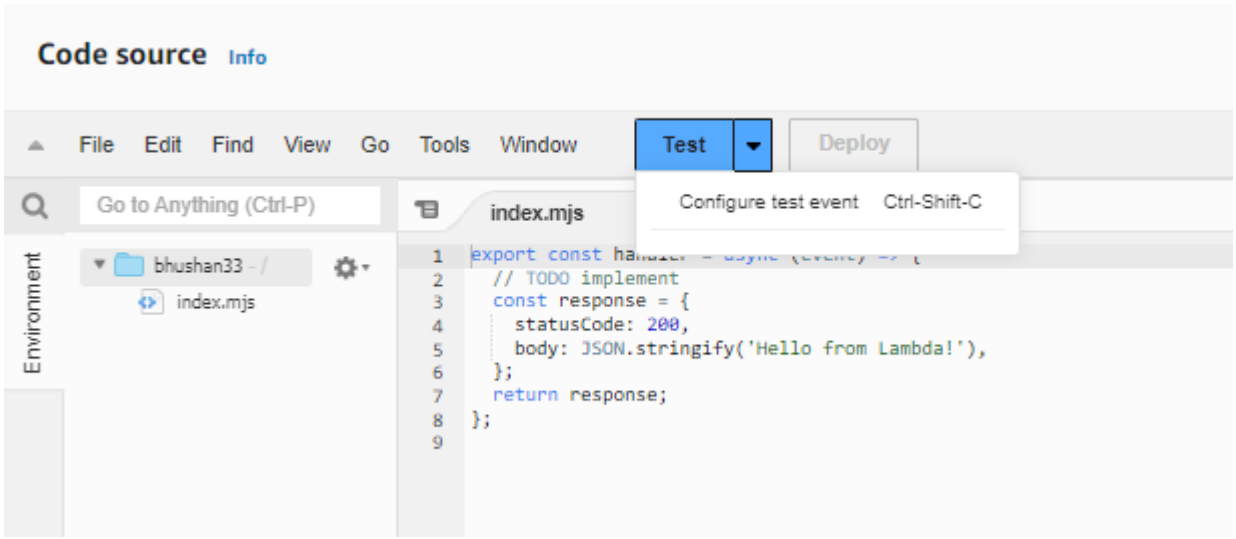
2) Here, change the Timeout to 1 sec. This is the time for which the function can be running before it is forcibly terminated.



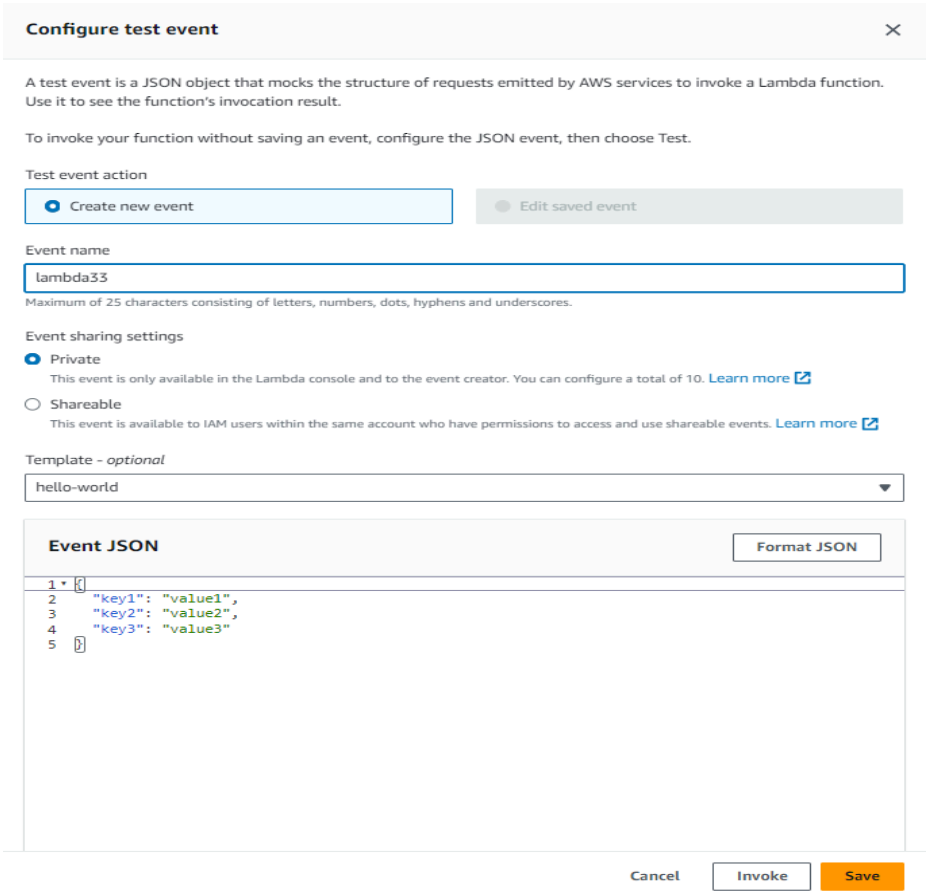
3) We can see the executed changes.



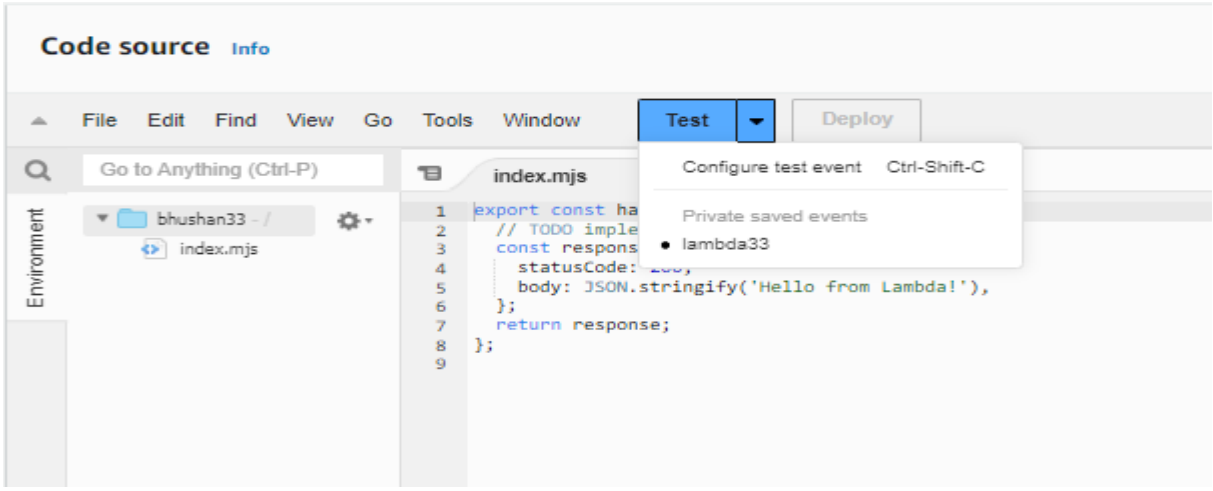
4) Switch back to the code tab. Click on the dropdown arrow near test. Then select configure test event.



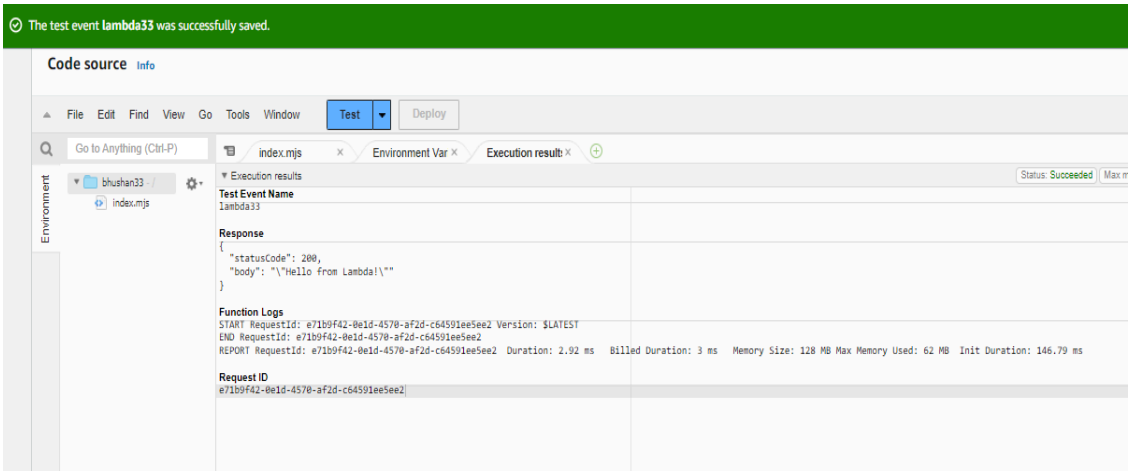
5) Here, create a new event, keep the other options default and save the event.



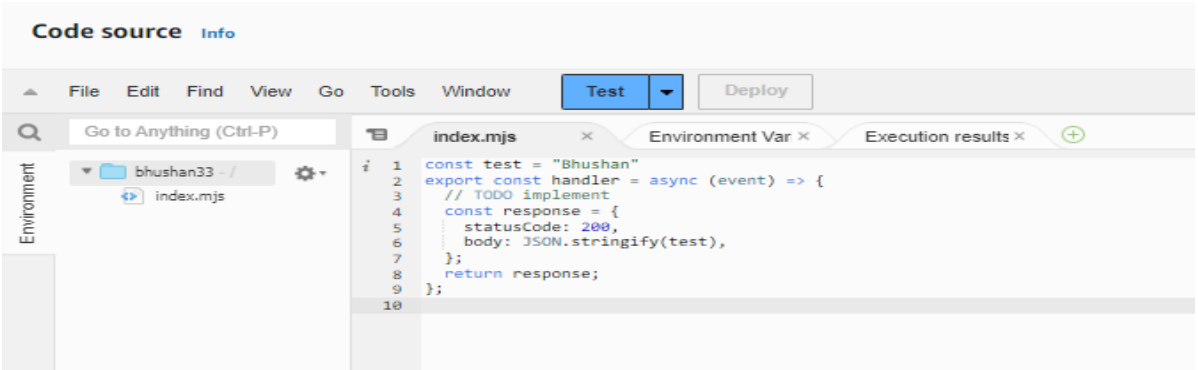
6) Now, again click on the dropdown. This time, select the event you have created. Then, click on TEST.



7) We can see the expected output for the sample code.



8) For a test, declare a string and call it in line 6. After making the changes click on deplpoy.



9) Run the test. We can see that the string we declared has come in the output.

Conclusion:

In this experiment, we effectively investigated the AWS Lambda service by developing and configuring Lambda functions using Python, Java, or Node.js. We discovered how to establish a Lambda function, tweak its settings (like modifying the timeout duration), and evaluate the function with personalized events. Throughout this experience, we noted how Lambda manages executions, including timeout handling and producing expected results according to modifications in the code.