# **Project Title:** AWS REST API Gateway with Lambda Integration - Mini Project

**Overview:** This mini project demonstrates how to build a serverless REST API using AWS API Gateway and AWS Lambda. The goal is to understand how API Gateway acts as a reverse proxy and integrates with AWS Lambda to expose backend functionalities securely and at scale.

**Objective:** To create and test a RESTful API with GET and POST methods connected to AWS Lambda functions using Postman.

---

**Technologies Used:**

- AWS API Gateway (REST API)
- AWS Lambda (Python runtime)
- Postman (API Testing Tool)

---

**Why AWS API Gateway?** AWS API Gateway is a fully managed service that allows developers to create, publish, maintain, monitor, and secure APIs at any scale. It acts as a bridge between frontend applications and backend services like Lambda, EC2, DynamoDB, etc.

**Types of APIs Supported:**

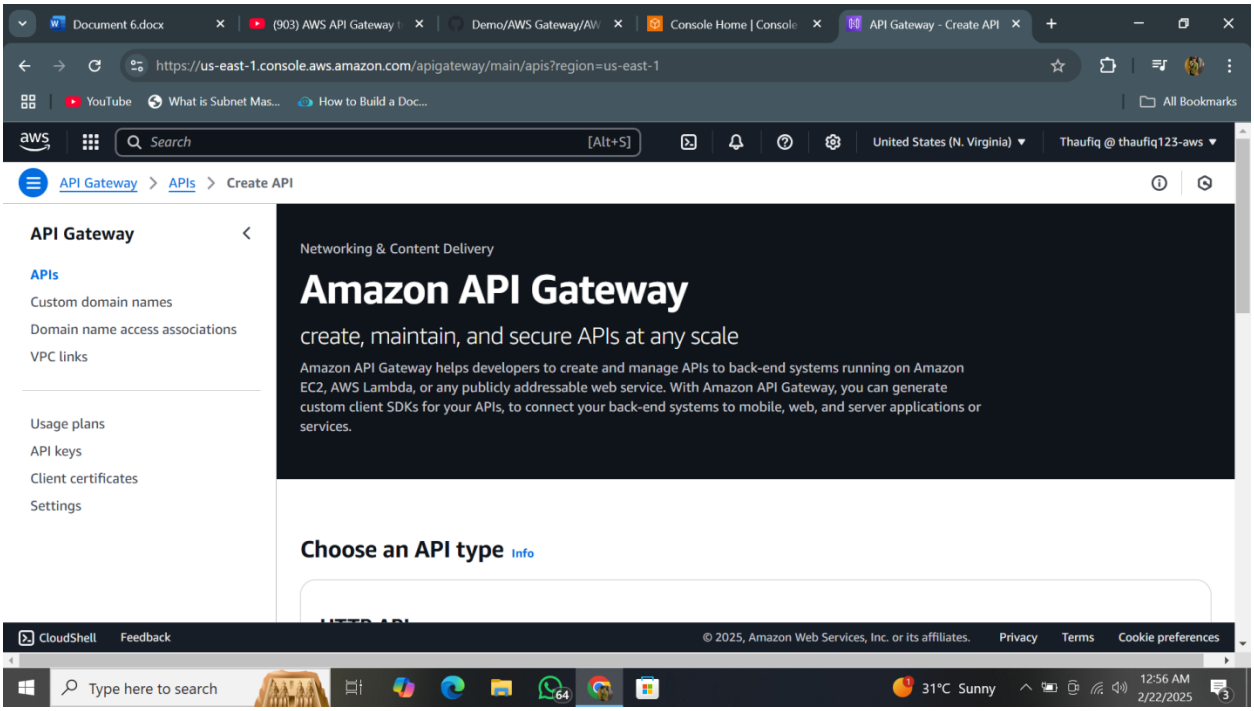
1. **REST API:** Best for standard RESTful services with full AWS integrations.
    - Supports IAM, Cognito, and Lambda authorizers
    - Can integrate with services like Lambda, EC2, DynamoDB
    - Features like throttling, caching, and request validation
2. **HTTP API:** Lightweight and cost-effective for simple APIs.
    - Faster and cheaper than REST APIs
    - Limited features and integrations
3. **WebSocket API:** For real-time, two-way communication.
    - Used for chat apps, stock price updates, etc.

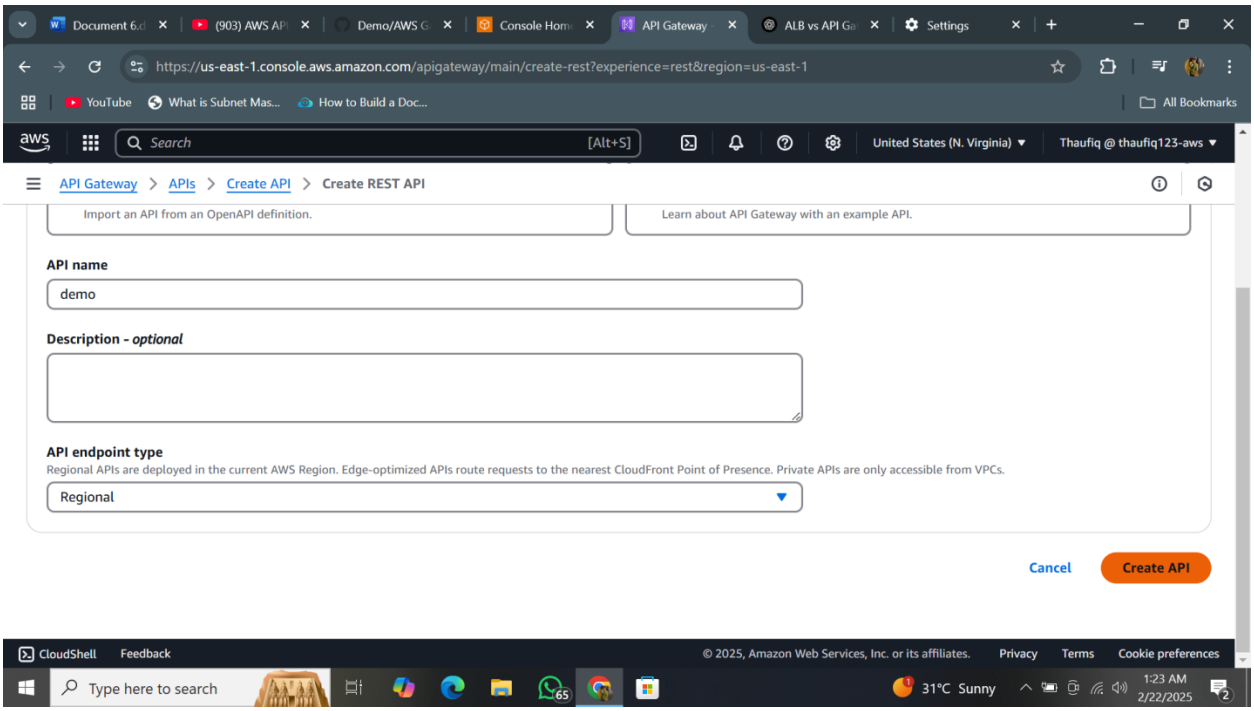
**This project uses the REST API.**

**Step-by-Step Implementation**

## Step 1: Login to AWS Console

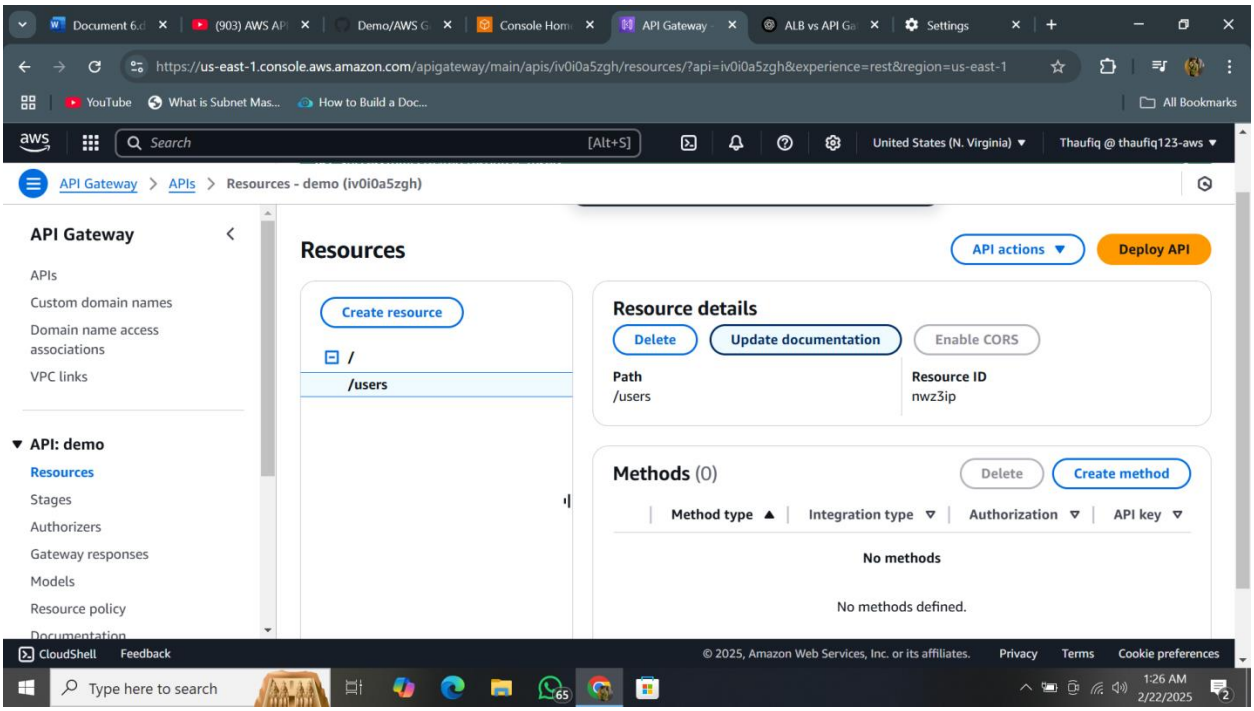
- Navigate to the **API Gateway** service
- Choose **Create API**
- Select **REST API** → Click **Build**
- API Name: `demoapi`
- Endpoint Type: `Regional`





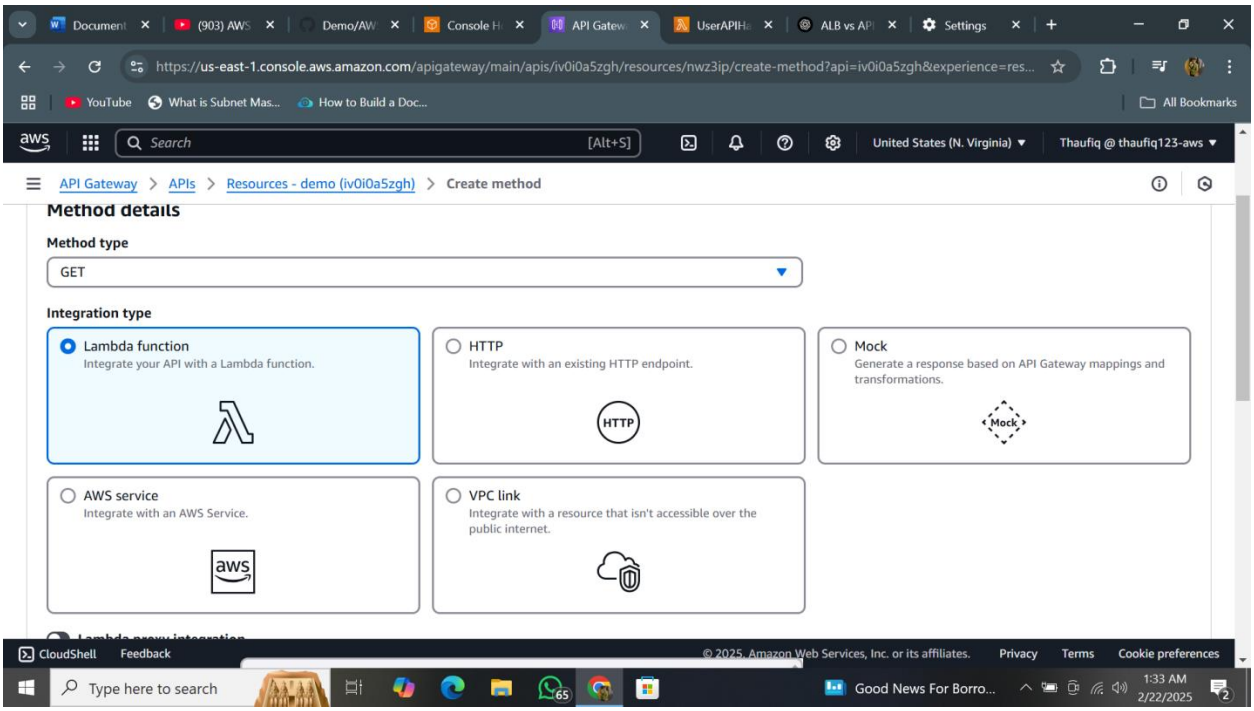

## Create a Resource

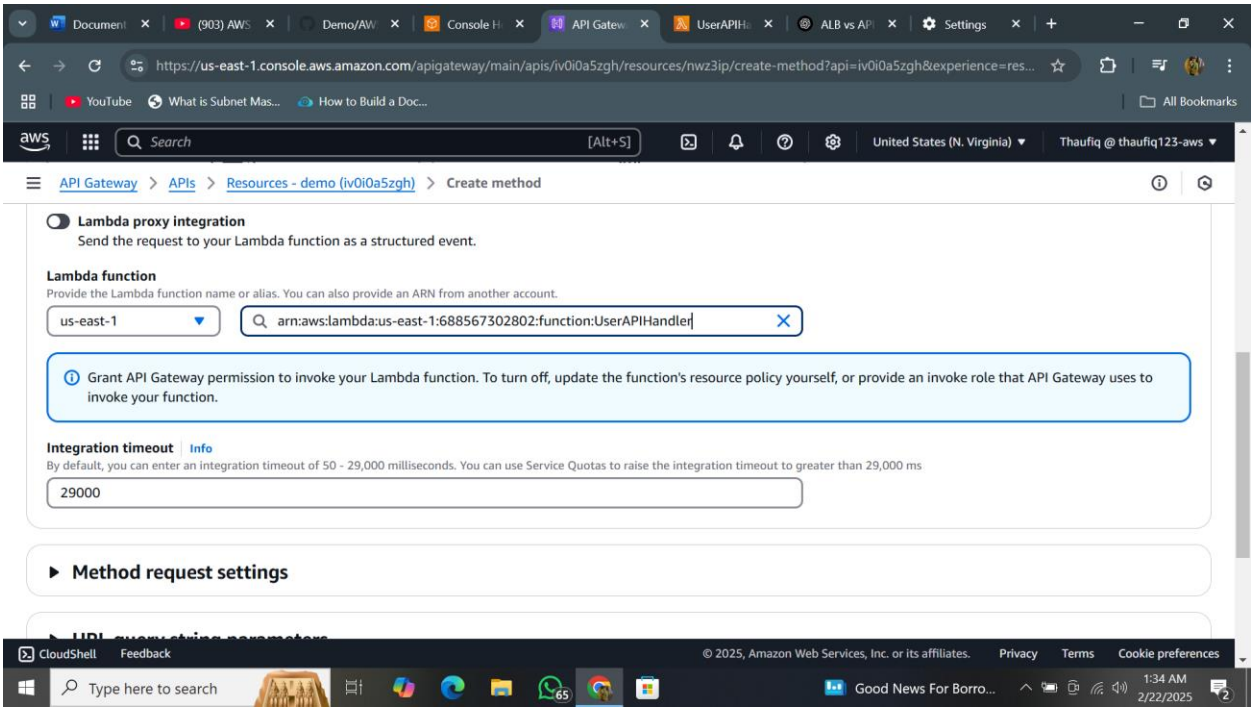
- Click **Actions → Create Resource**.
- Resource name: `users`.
- Resource Path: `/users`.
- Click **Create Resource**.




## Create a Method (GET & POST)

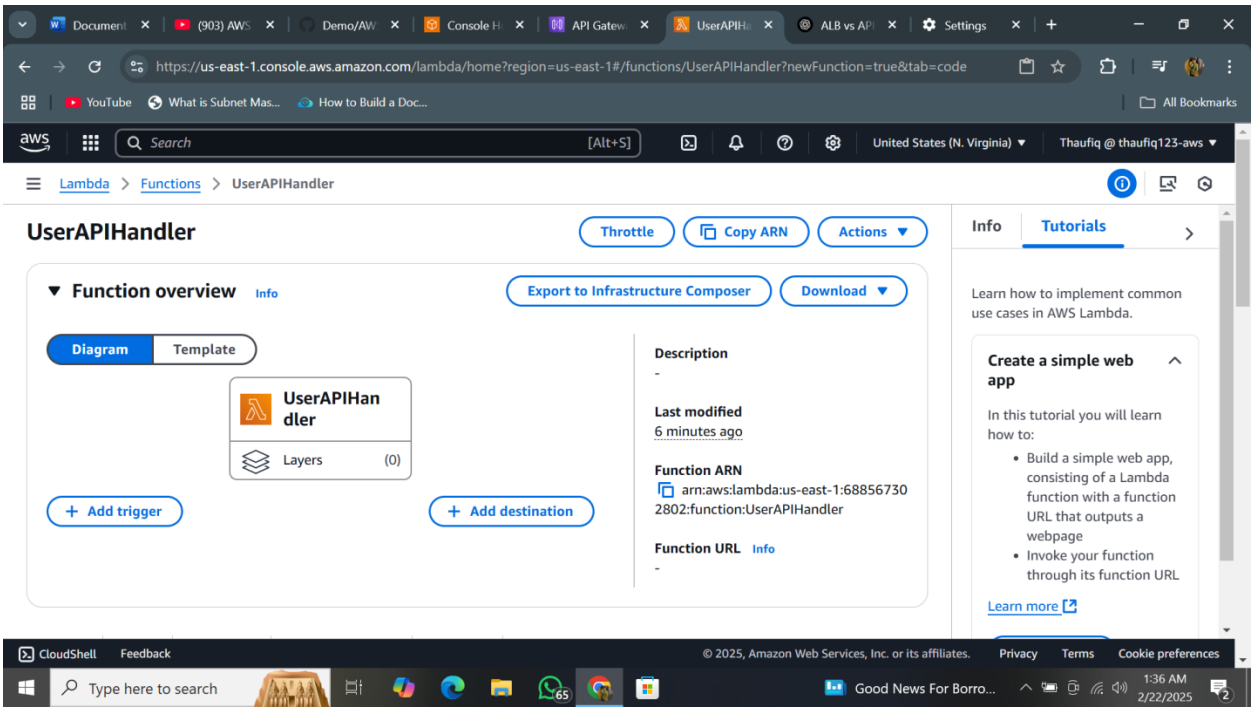
- Select `/users` → Click **Actions** → **Create Method**.
- Choose **GET** → Click the checkmark .
- Integration type: **Lambda Function**.
- Enter Lambda function name: `UserAPIHandler`.
- Click **Save** → **OK**.

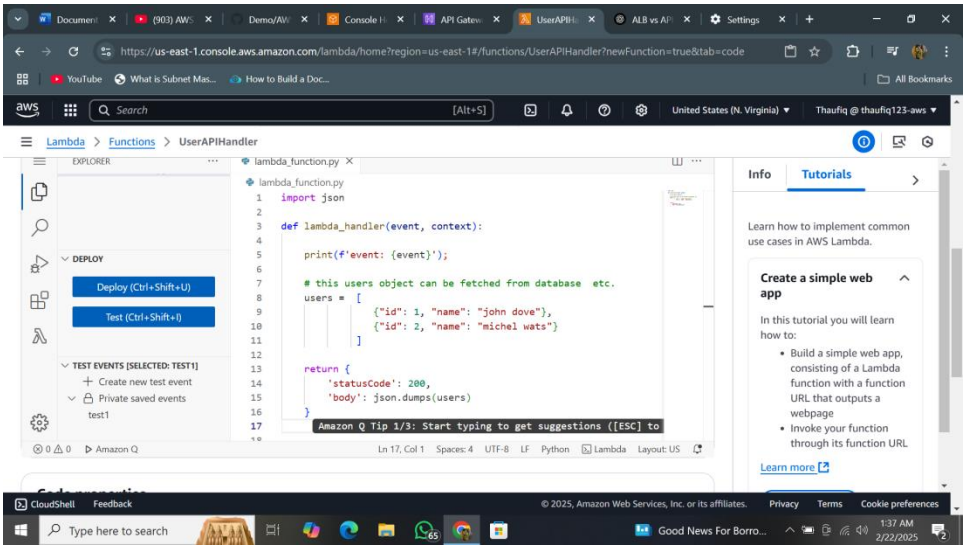



Click create .

And this my lambada function i create for get users in python language

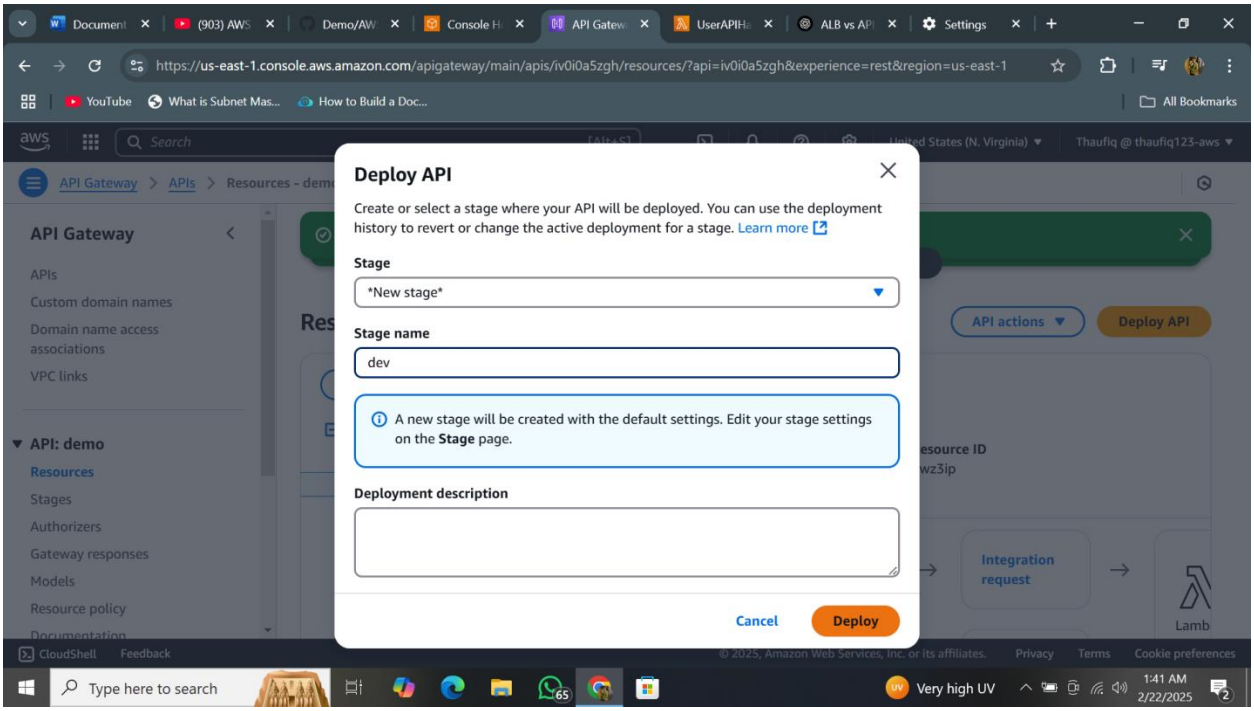


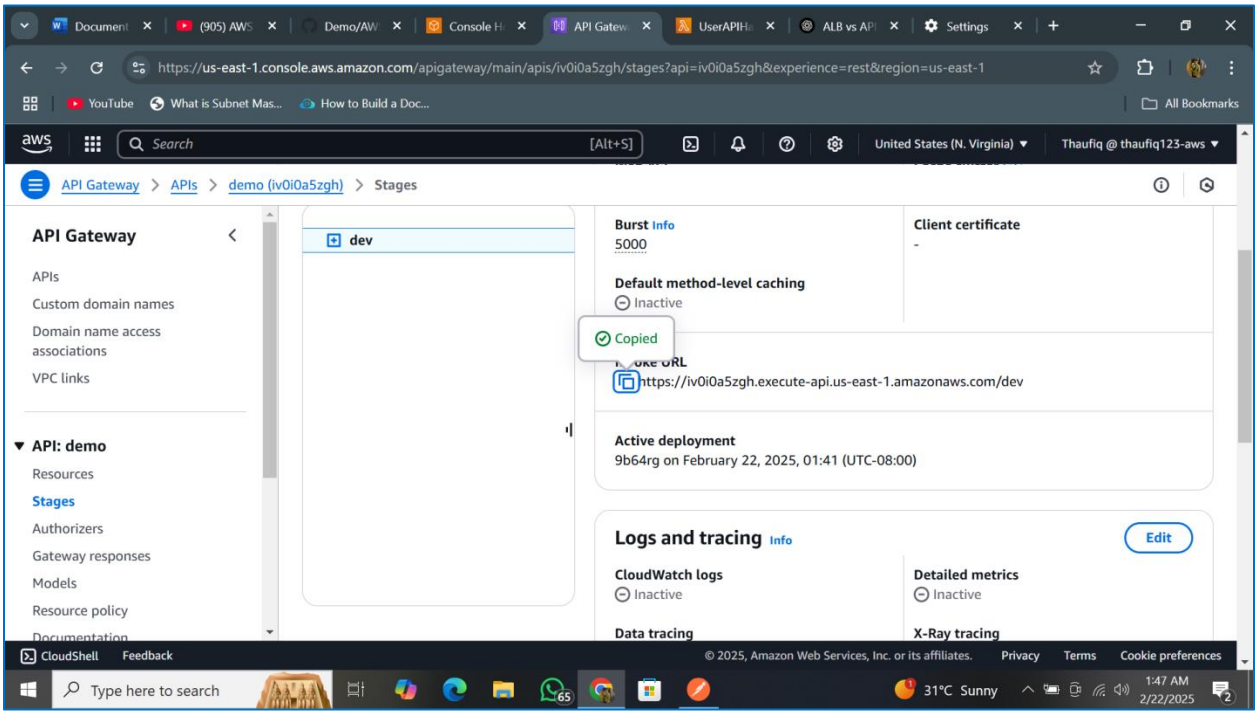




# Deploy the API

- Click **Actions → Deploy API**.

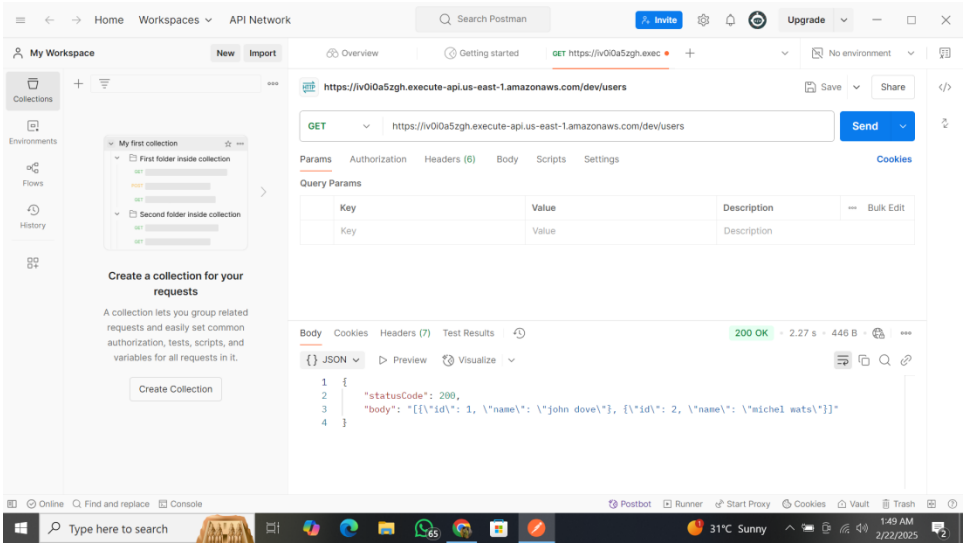
- Stage name: dev.
- Click deploy



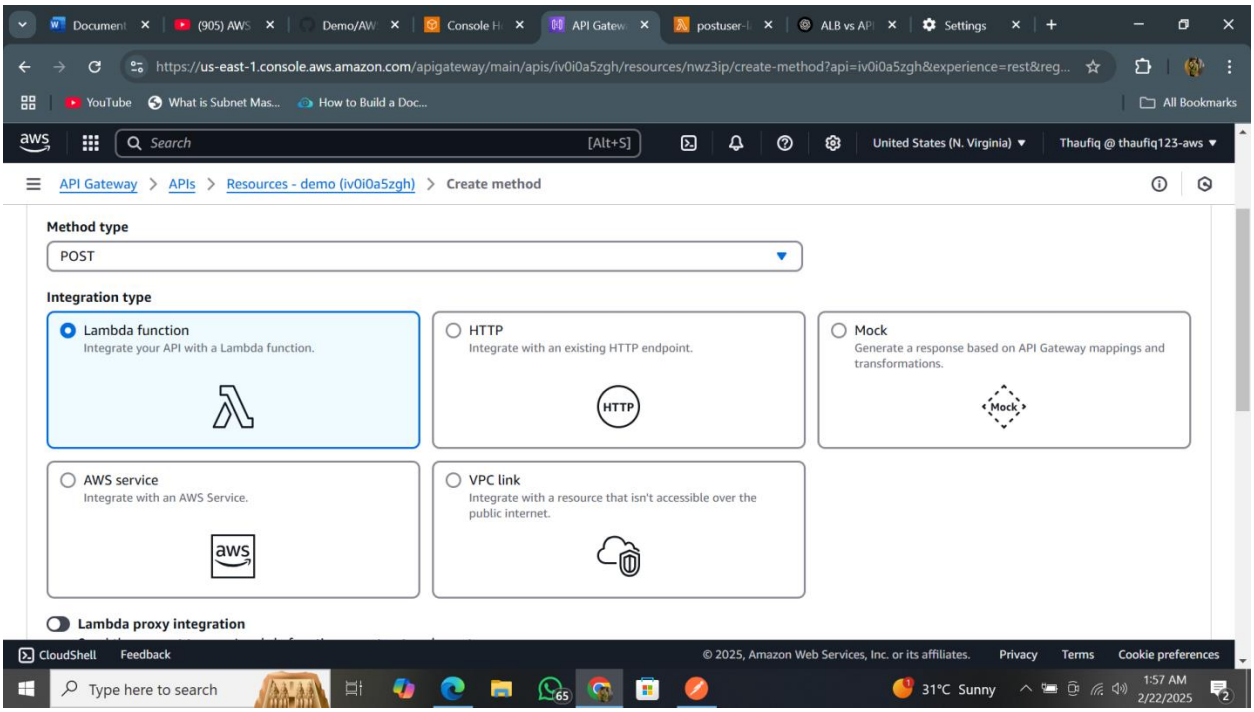

Then you get link i my is this [https://iv0i0a5zgh.execute-api.us-east-1.amazonaws.com/dev](https://iv0i0a5zgh.execute-api.us-east-1.amazonaws.com/dev)

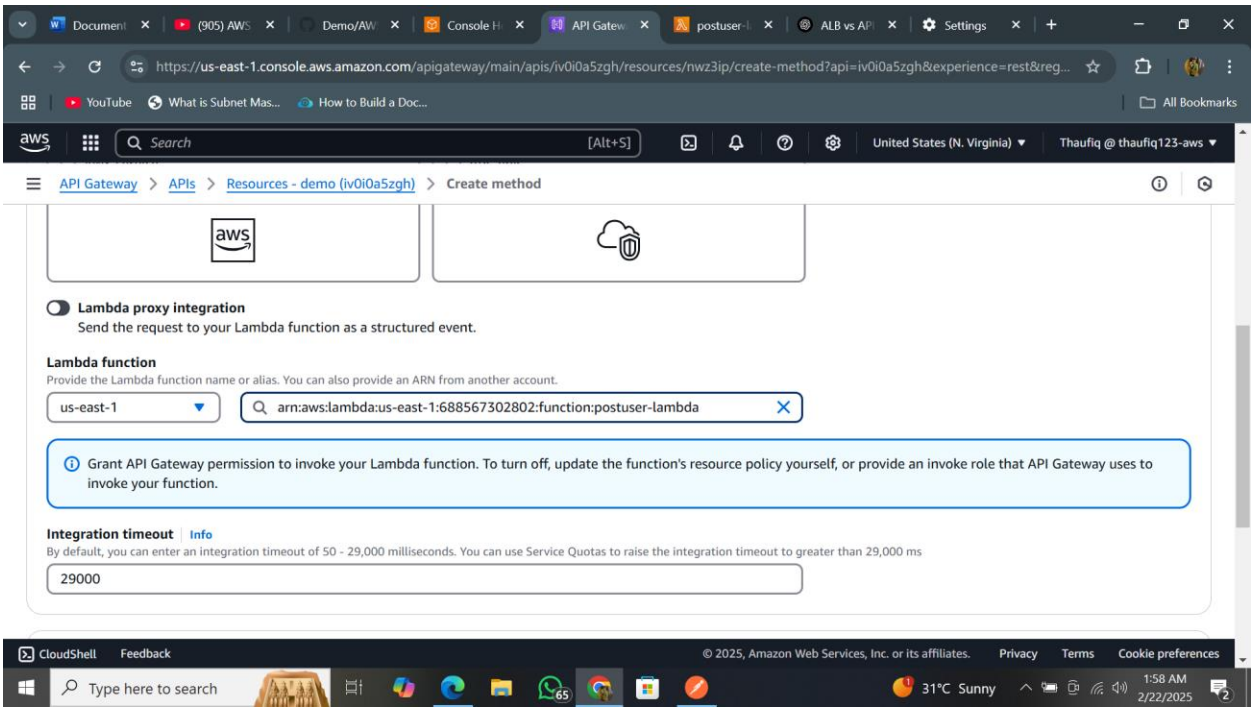
Lets check it in postman it working or not

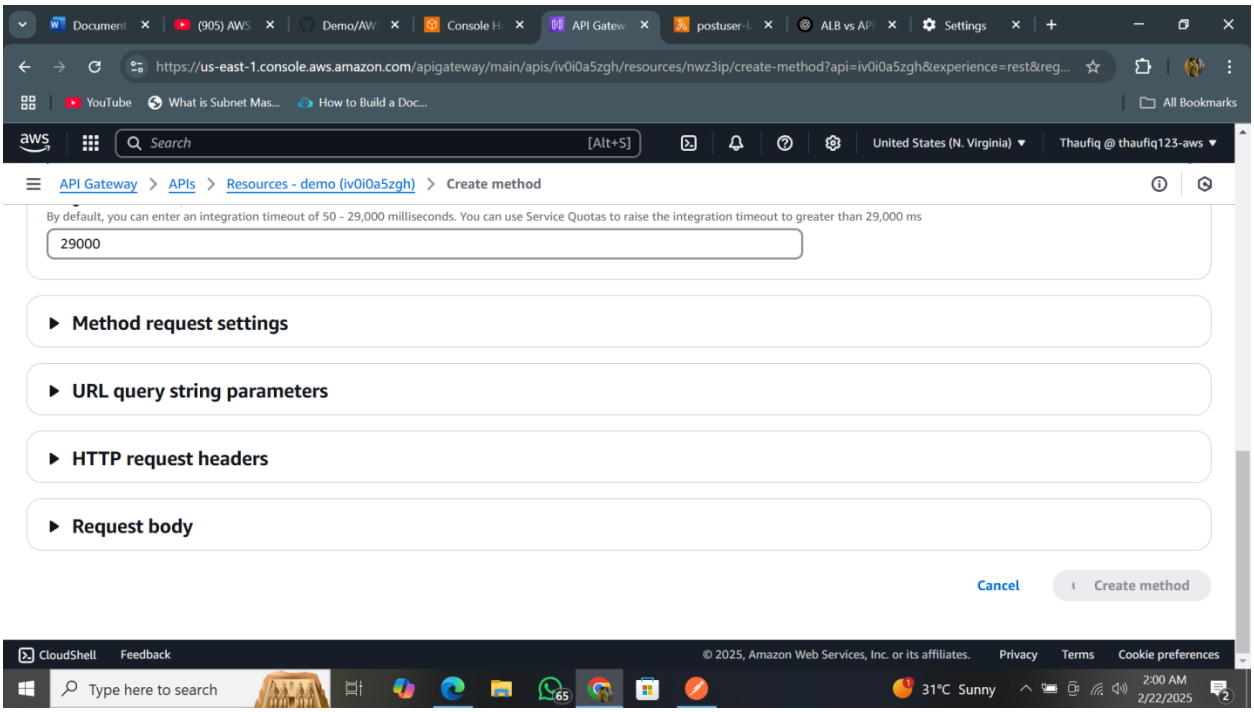


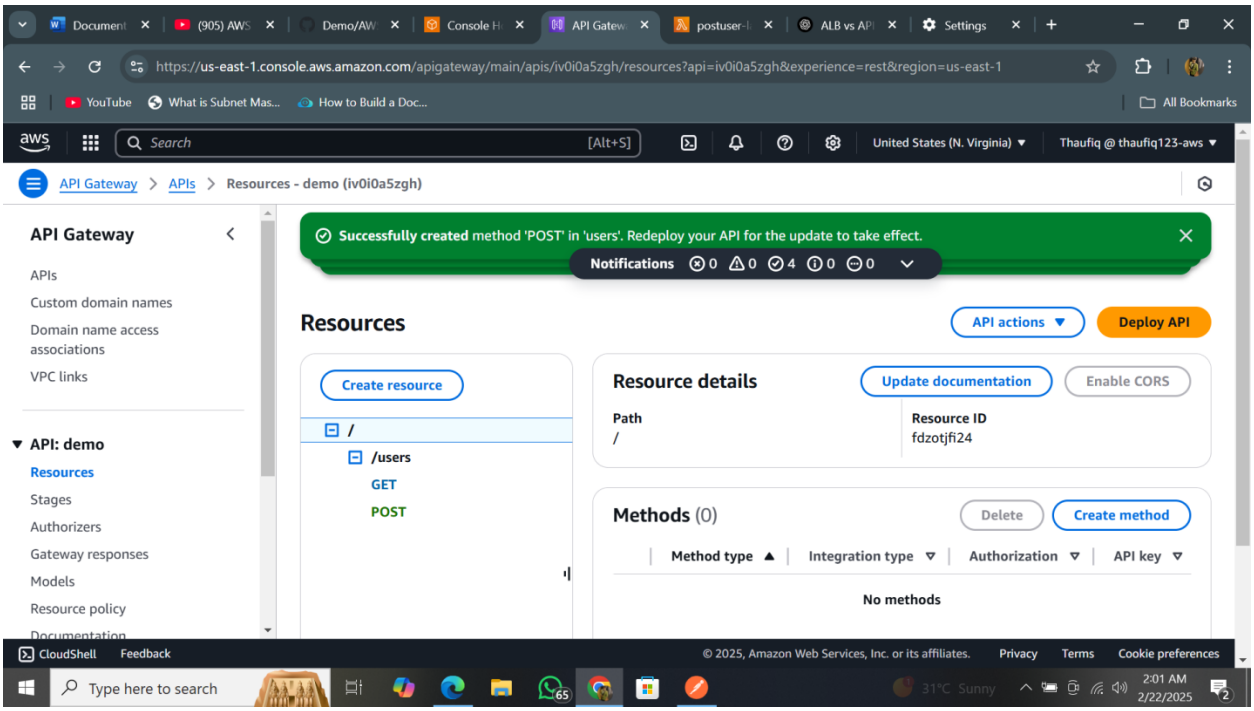

Yes IT WORKING WE GET JSON OUT PUT that we write in lambda code right same for post request also

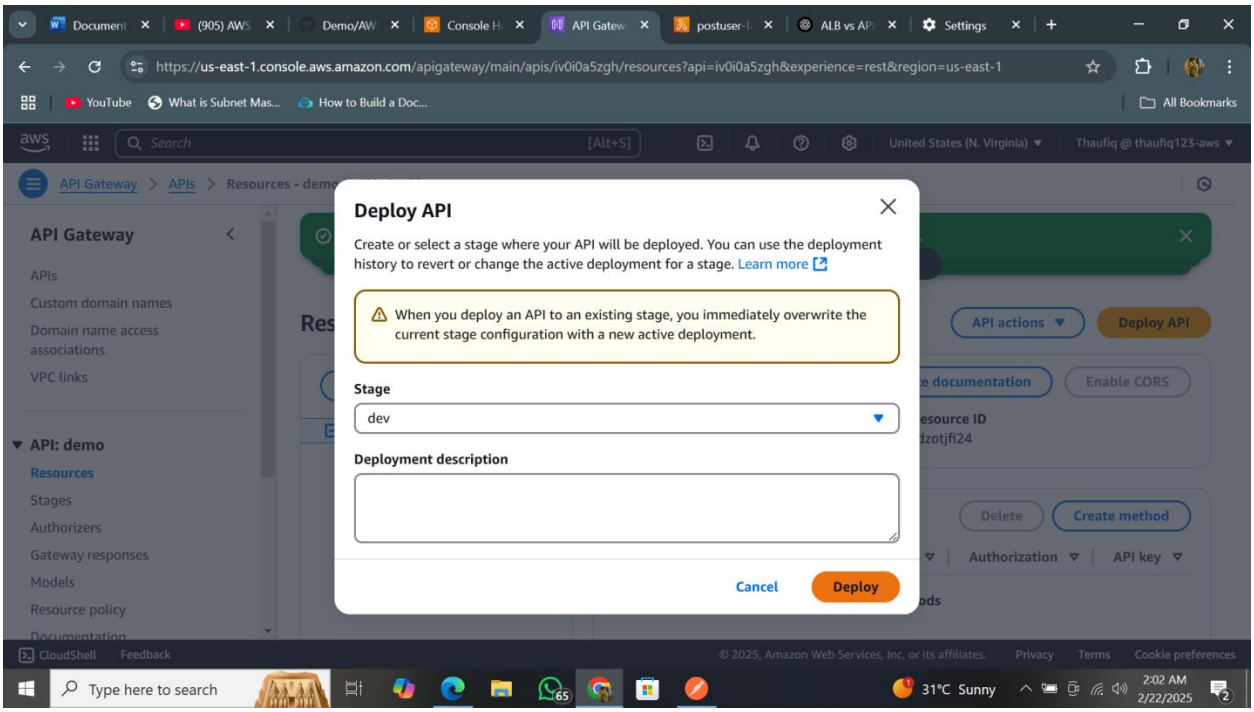
## LEST BEGIN FOR POST REQUEST

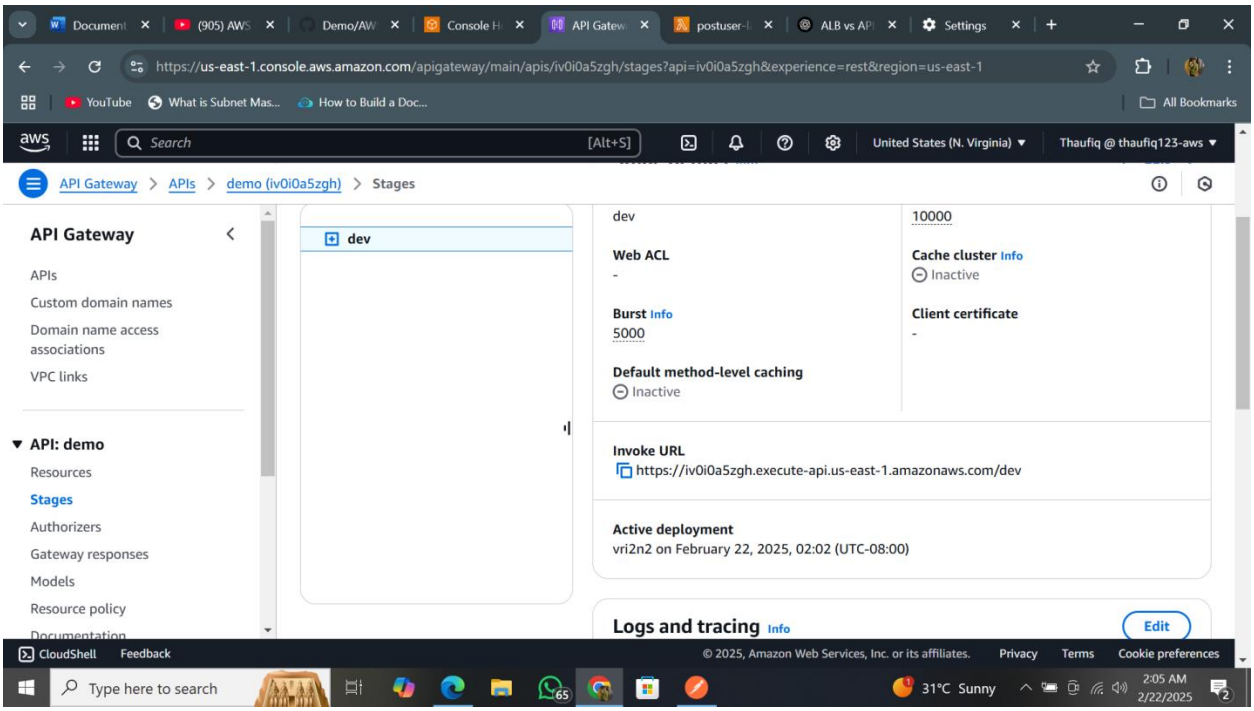


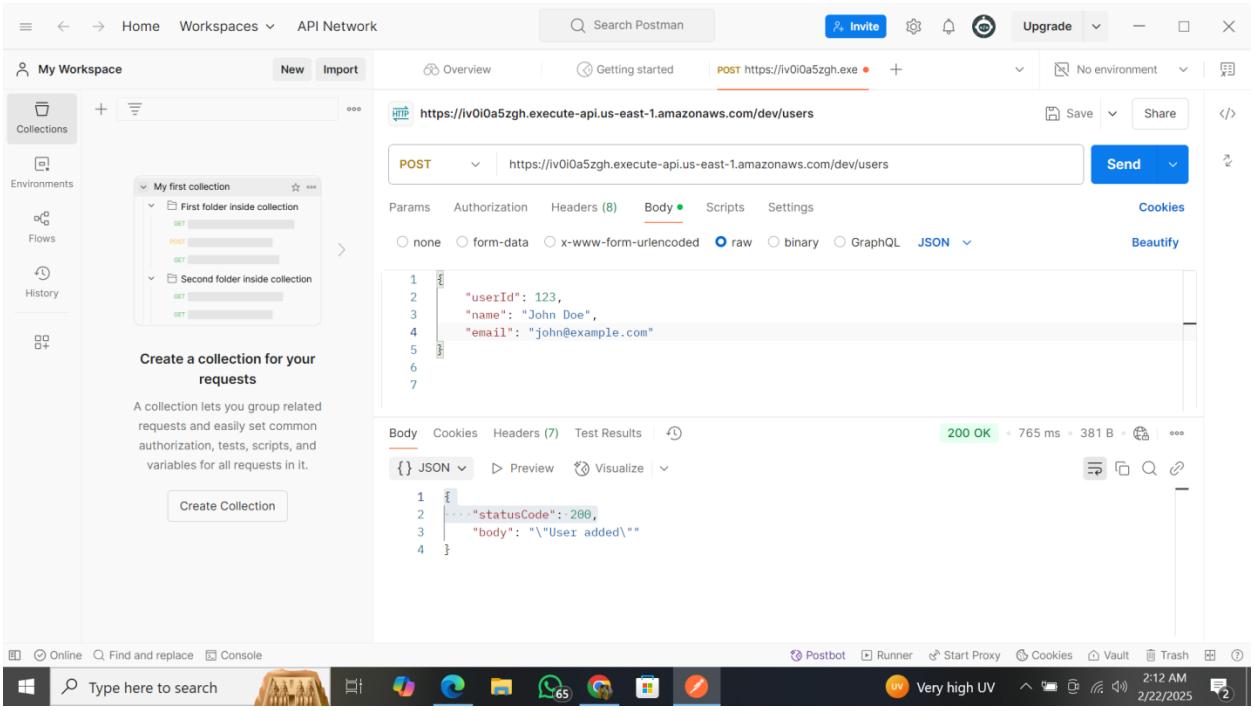

https://us-east-1.console.aws.amazon.com/apigateway/main/apis/iv0i0a5zgh/resources/nwz3ip/create-method?api=iv0i0a5zgh&experience=rest&reg...
YouTube
What is Subnet Mas...
How to Build a Doc...
All Bookmarks
Search
[Alt+S]
United States (N. Virginia)
Thaufiq @ thaufiq123-aws
API Gateway > APIs > Resources - demo (iv0i0a5zgh) > Create method
Lambda proxy integration
Send the request to your Lambda function as a structured event.
Lambda function
Provide the Lambda function name or alias. You can also provide an ARN from another account.
us-east-1
arn:aws:lambda:us-east-1:688567302802:function:postuser-lambda
Grant API Gateway permission to invoke your Lambda function. To turn off, update the function's resource policy yourself, or provide an invoke role that API Gateway uses to invoke your function.
Integration timeout
Info
By default, you can enter an integration timeout of 50 - 29,000 milliseconds. You can use Service Quotas to raise the integration timeout to greater than 29,000 ms
29000
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
31°C Sunny
1:58 AM
2/22/2025


https://us-east-1.console.aws.amazon.com/apigateway/main/apis/iv0i0a5zgh/resources/nwz3ip/create-method?api=iv0i0a5zgh&experience=rest&reg...
YouTube
What is Subnet Mas...
How to Build a Doc...
All Bookmarks
Search
[Alt+S]
United States (N. Virginia)
Thaufiq @ thaufiq123-aws
API Gateway > APIs > Resources - demo (iv0i0a5zgh) > Create method
By default, you can enter an integration timeout of 50 - 29,000 milliseconds. You can use Service Quotas to raise the integration timeout to greater than 29,000 ms
29000
Method request settings
URL query string parameters
HTTP request headers
Request body
Cancel
Create method
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
31°C Sunny
2:00 AM
2/22/2025


https://us-east-1.console.aws.amazon.com/apigateway/main/apis/iv0i0a5zgh/resources?api=iv0i0a5zgh&experience=rest&region=us-east-1
API Gateway > APIs > Resources - demo (iv0i0a5zgh)
API Gateway
APIs
Custom domain names
Domain name access associations
VPC links
API: demo
Resources
Stages
Authorizers
Gateway responses
Models
Resource policy
Successfully created method 'POST' in 'users'. Redeploy your API for the update to take effect.
Notifications
Resources
API actions
Deploy API
Create resource
/
/users
GET
POST
Resource details
Update documentation
Enable CORS
Path
/
Resource ID
fdzotjfi24
Methods (0)
Delete
Create method
Method type
Integration type
Authorization
API key
No methods
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
2:01 AM
2/22/2025


Deploy API
Create or select a stage where your API will be deployed. You can use the deployment history to revert or change the active deployment for a stage. Learn more
When you deploy an API to an existing stage, you immediately overwrite the current stage configuration with a new active deployment.
Stage
dev
Deployment description
Cancel
Deploy
2:02 AM
2/22/2025




**Conclusion:** This project showcases a practical implementation of AWS REST API Gateway and Lambda integration. It demonstrates how to set up resources, methods, integrate with backend Lambda functions, and test APIs using Postman

**Future Enhancements:**

- Connect API Gateway to DynamoDB for storing users
- Add authentication using AWS Cognito
- Implement validation and error handling in Lambda functions