**What is an API (first)?**

* **API** stands for **Application Programming Interface**.
* It’s a **set of rules** that lets two programs talk to each other.
* Example: Your phone’s weather app talks to a weather service API to get the latest temperature.

## ****What is REST?****

* **REST** stands for **Representational State Transfer**.
* It’s a **style/rules** for designing APIs so they are **simple, scalable, and easy to use**.
* REST uses **HTTP (the same protocol websites use)**.

## ****What is REST?****

* **REST** stands for **Representational State Transfer**.
* It’s a **style/rules** for designing APIs so they are **simple, scalable, and easy to use**.
* REST uses **HTTP (the same protocol websites use)**.

### ****Why REST APIs are popular?****

* Simple to understand.
* Works over the web using HTTP.
* Can be used by web apps, mobile apps, IoT, etc.

**In one line:**  
A **REST API** is a simple way for different apps to communicate over the web using standard HTTP rules.

**how we rest apis configured to interact between two programing language?**

### ****Using the E-commerce & Bank Example:****

* The **Bank** provides a payment service.
  + It **configures a REST API** endpoint like:  
    POST https://bank.com/api/payment
  + It defines what data it expects (JSON with account number, amount, etc.) and how it responds.
* The **E-commerce app (Java)** just **calls** the bank’s API.
  + It does **not need to create its own REST API** for payment.
  + It just needs to **send HTTP requests** and **process responses**.

## ****1) Why do we need REST APIs between two languages?****

* Programs written in **different languages** (like Java and Python) cannot directly call each other's functions because their code is different.
* But **both understand HTTP and JSON** (or XML).  
  REST APIs act as a **translator** using HTTP and a **common data format (like JSON)**.

## ****2) How REST API Makes This Work****

1. One program **exposes an API** (makes itself available on an HTTP endpoint).
2. The other program **calls that API** using HTTP requests.
3. They exchange data in a **neutral format (JSON/XML)** that both understand.

## ****3) Real-Time Example****

### Scenario:

* **Frontend:** Written in JavaScript (React web app).
* **Backend:** Written in Python (Flask framework).
* **Goal:** Show list of products from Python backend on the React website.

In this specific project primarily I am using 3 components

1. DynamoDB (for backend database storage)
2. Lambda (Act as a web server)
3. API Gateway (Acr as a frontend)

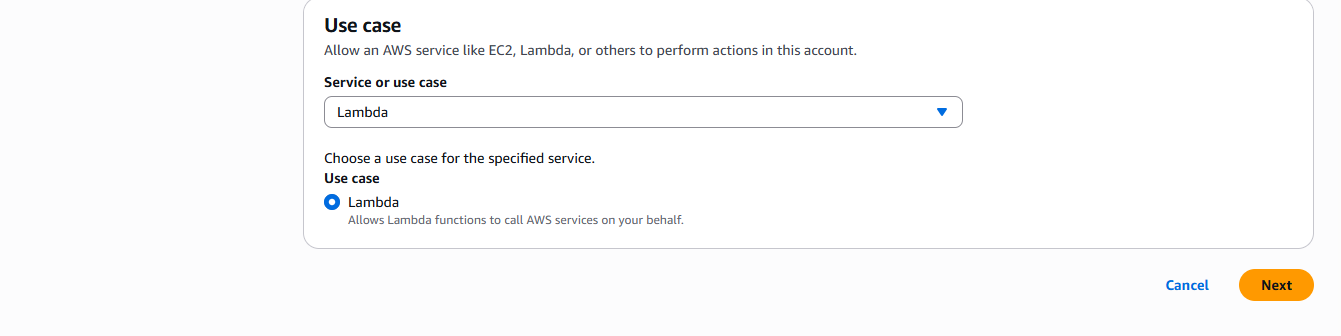
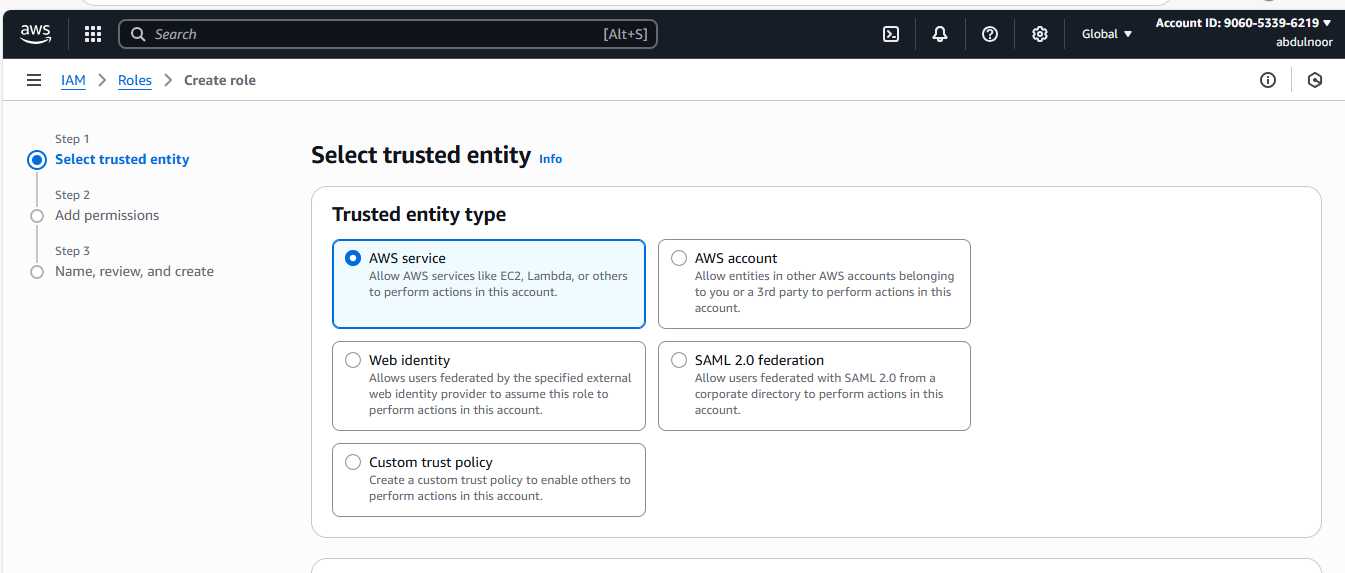
These 3 are server less components

Whenever any customer trying to access my application the request will come to API Gateway, APG is going to read that API call, and it is going to process by the lambda and I am going read/write the information at backend dynamodb table.

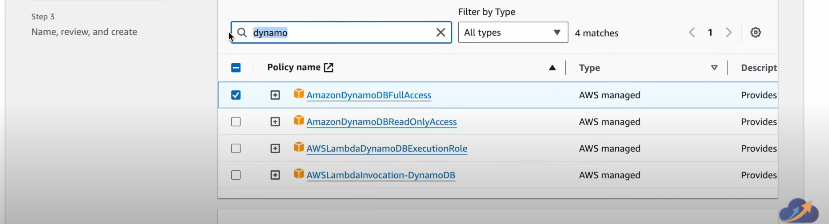
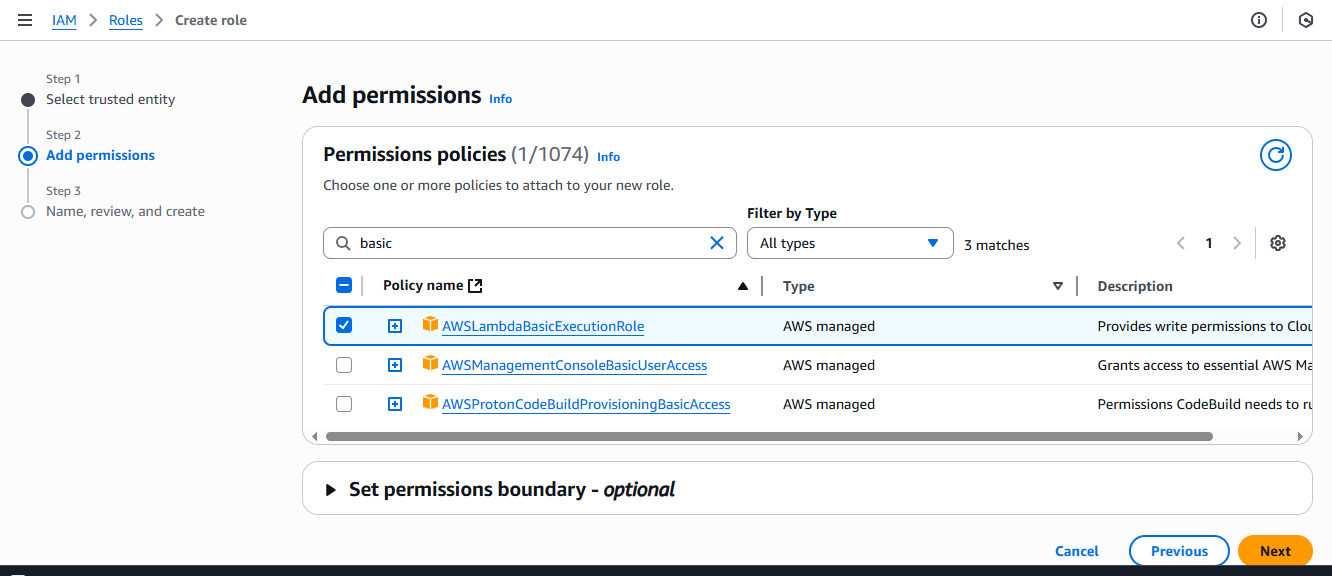
Step 1. IAM Role for this LAMBDA Function. So this Lambda unction should able to write data at backend dynamodb. So create a role and associate role lambda function.

I am creating this role for lambda of AWS service

Goto IAM – Click on create role

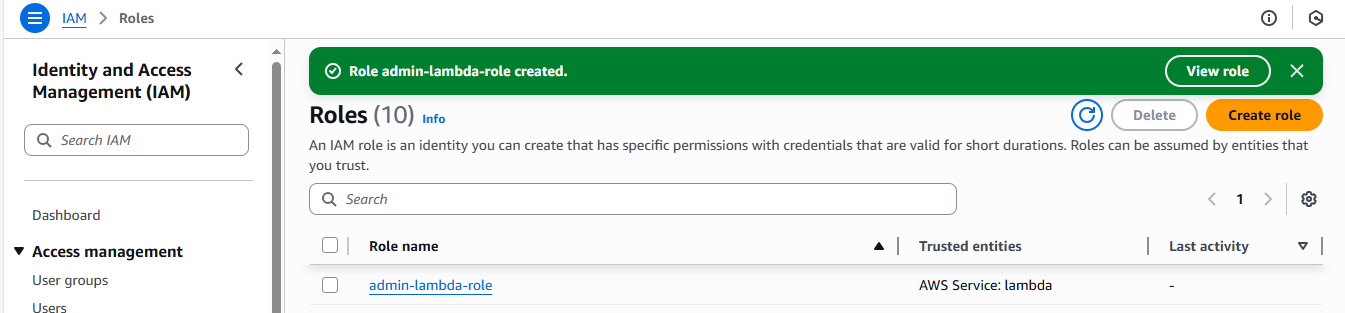


Click on Next



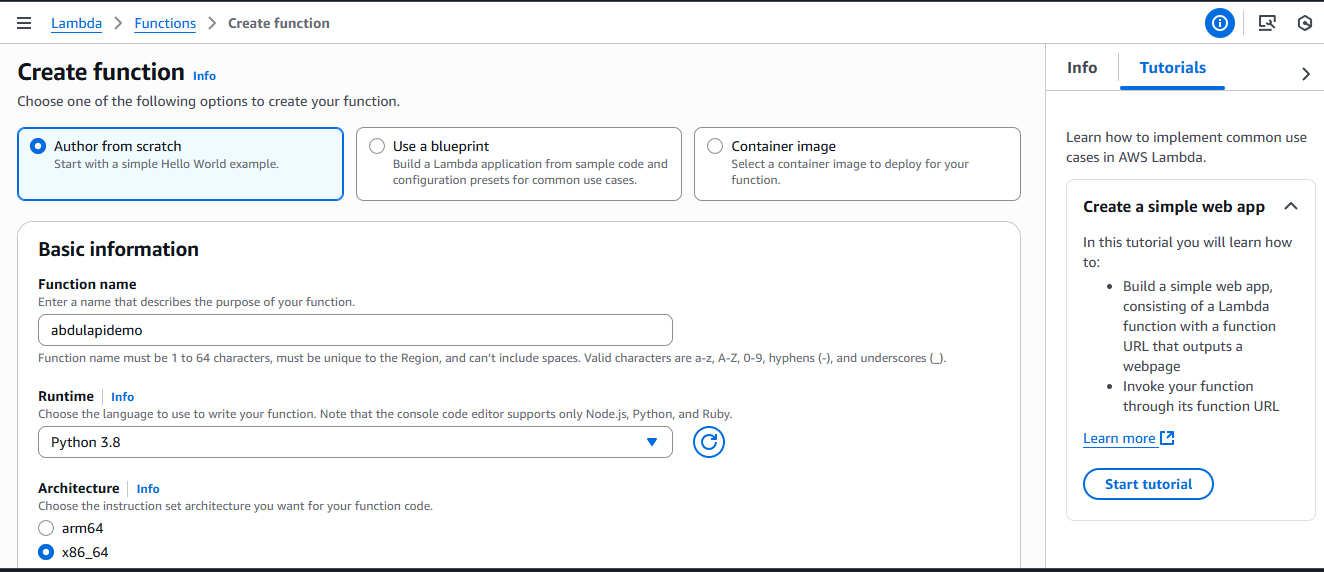
Click on Next

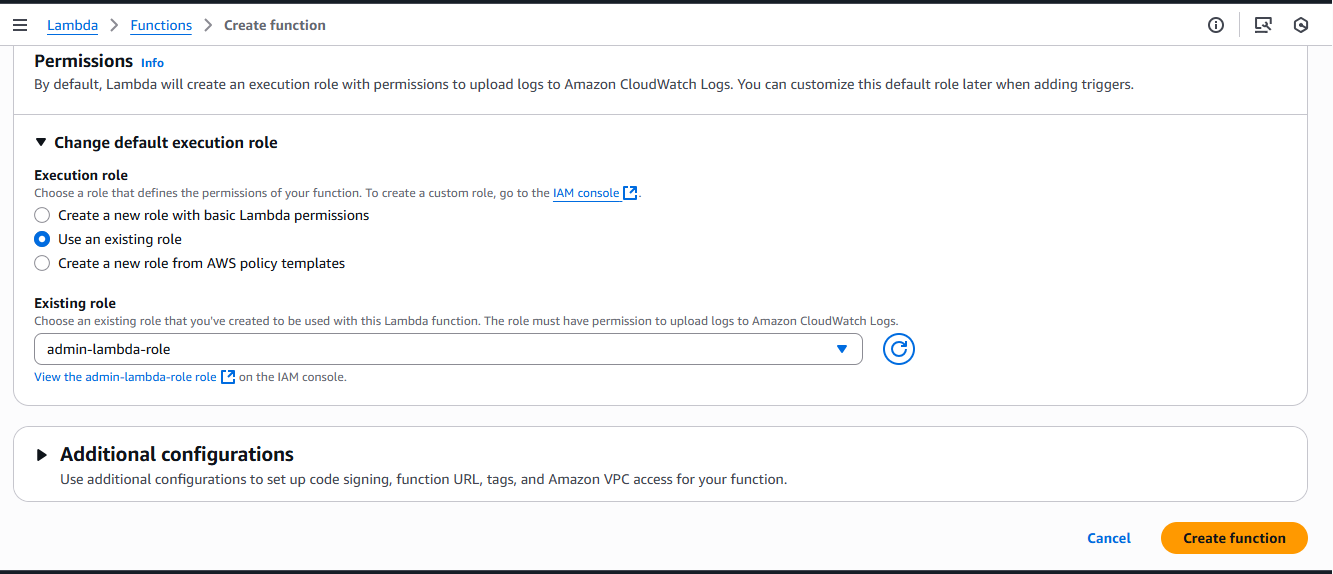
Give name admin-lambda-role



STEP 2: Create a LAMBDA function

Goto Lambda – create lambda function

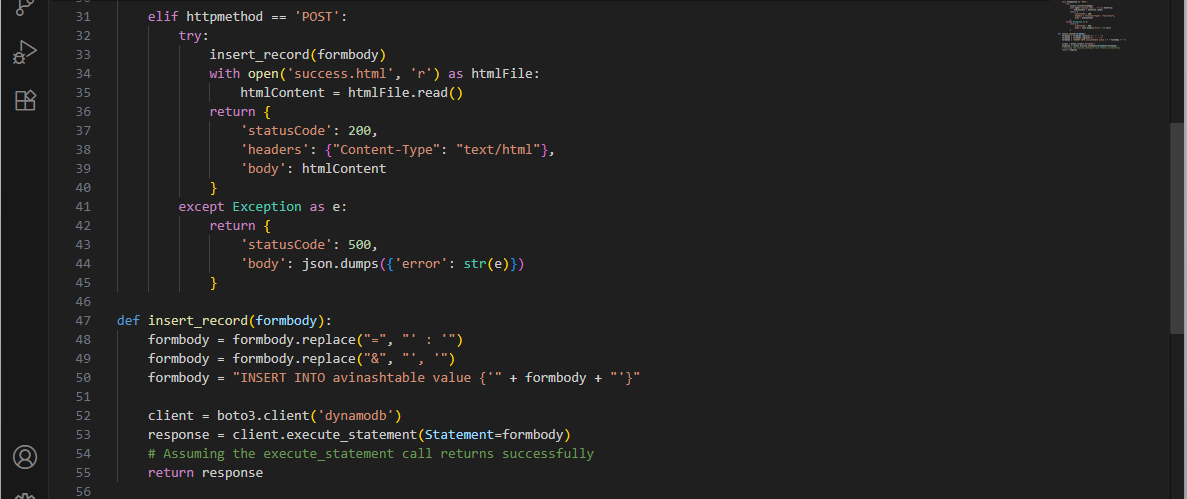
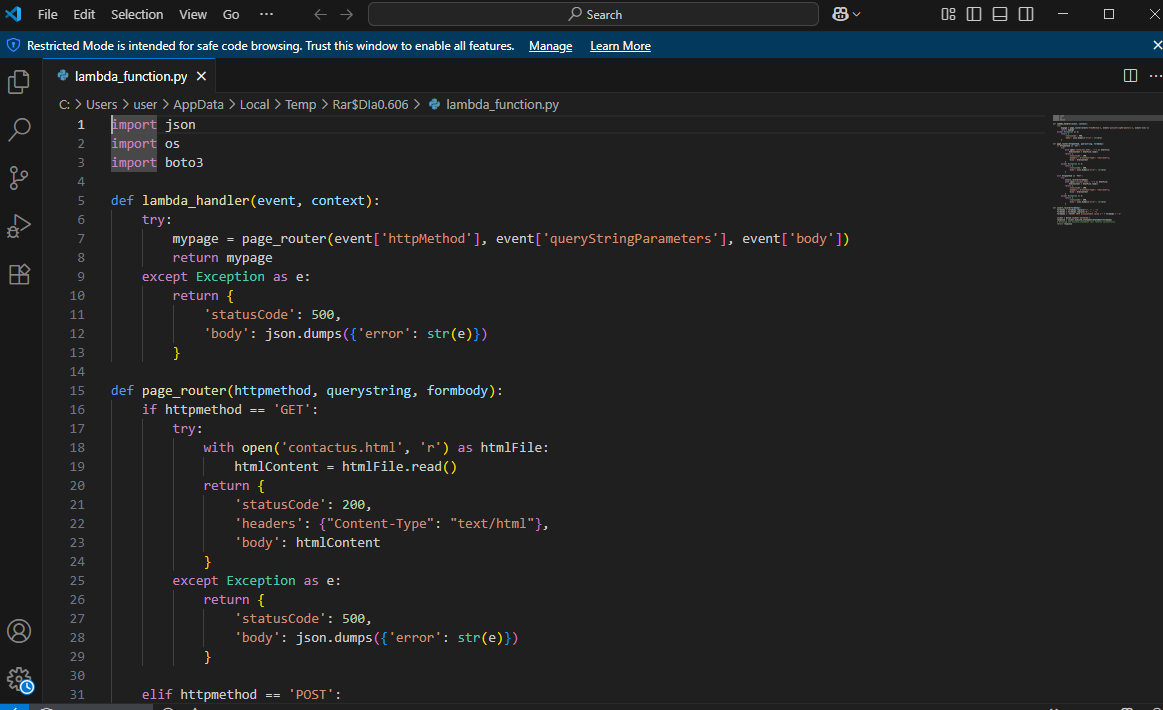




Click on create a function

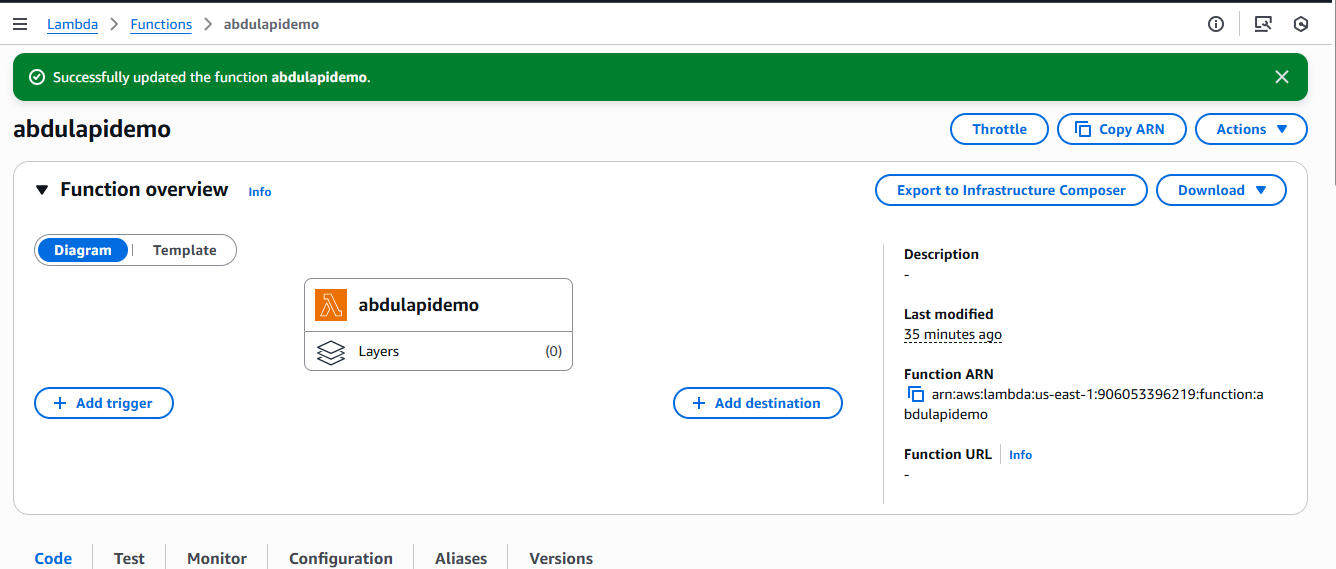
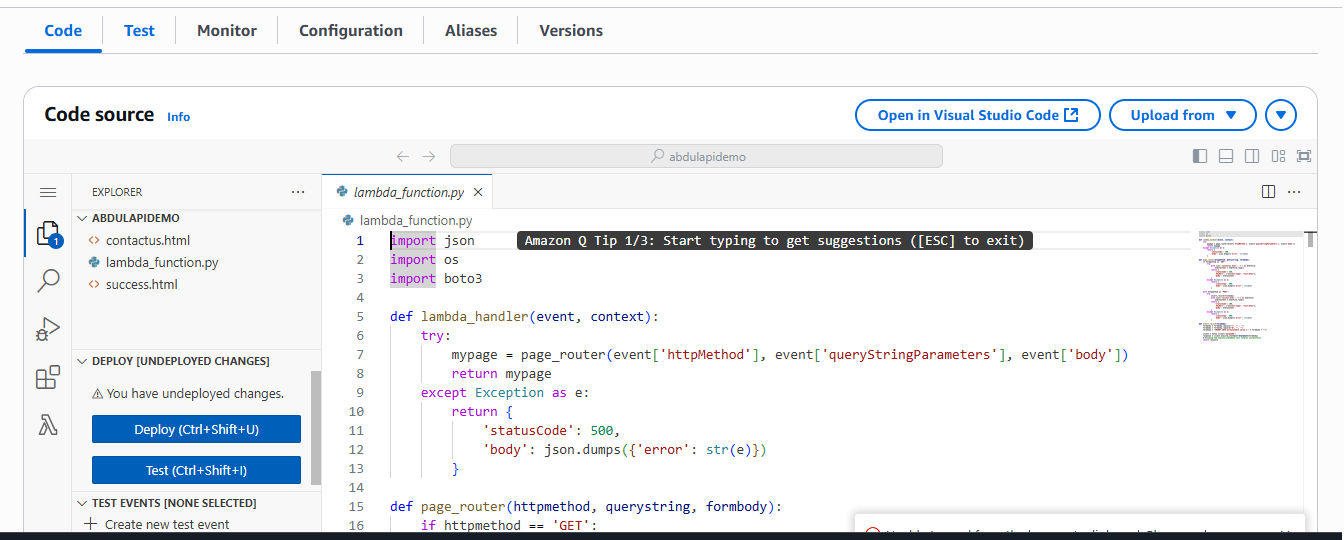
Lambda function contains 3 files

1. Contact us page : whenever customer trying to access our application, first this contact page will open. Whatever the information he is going to fill here details and click on submit, it actually stores in backend database.
2. If the data succesully written to the backend dynamodb table, the **success.html** page going to load
3. And also this lambda unction is going to through the an error if it has any error sign.

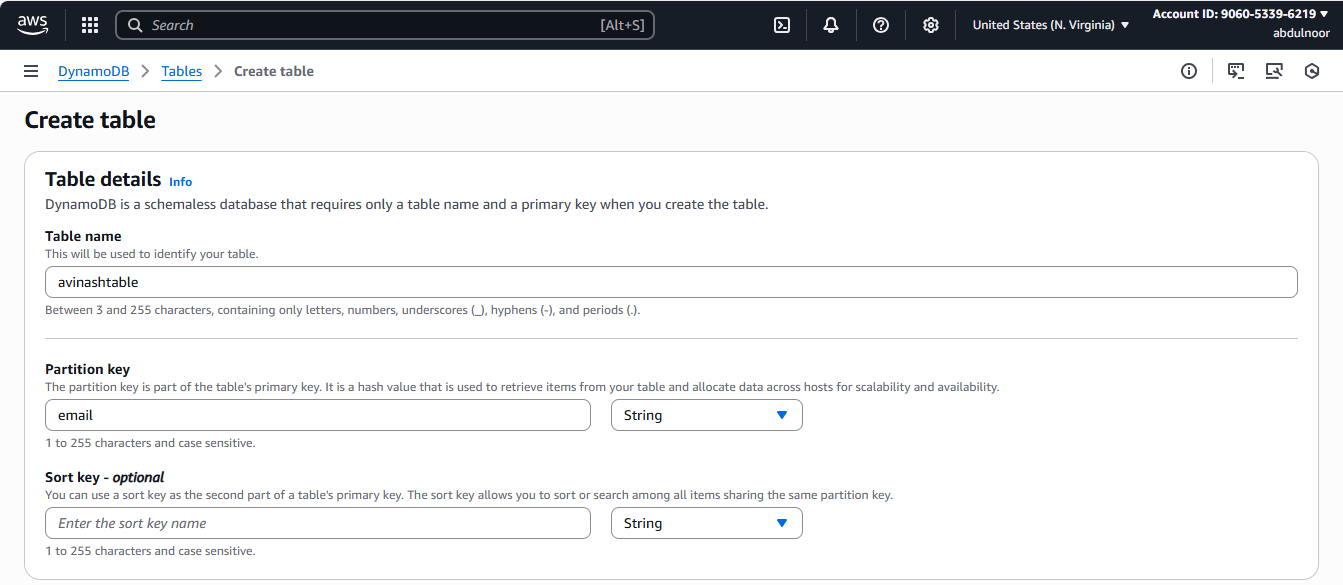
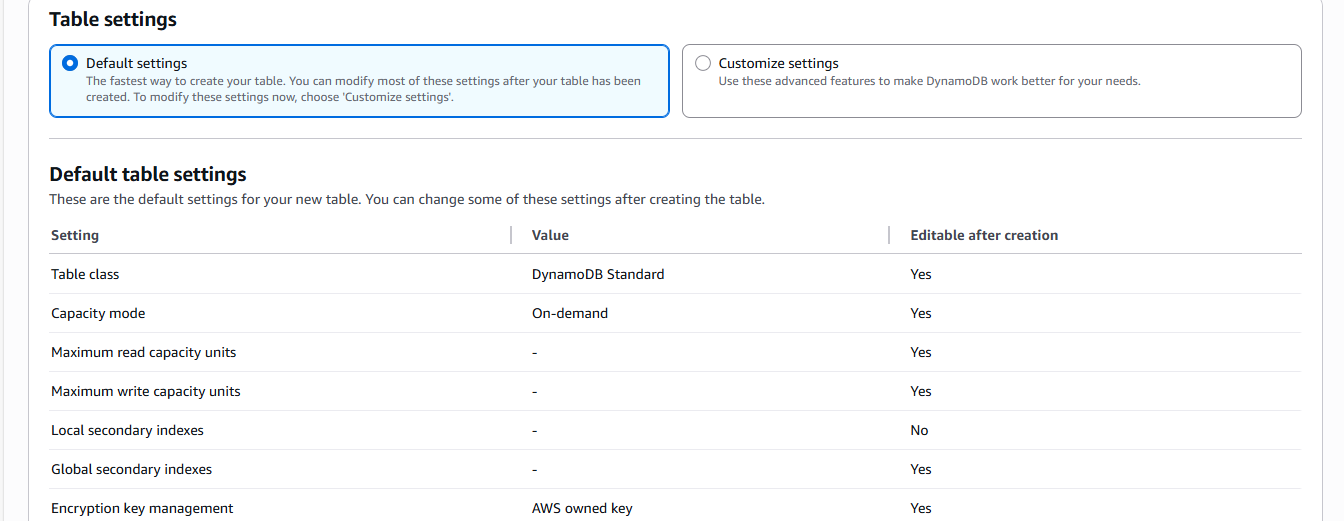
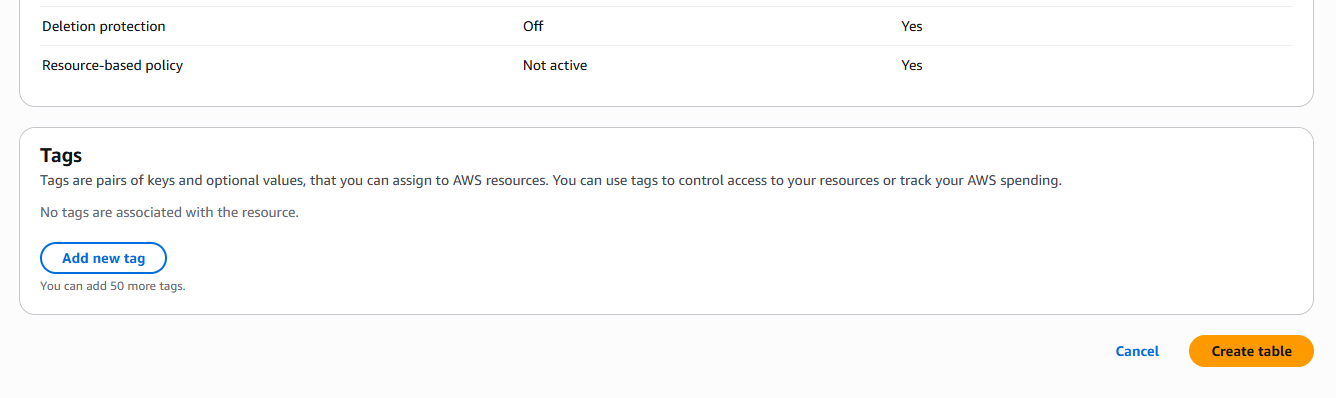


Let me explain this lambda function what this is going to do.

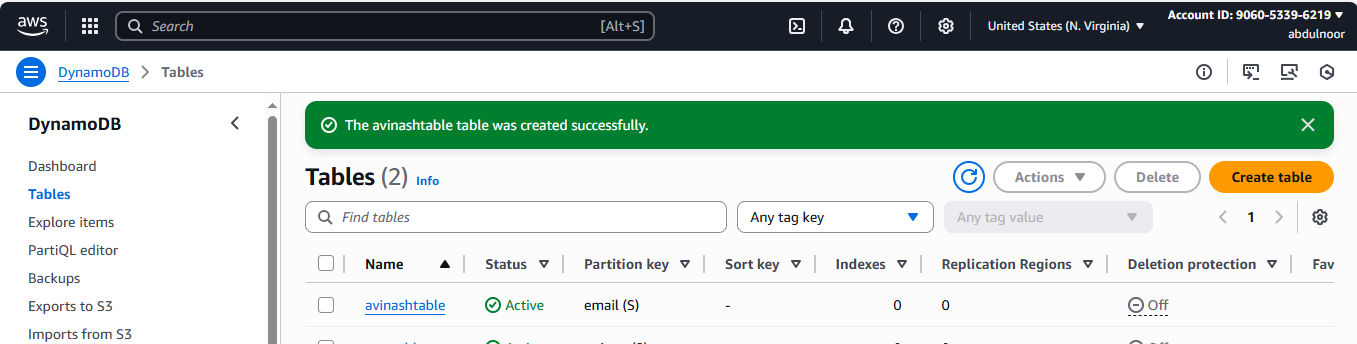
* Basically this lambda function is going to **act as a simple webserver, so it serves “contactus.HTML” page** whenever you get request from our API Gateway. So it going to load contact.html.
* And whenever post request is coming/httpmethod If it is a post request, what’s happening this success.HTML page is going to load and not only that or this post request what’s happening a Dyanamodb is loading here and in DynamoDb there is table name here avinashtable.
* Within this avinashtable I am going insert all this information whatever the information I am getting it from this contctus.HTML
* All that information I am going write in this backend dynamodb table.
* So this is LAMBDA code is doing.

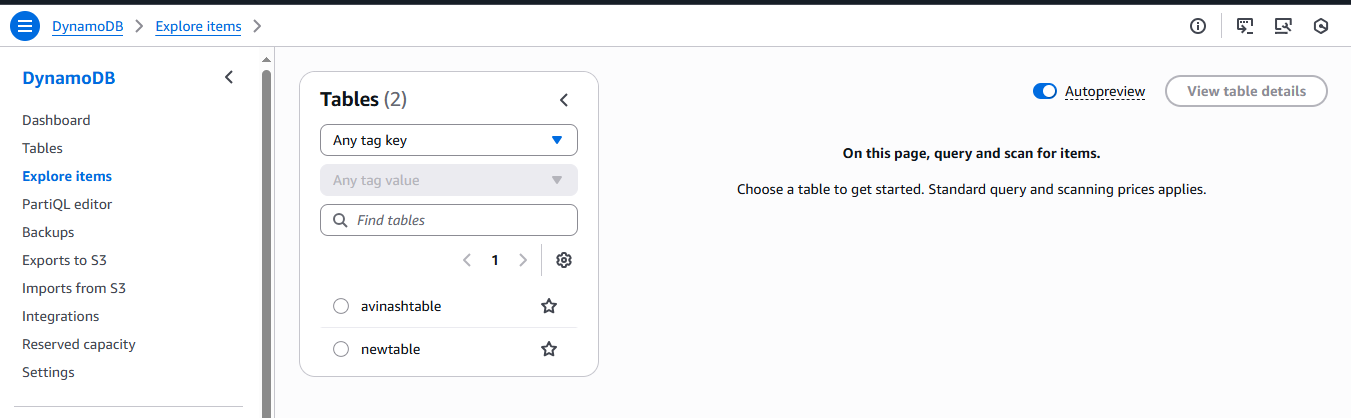
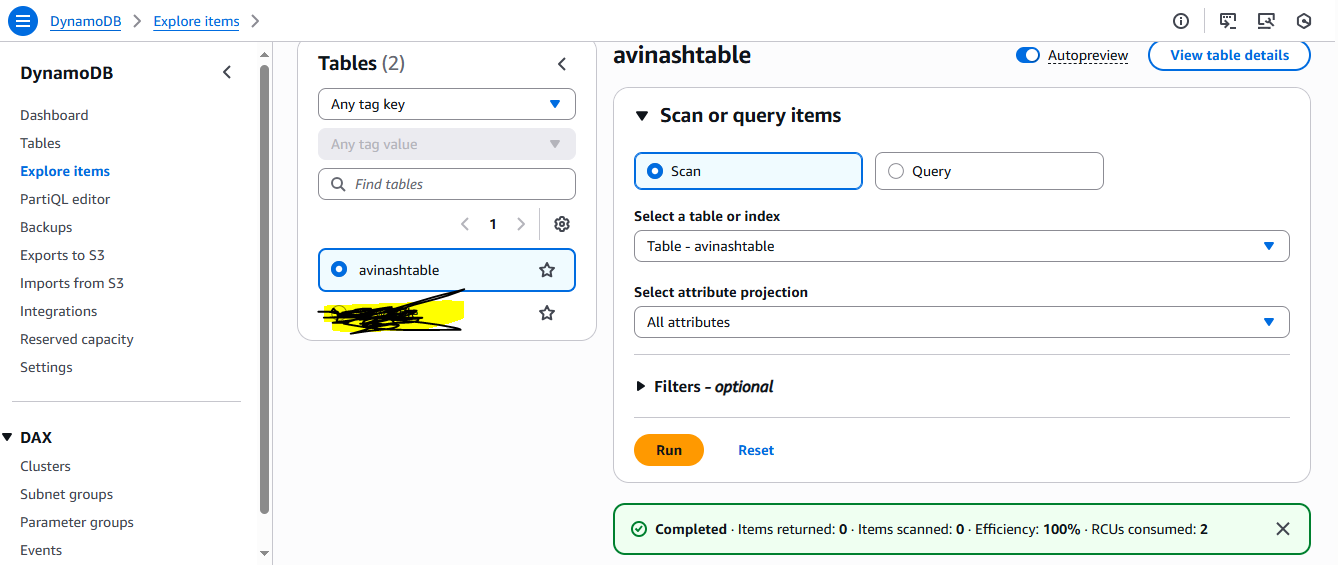
Step 3: Go to dynamodb – tables – Create table

Click on create table

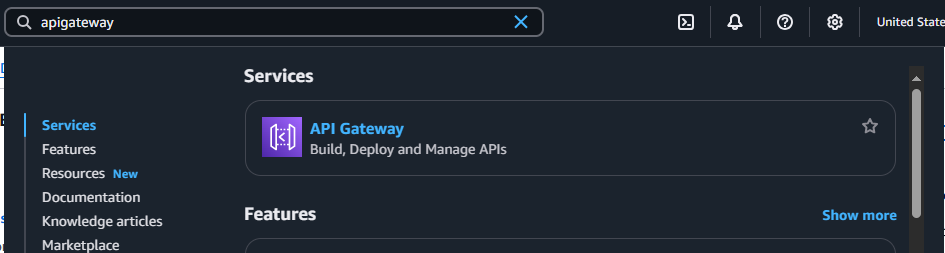


Click on exploring items

So we don’t have any items.

**Now configure API Gateway**

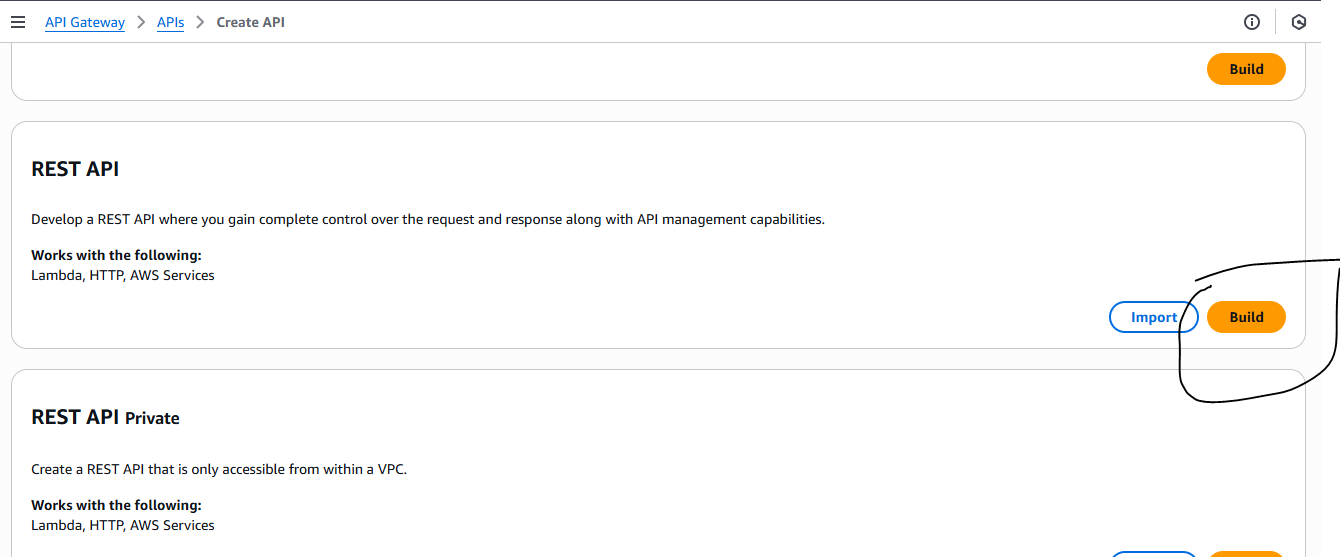


Click on API Gateway

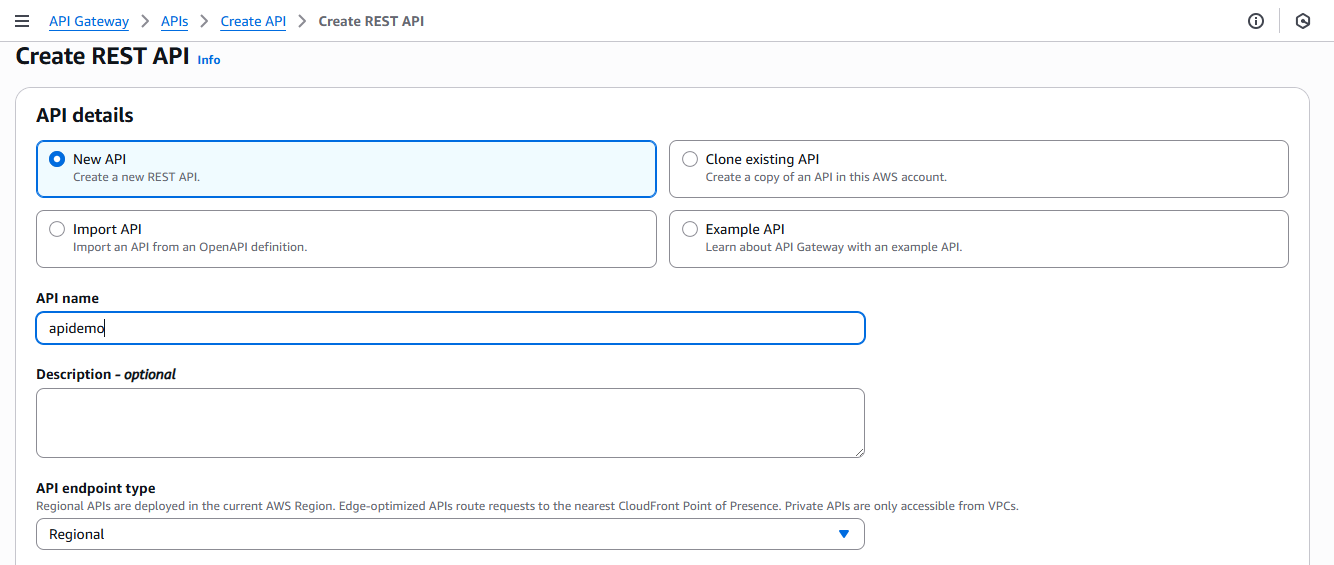
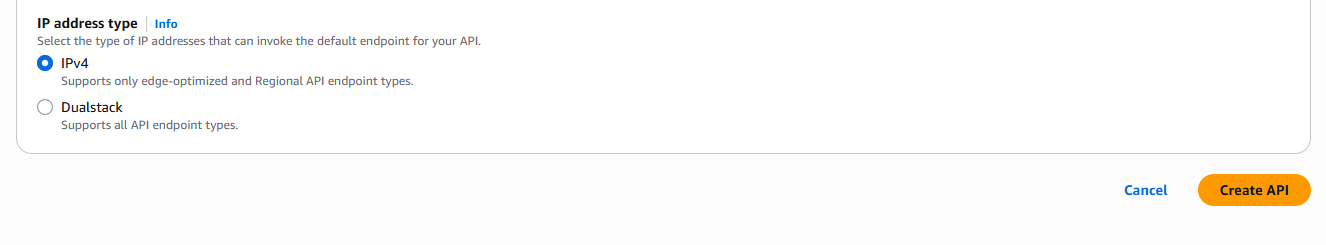
Click on create an API

Within this API Gateway we have option REST API – This restapi works over the internet

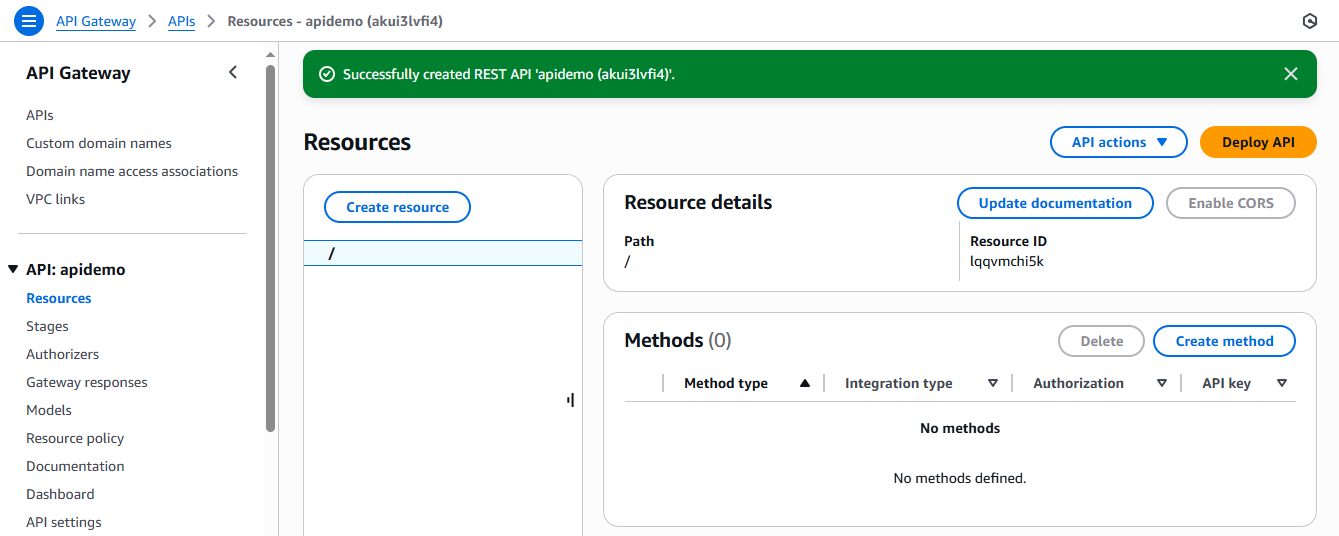
Click on Build - REST API



Click on build

Click on Create API

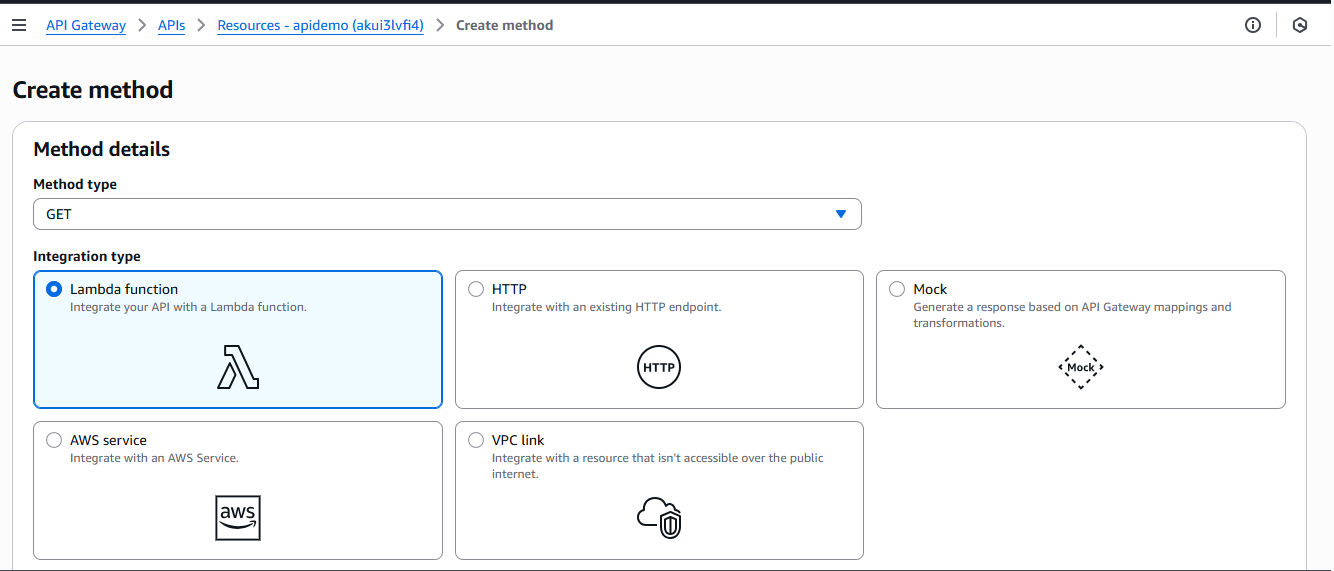
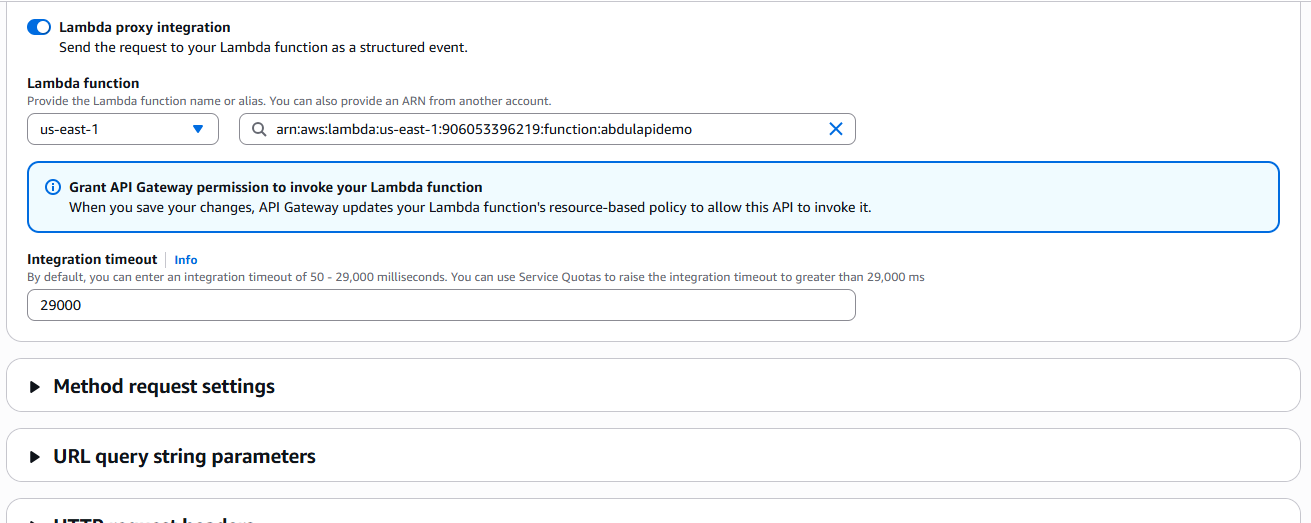


Now create a Method

(Lambda function here we are primarily performing GET operation, and POST operation. So we need to conigure both methods here

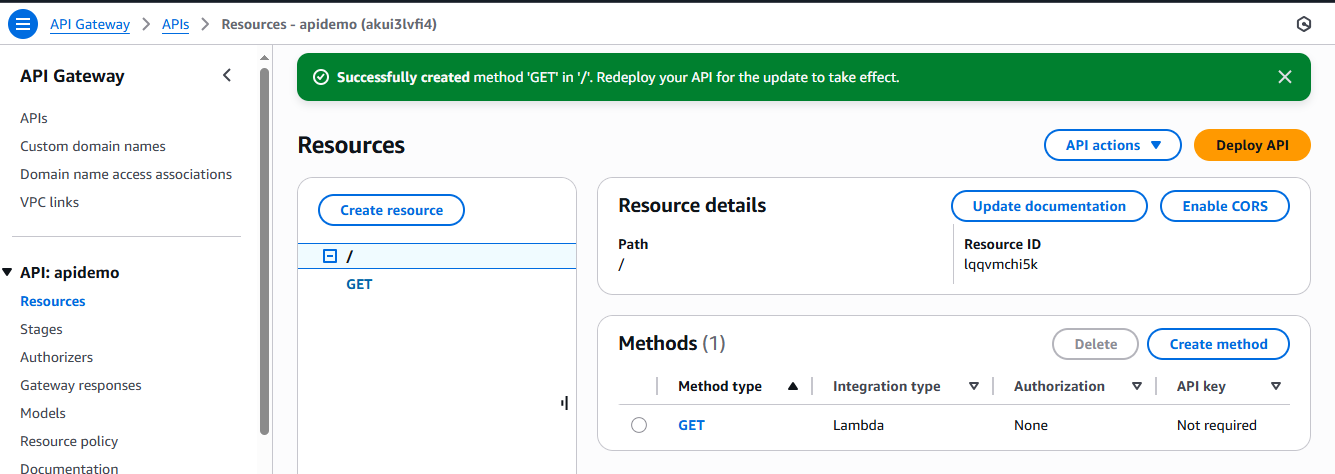
Clcik on create a method

Under method type we have a option GET

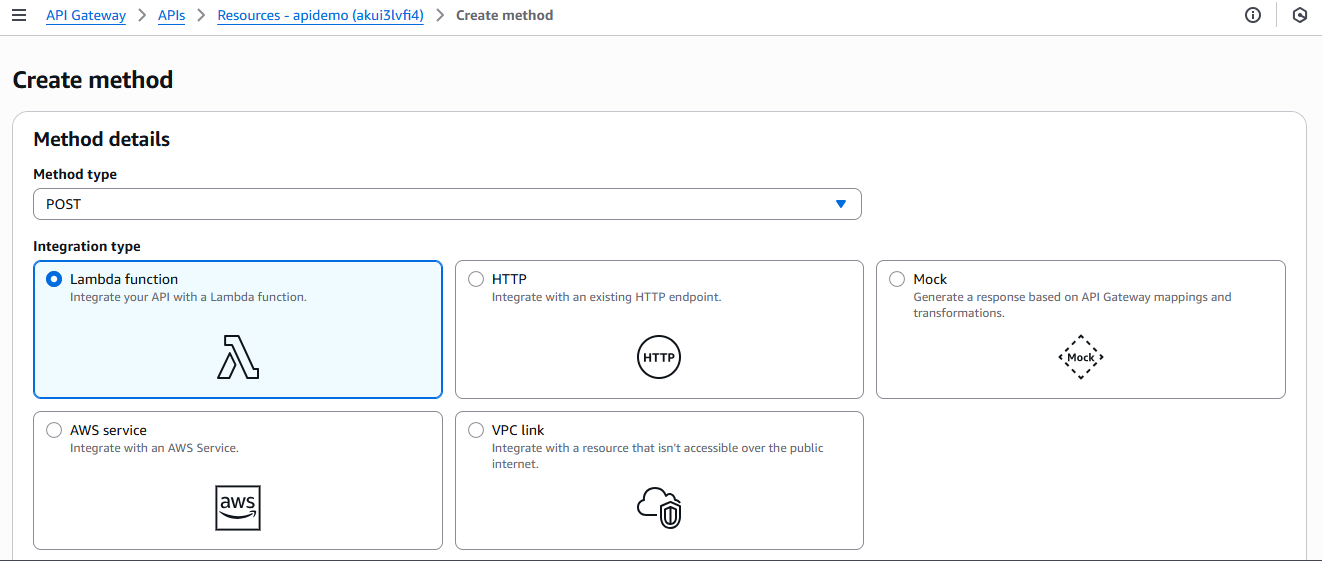
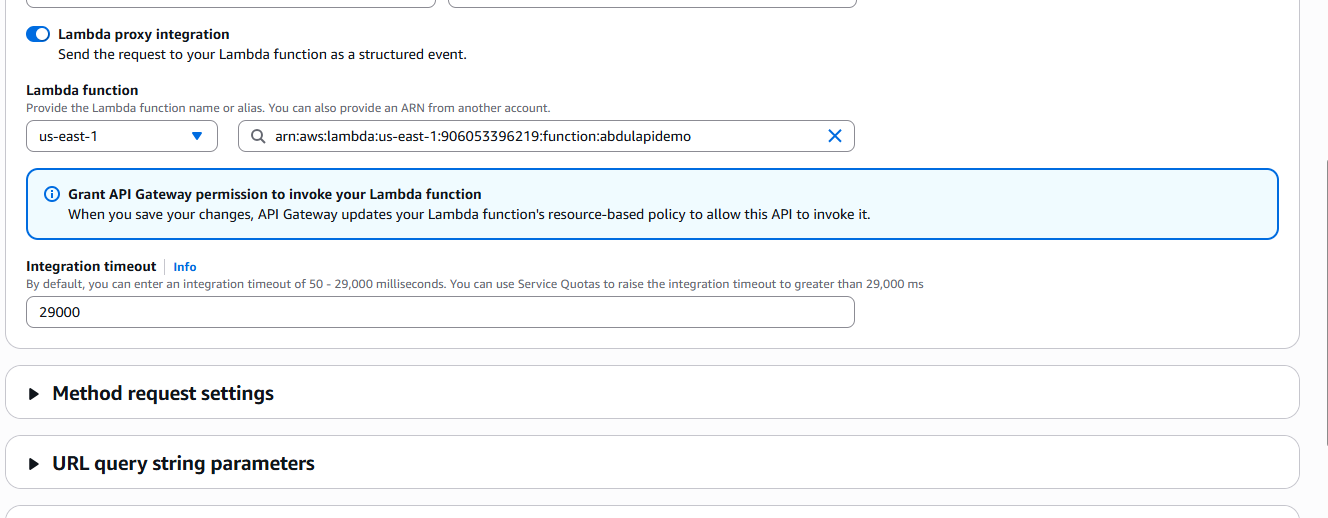
Clcik on create method

Click on / root

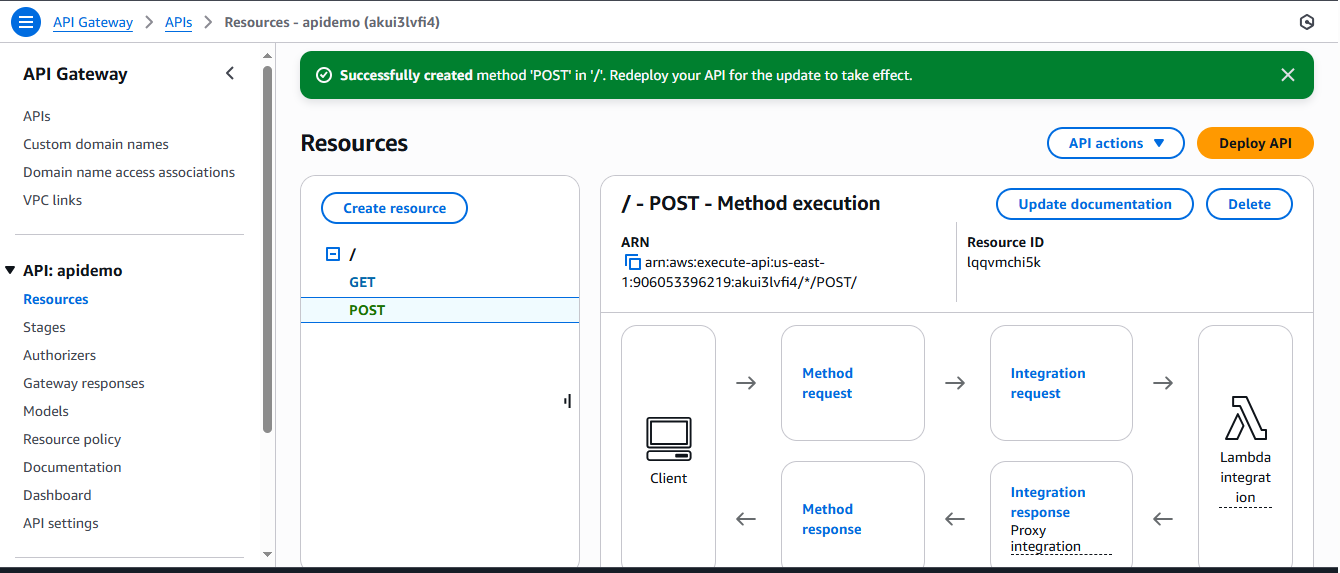


Click on method (to create one more method)

Now select POST as method type

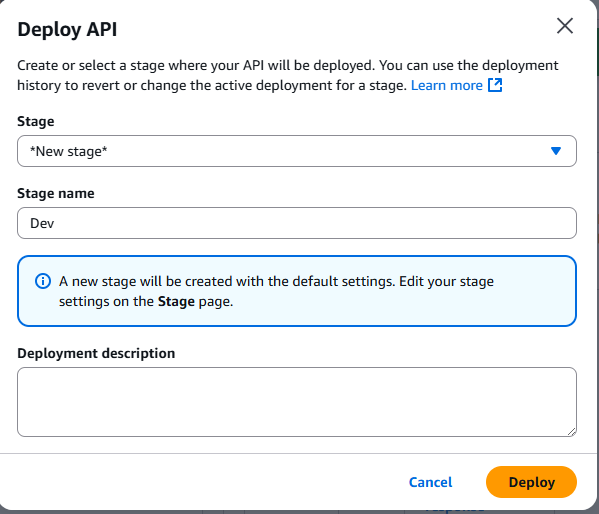
 

Click on create method

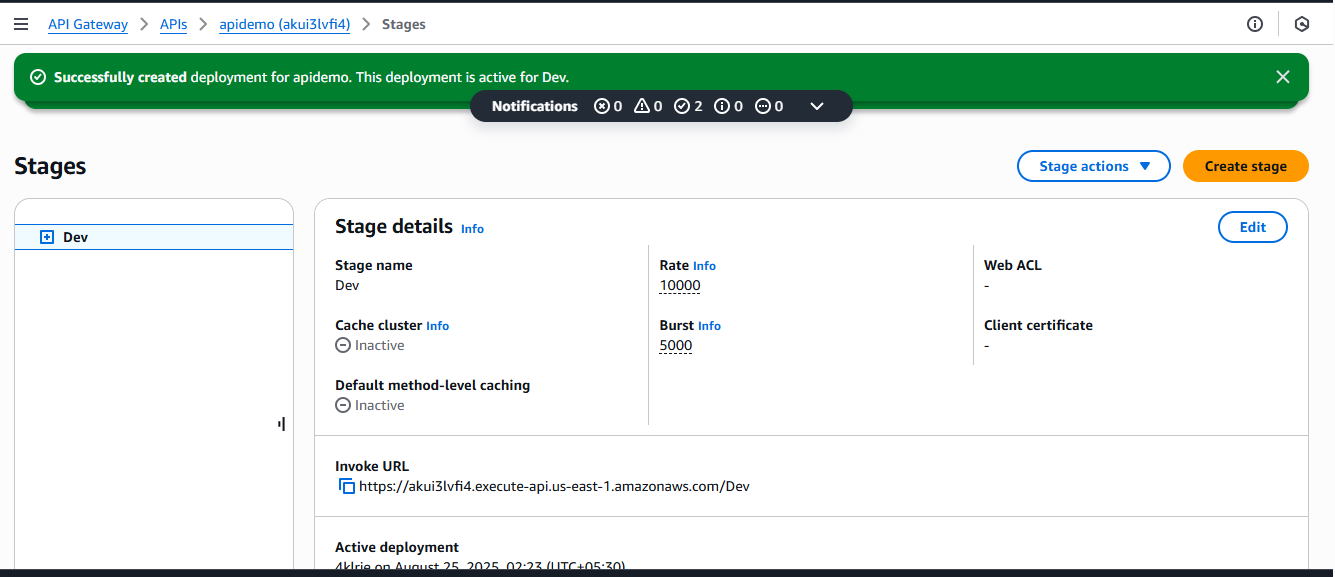


we have created API calls and now we need to deploy API

click on Deploy API



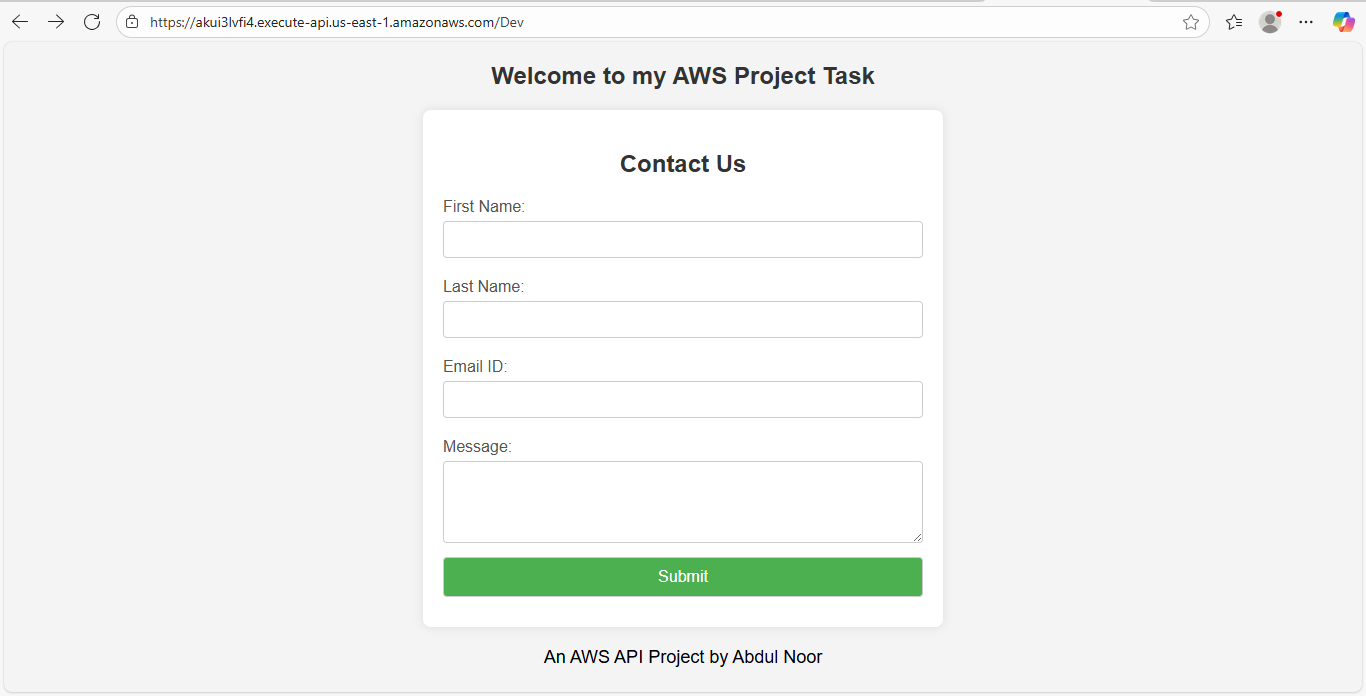
Click on Deploy



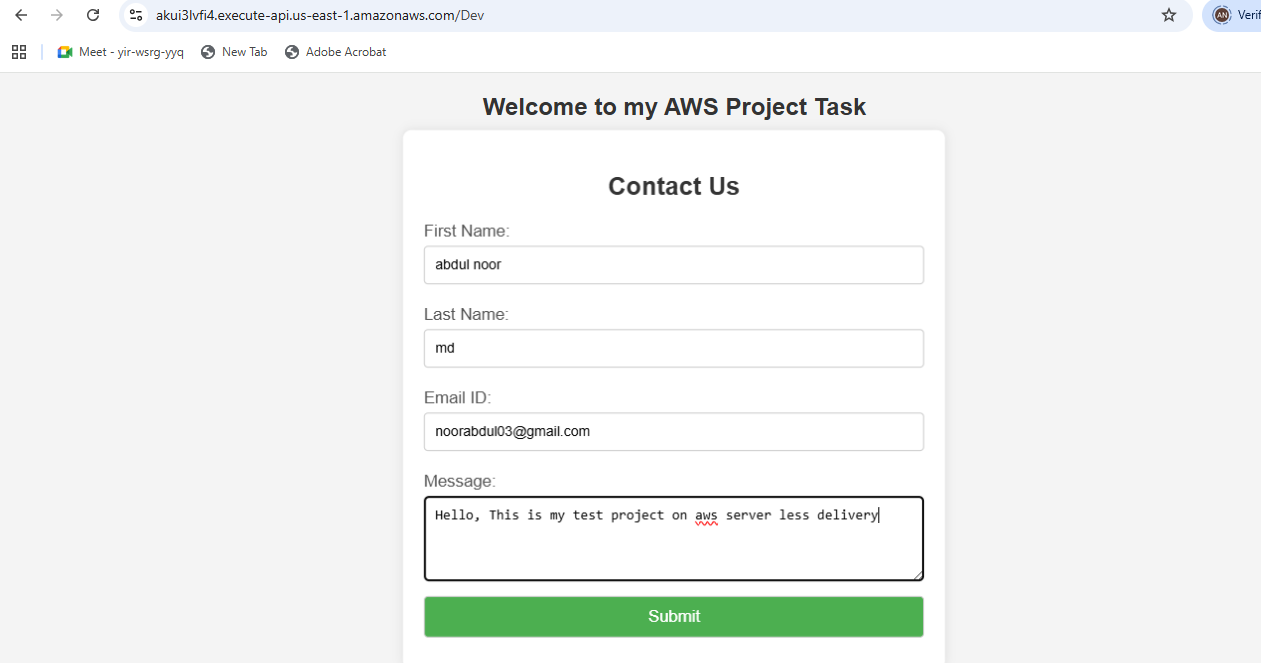
Whenever you Deploy API you are going to get an invoke URL. Copy this Invoke (Raise) URL

<https://akui3lvfi4.execute-api.us-east-1.amazonaws.com/Dev>

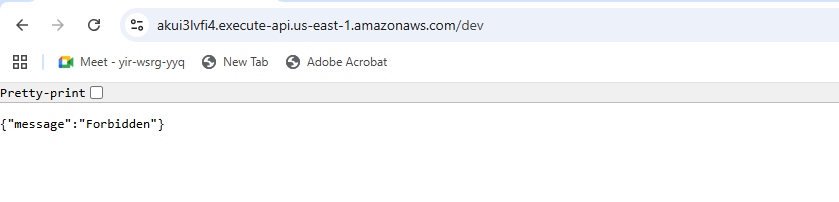
\*\*\*Now whenever you are giving this invoke url in browser, it is loading this webpage. This webpage which is in Lambda function.



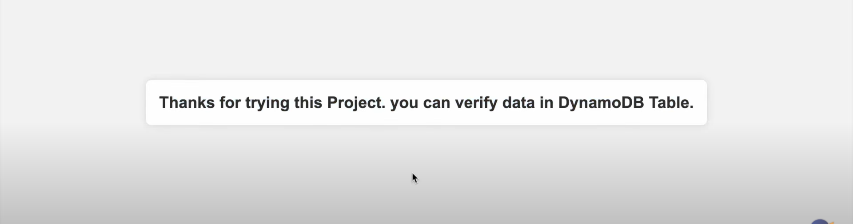
Now fill some details



Then click on submit. Whenever we are submitting this, then storing in backend dynamodb table, and it’s loading this page. Basically the below is another web page i.e success.html webpage which is in lambda function.



here above it should come as the below page. But coming error.



Now how to verify this

Go to DynamoDb

Go to Explore Items