Lambda

AWS Lambda is a serverless compute service from [Amazon Web Services (AWS)](https://aws.amazon.com/lambda/features/), allowing you to run code without managing servers. You simply upload your code, and AWS Lambda takes care of the underlying infrastructure, scaling, and management. This allows you to build various applications and services quickly and efficiently.

### AWS Lambda - GeeksforGeeks Key Characteristics

| **Feature** | **Description** |
| --- | --- |
| **Serverless** | No need to manage infrastructure or servers |
| **Event-driven** | Executes in response to triggers |
| **Auto-scaling** | Automatically handles any number of requests concurrently |
| **Short-lived** | Each execution runs for up to 15 minutes |
| **Language support** | Python, Node.js, Java, Go, .NET, Ruby, and custom runtimes (e.g., Rust) |
| **Billing** | Pay only for the compute time your code uses (per millisecond) |

**Core Concept**

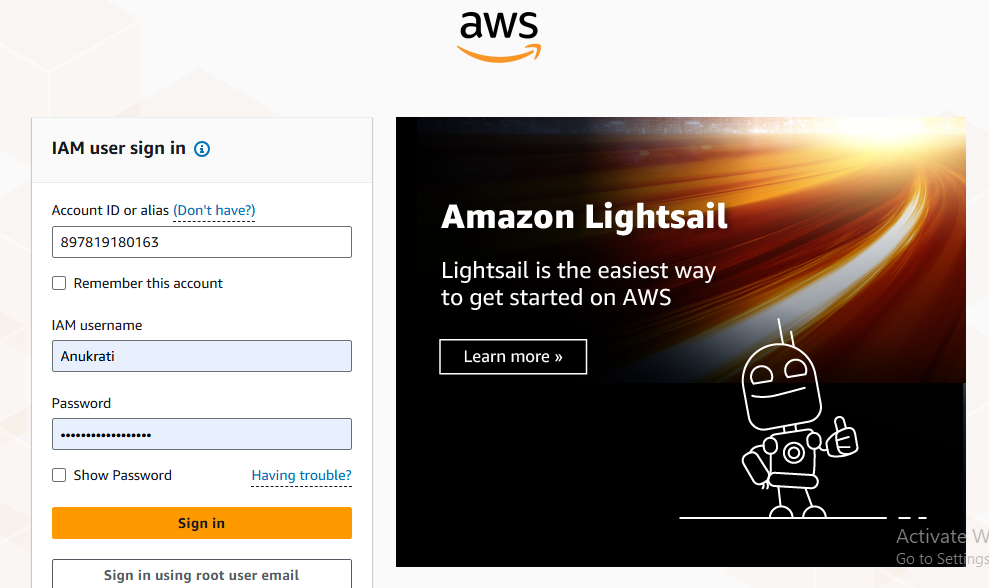
AWS Lambda runs your code **in response to events**, such as:

* HTTP requests via **API Gateway**
* File uploads to **Amazon S3**
* Changes in **DynamoDB** or **SNS** messages
* Scheduled times via **EventBridge (CloudWatch Events)**

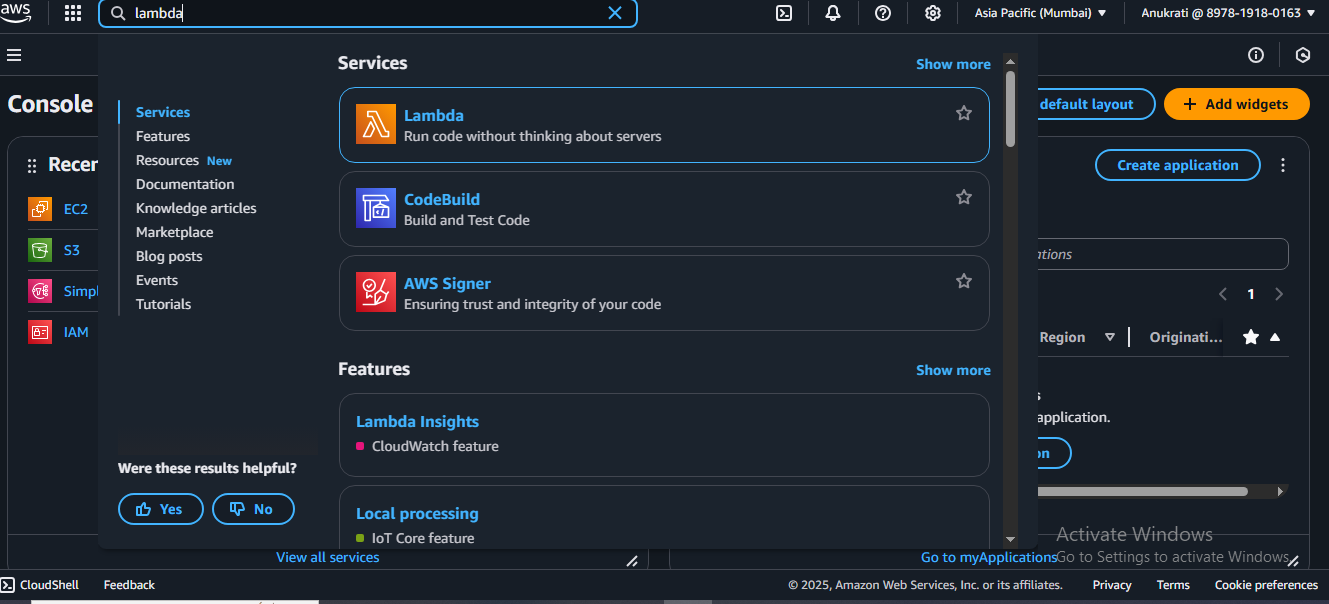
Lab steps:

**Step 1: Sign In to AWS Console**

* Go to: AWS console.
* Sign in with your AWS account.

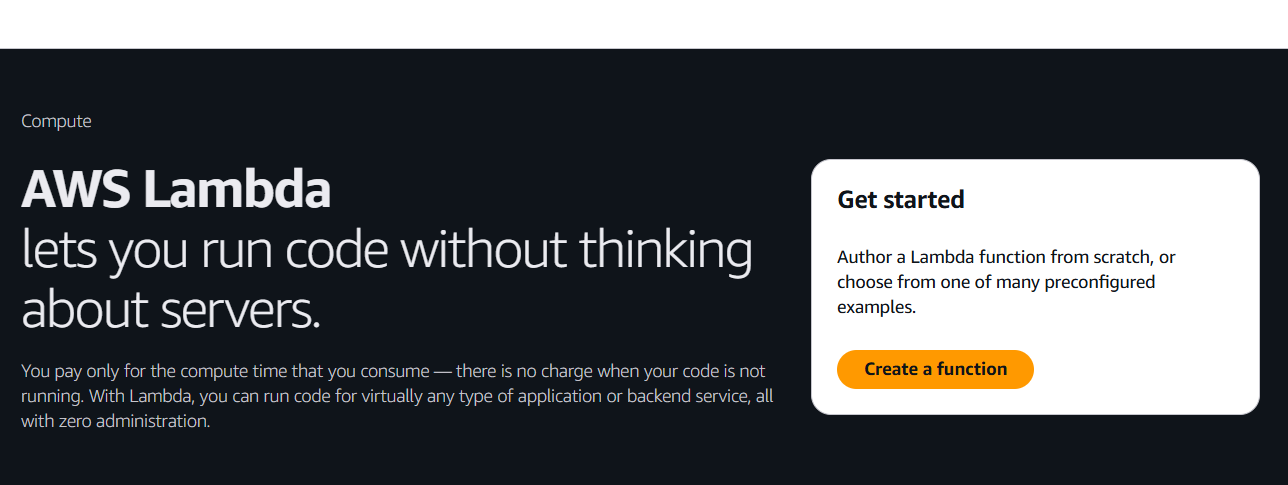


**Step 2:**Log in to your AWS console and search for Lambda. As shown in the following image.

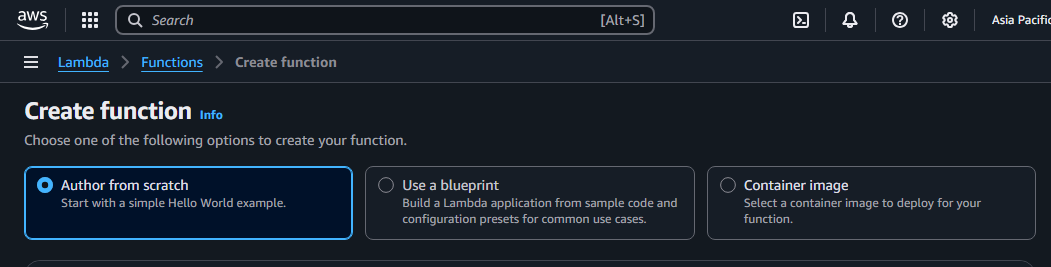


**Step 3:Create a Lambda Function**

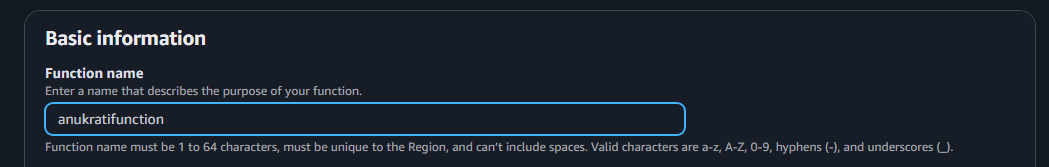
* Go to **AWS Lambda > Functions > Create Function**
  + Runtime: Choose (e.g., Python, Node.js, Java, etc.)



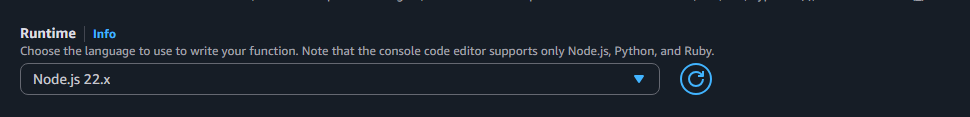
Author from scratch



* + Name your function

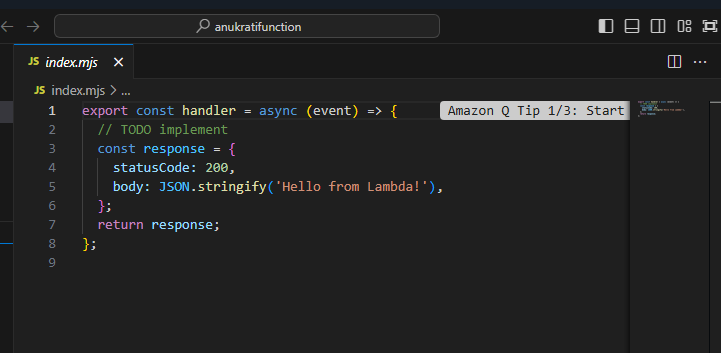


* + Runtime: Choose (e.g., Python, Node.js, Java, etc.)



* + Execution role: Choose existing or create new with the role above
  + Click “Create function”.

Step 4: **Write or Upload Code**

* In the **Function code** editor:
  + You can write code directly in the console or upload a .zip or container image.
  + Example (python):
* Click on **Deploy** to save the code.

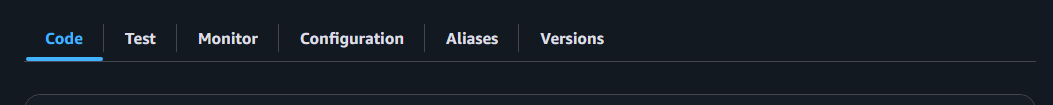
### 

### Step 5: Add a Trigger (Optional, e.g., API Gateway or S3)

1. Scroll to **Function overview**.
2. Click **+ Add trigger**.
3. Choose an event source, such as:
   * **API Gateway** (for REST APIs)
   * **S3** (for file upload)
   * **CloudWatch Events** (for scheduled tasks)
4. Configure the trigger settings.
5. Click **Add**.

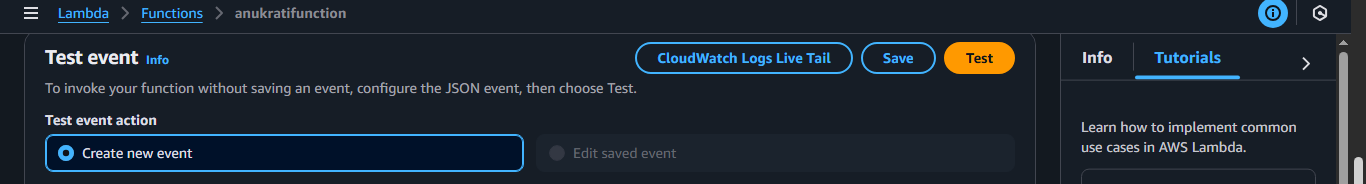
**Step 6: Test the Lambda Function**

* + 1. Click **Test** at the top.

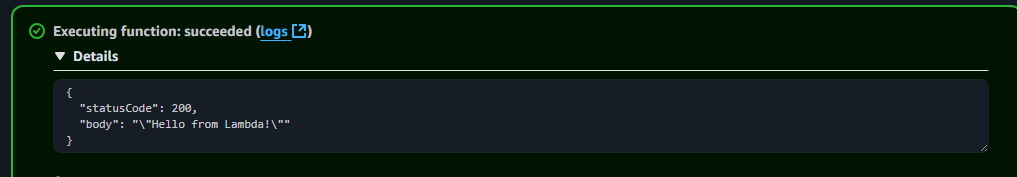
2. Create a new test event:

* + Choose **“Create new event”**.
  + Give it a name, and use a sample payload or leave as default.

Click **Test** again to invoke the function.

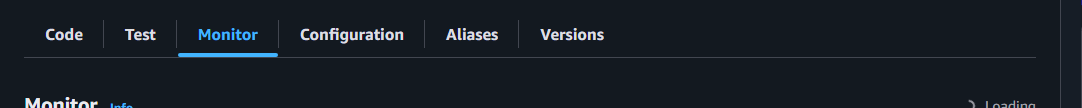


* You should see output in the console under **Execution result**.



**Step 7: Monitor and Logs**

* Go to the **Monitor** tab to view metrics (invocations, duration, errors).



* Logs go to **CloudWatch Logs**
* Use **Amazon CloudWatch Metrics** for invocation count, errors, duration, etc.

**Step 8: Modify, Version, or Delete**

* You can:
  + Add environment variables
  + Set concurrency and memory limits
  + Create **aliases** and **versions** of your function
  + Delete the function when done

