Experiment No: 03

Working with Amazon DynamoDB

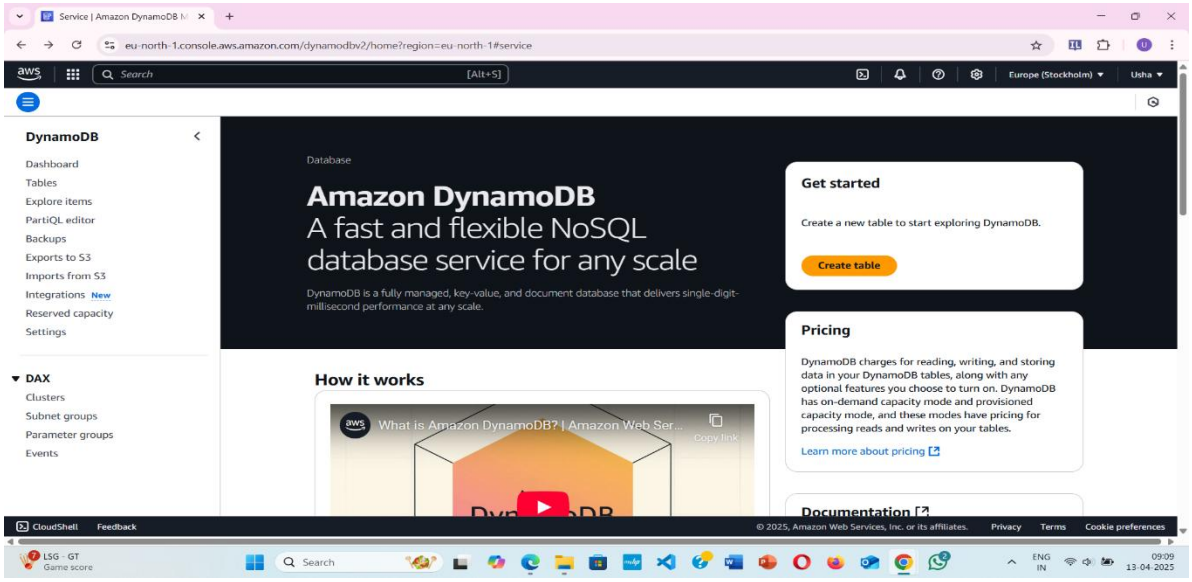
Apparatus Required: AWS Account, AWS Dynamo DB

Pre-Requisite: Aws basics, DBMS basics

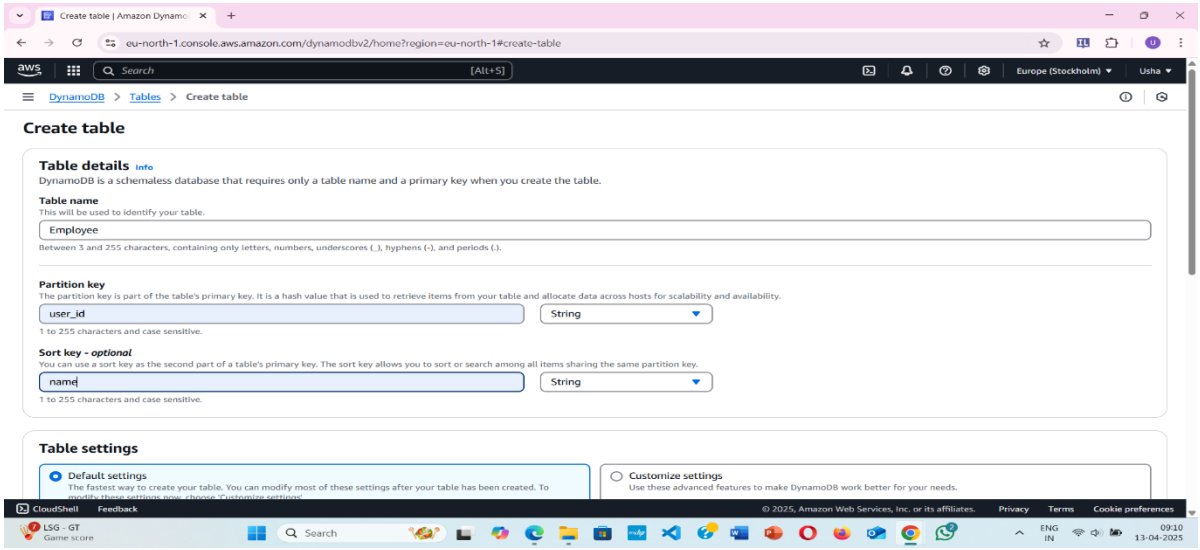
Introduction: AWS DynamoDB is a fast and flexible NoSQL database service for all applications that need consistent, single-digit-millisecond latency at any scale. Its flexible data model and reliable performance make DynamoDB a great fit for mobile, web, gaming, advertising technology, Internet of Things, and other applications.

Procedure

Navigate to Amazon DynamoDB→click on create table



Give table name, Partition Key (is a primary key)



Successfully created the table

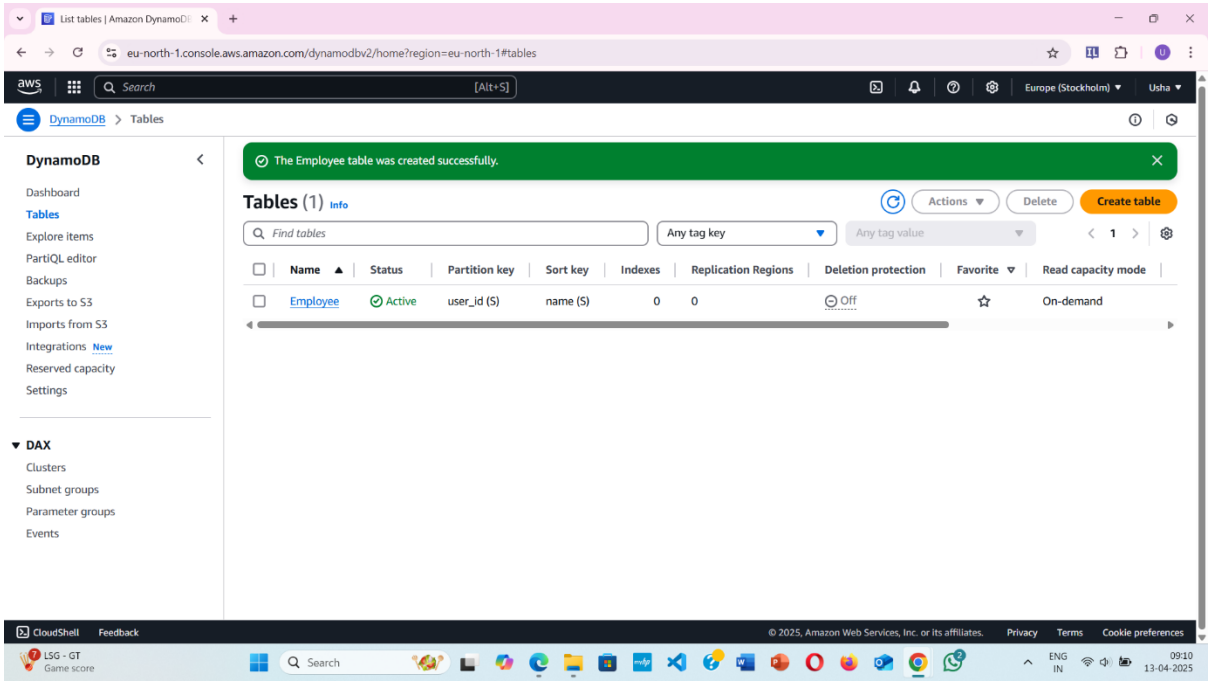
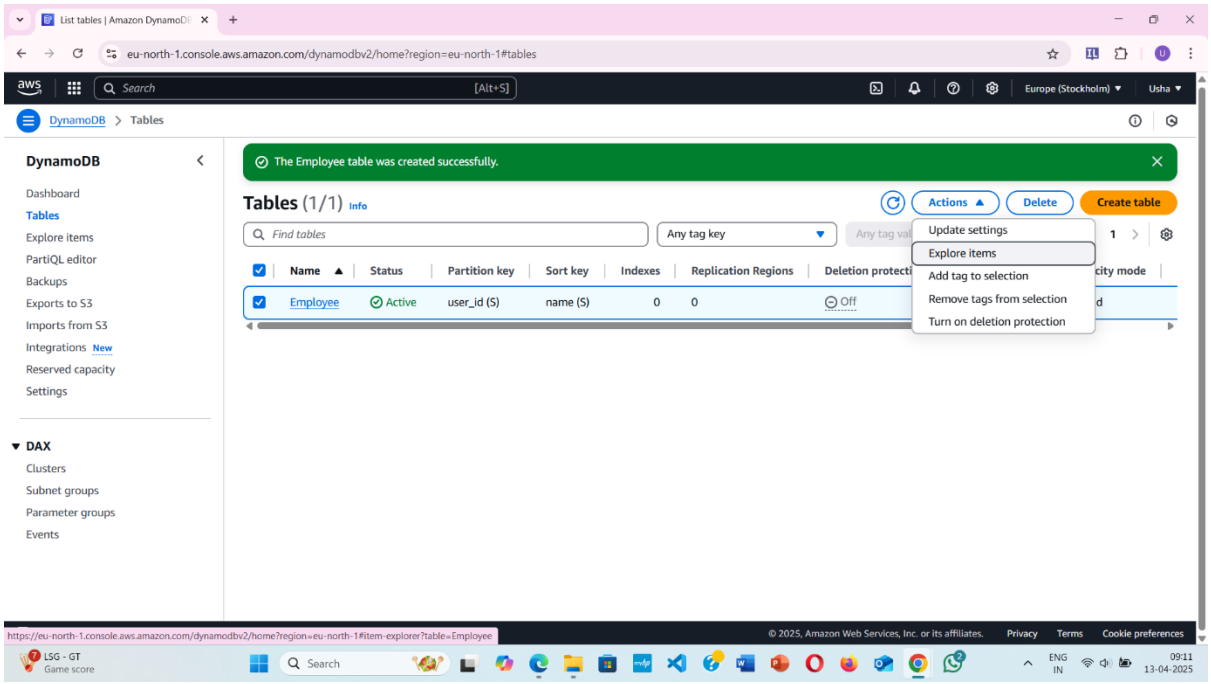
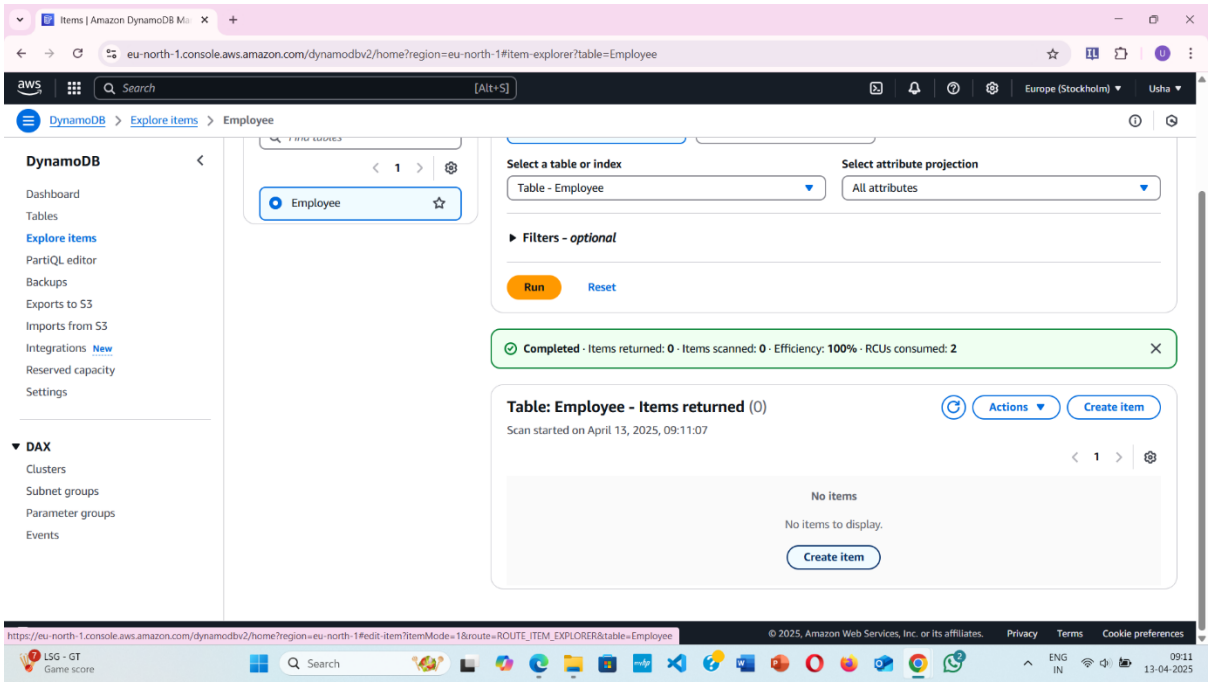
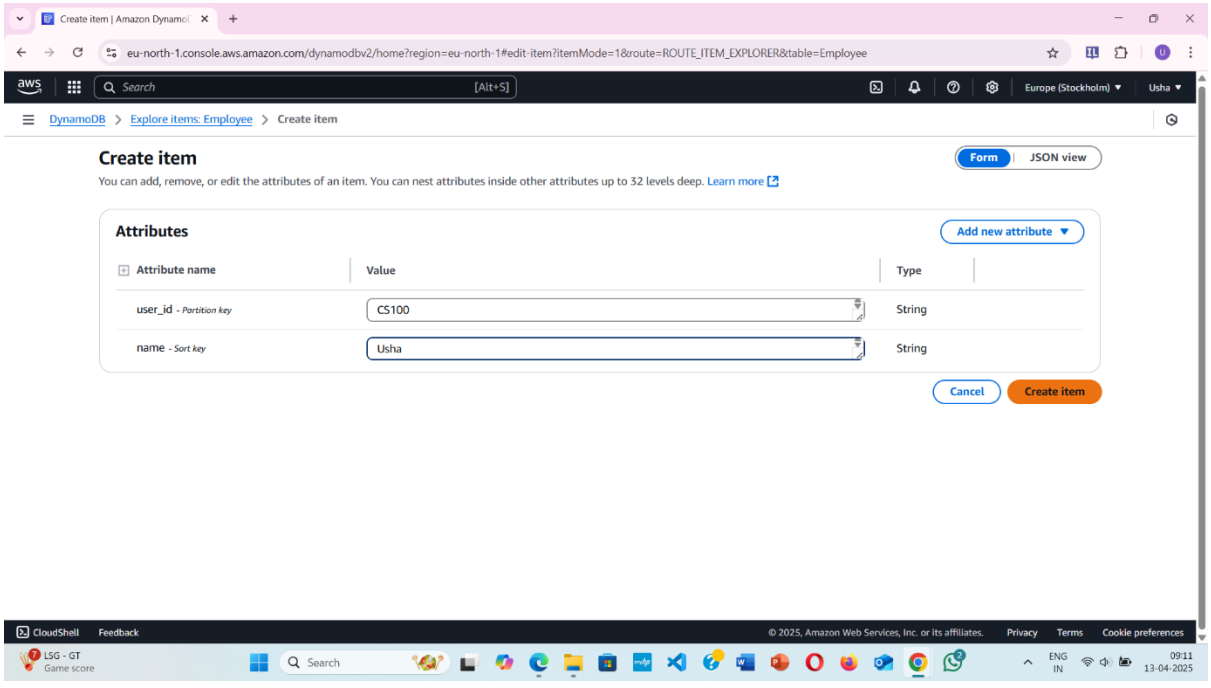
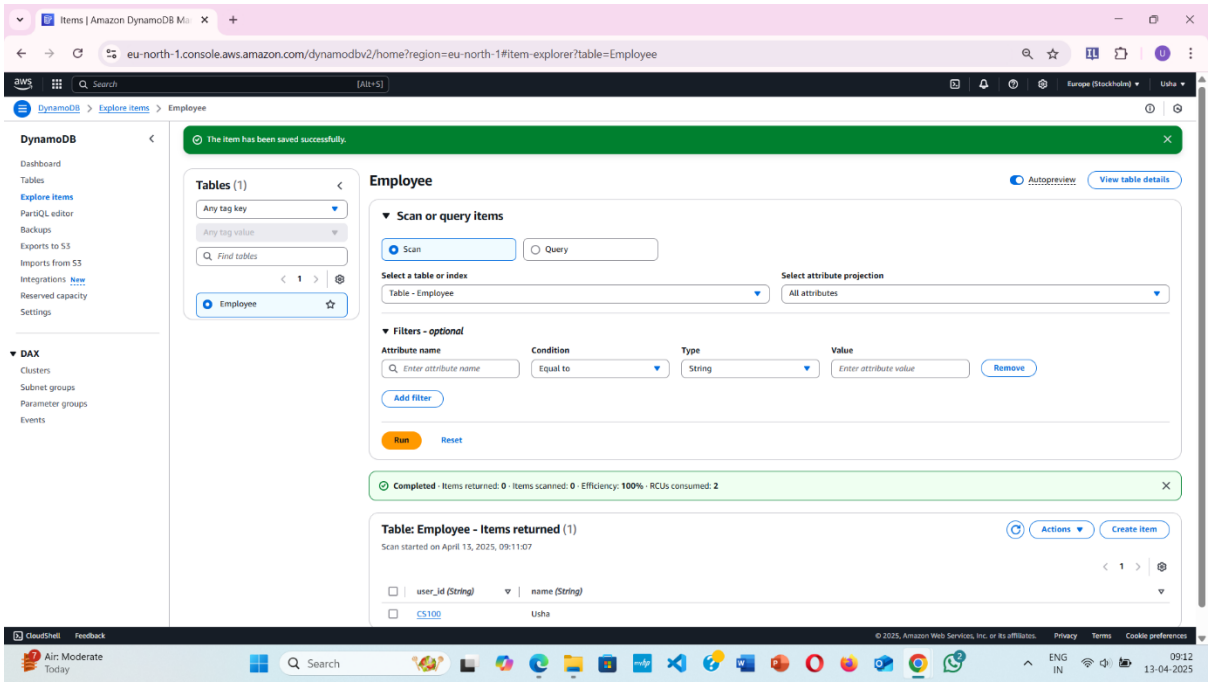


Table created→ select your table→ explore table items→ create items

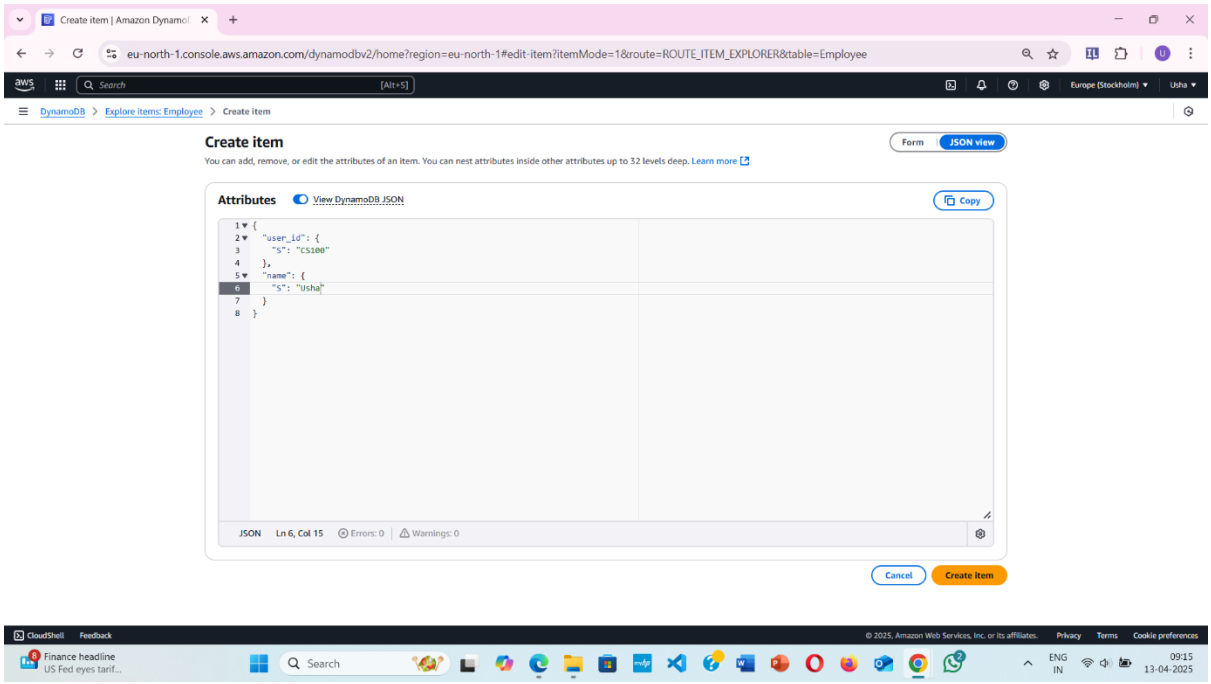


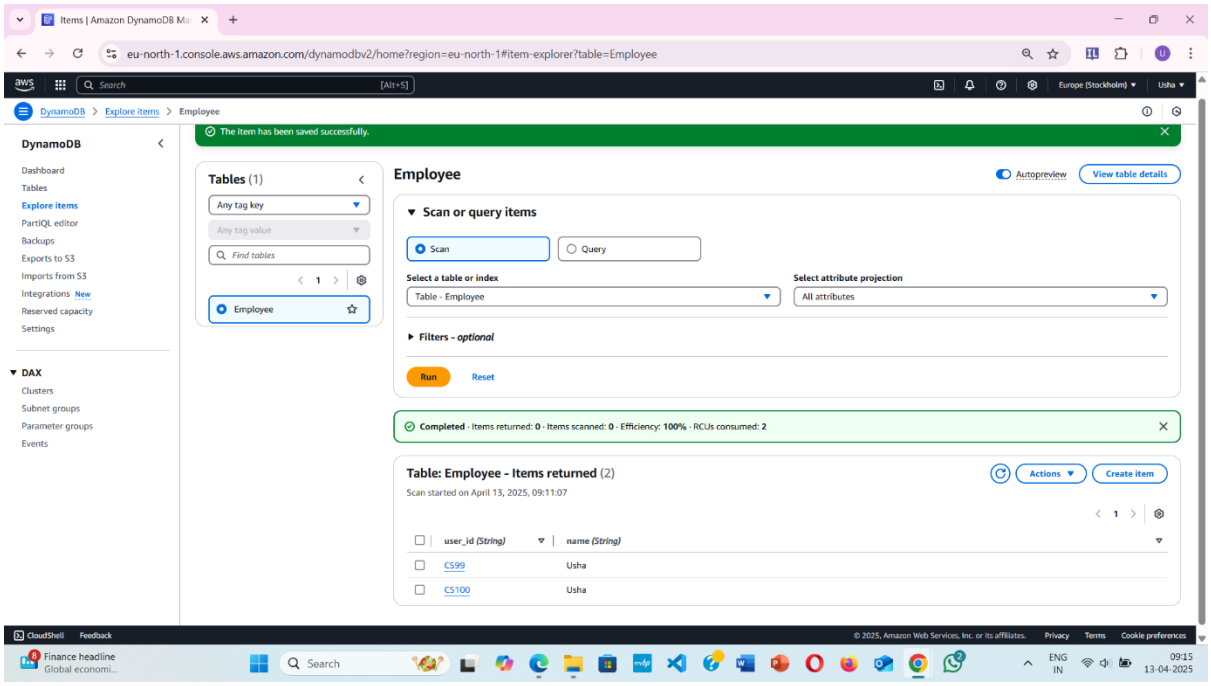

Items | Amazon DynamoDB
eu-north-1.console.aws.amazon.com/dynamodbv2/home?region=eu-north-1#item-explorer?table=Employee
Search [Alt+S]
Europe (Stockholm)
Usha
DynamoDB > Explore items > Employee
DynamoDB
Dashboard
Tables
Explore items
PartiQL editor
Backups
Exports to S3
Imports from S3
Integrations New
Reserved capacity
Settings
DAX
Clusters
Subnet groups
Parameter groups
Events
Employee
Select a table or index
Table - Employee
Select attribute projection
All attributes
Filters - optional
Run
Reset
Completed · Items returned: 0 · Items scanned: 0 · Efficiency: 100% · RCUs consumed: 2
Table: Employee - Items returned (0)
Scan started on April 13, 2025, 09:11:07
Actions
Create item
No items
No items to display.
Create item
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Create item | Amazon DynamoDB
eu-north-1.console.aws.amazon.com/dynamodbv2/home?region=eu-north-1#edit-item?itemMode=1&route=ROUTE_ITEM_EXPLORER&table=Employee
DynamoDB > Explore items: Employee > Create item
Create item
Form
JSON view
You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. Learn more
Attributes
Add new attribute
Attribute name
Value
Type
user_id - Partition key
CS100
String
name - Sort key
Usha
String
Cancel
Create item
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



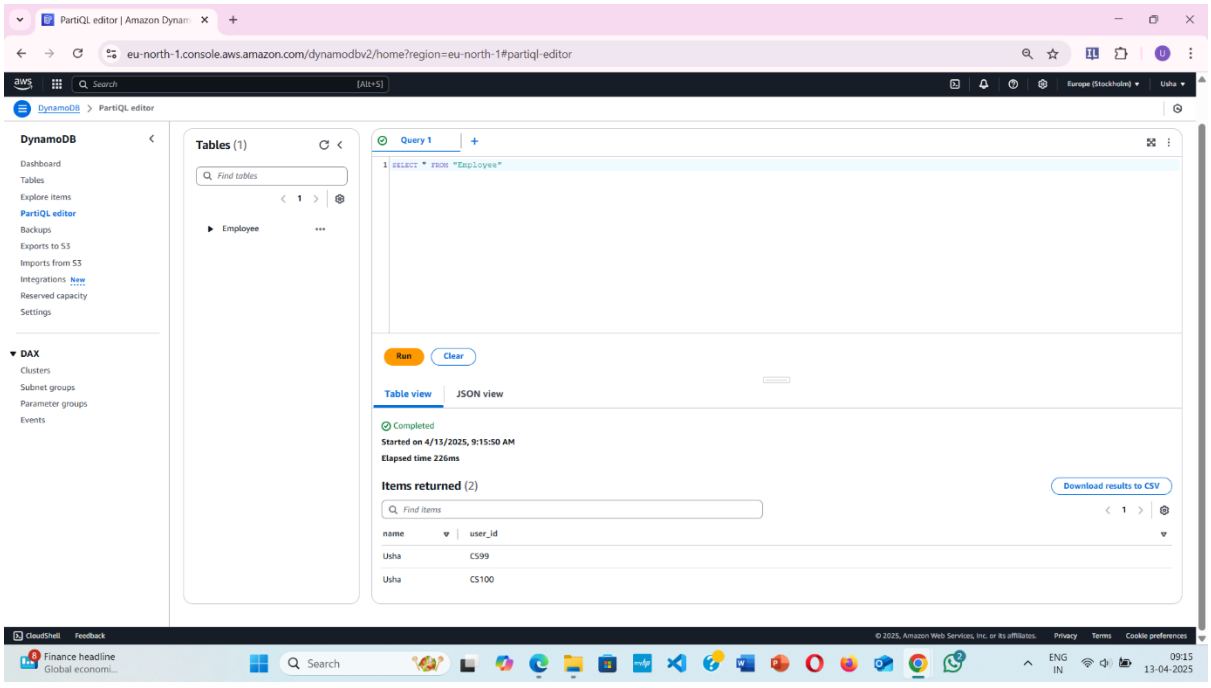
Using JSON to create items



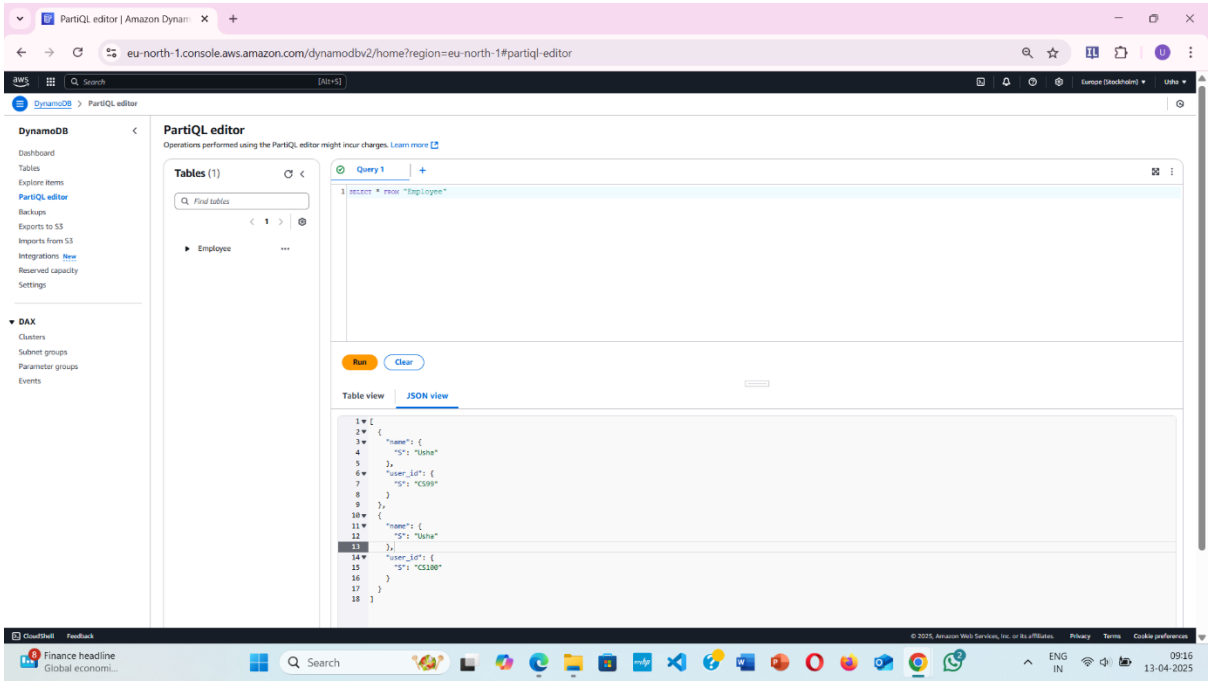


DynamoDB→ PartiQL editor→ write the queries

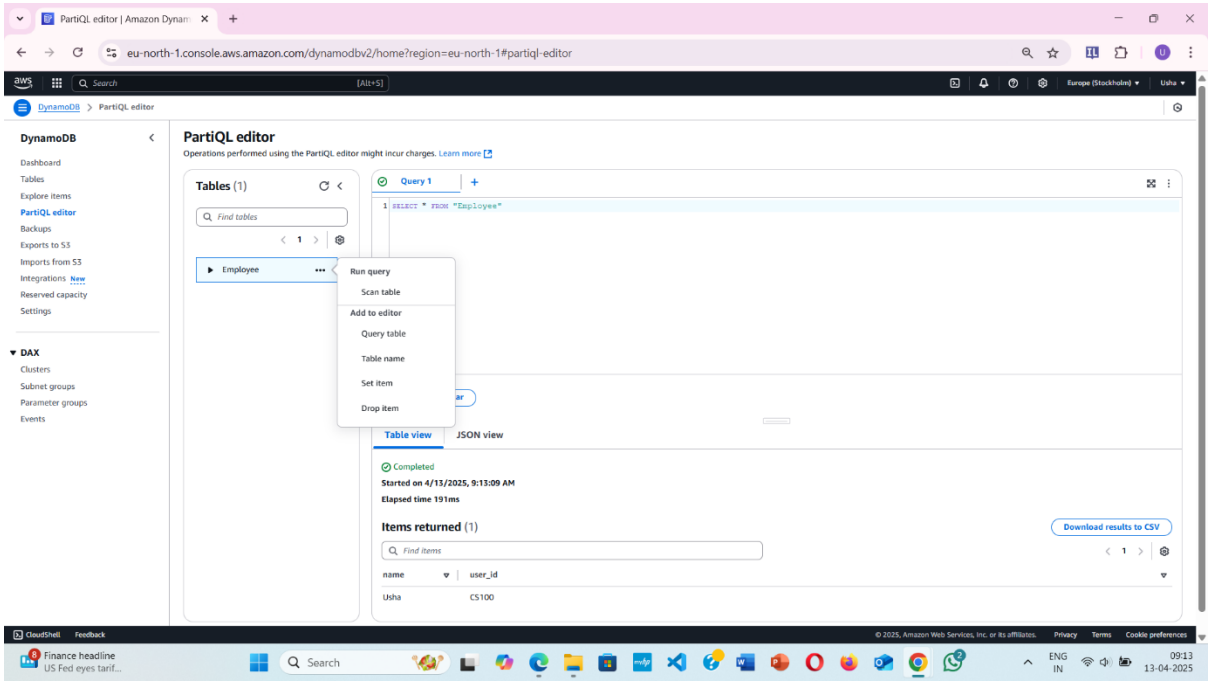
Table view



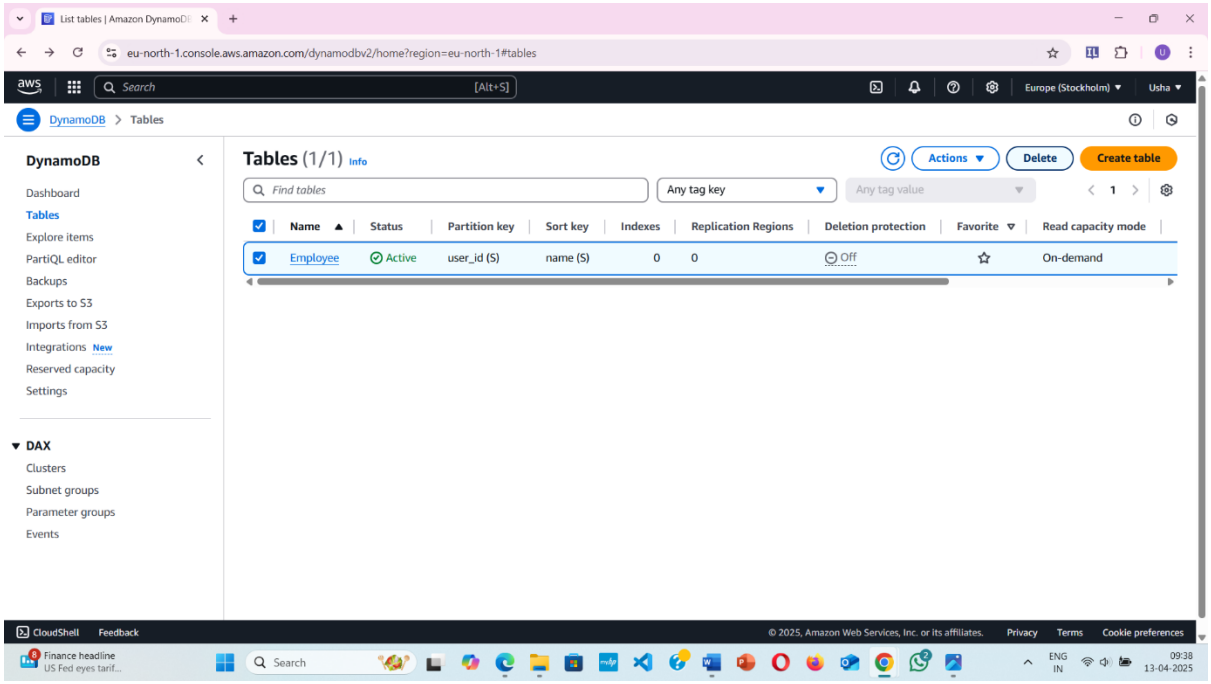
JSON view

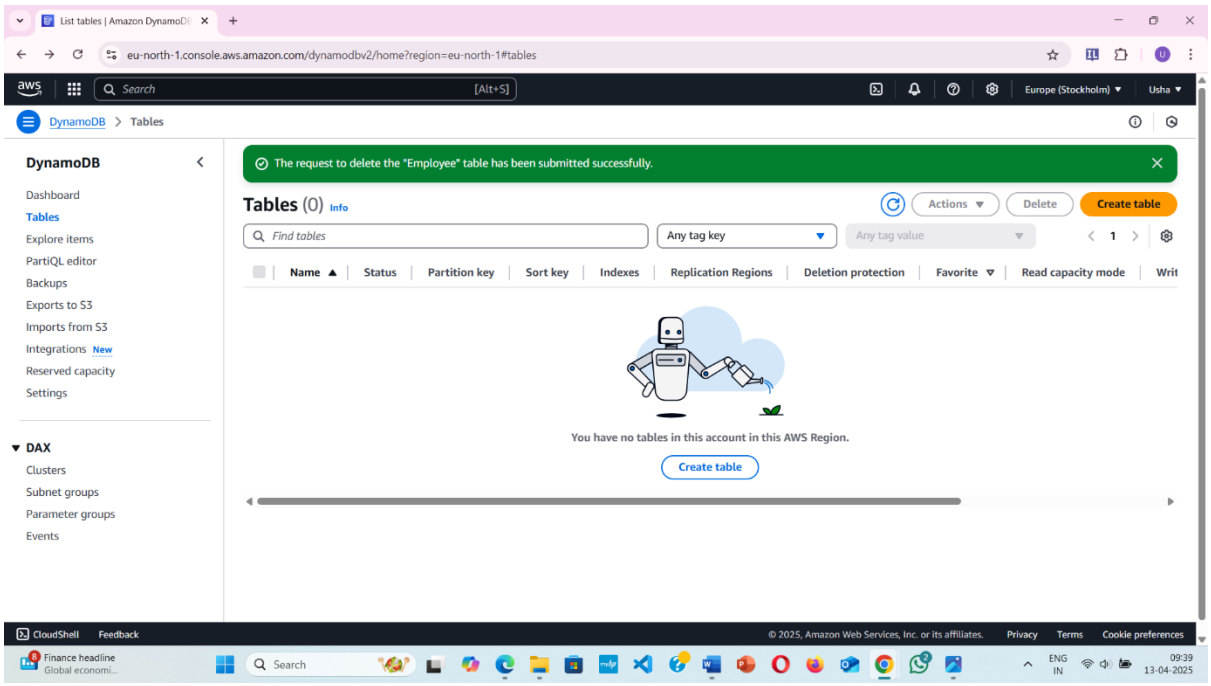


Update Command



Delete table





Observations:

Tables and items are created, and we can run the commands to manipulate the data in the tables.

Experiment No – 04

Developing REST APIs with Amazon API Gateway

Objectives: Developing REST APIs with Amazon API Gateway.

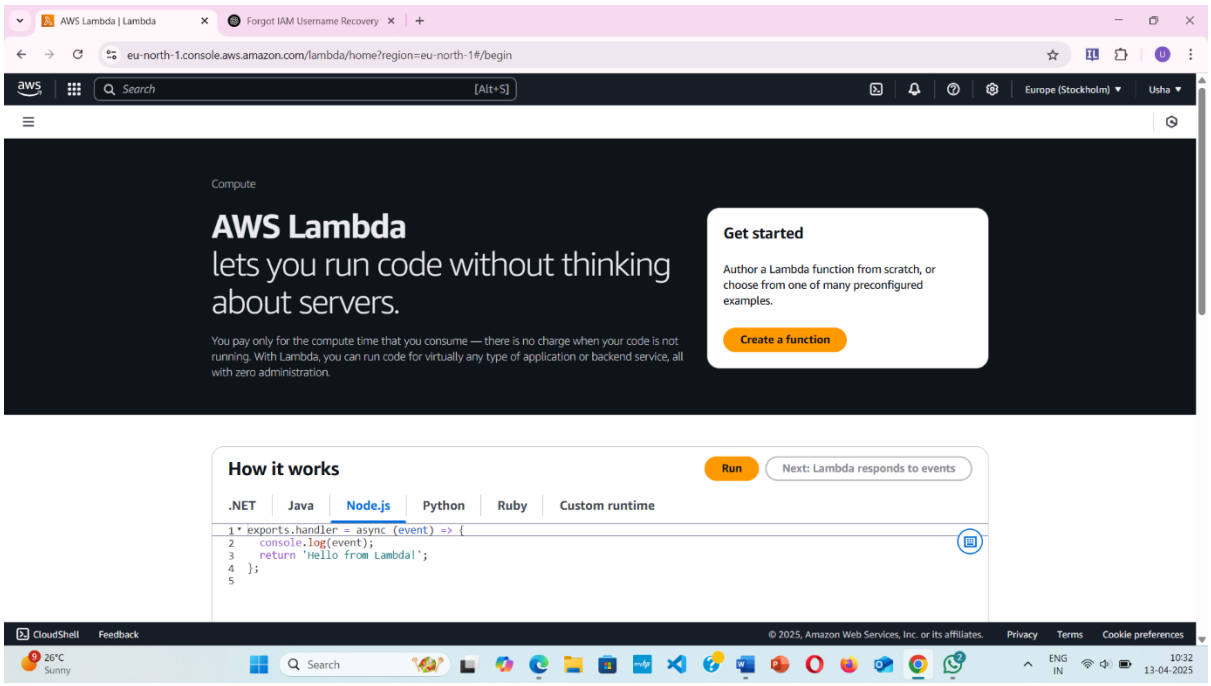
Apparatus Required: AWS Account

Pre-Requisite: Pre - requisite is creation of Lambda Function

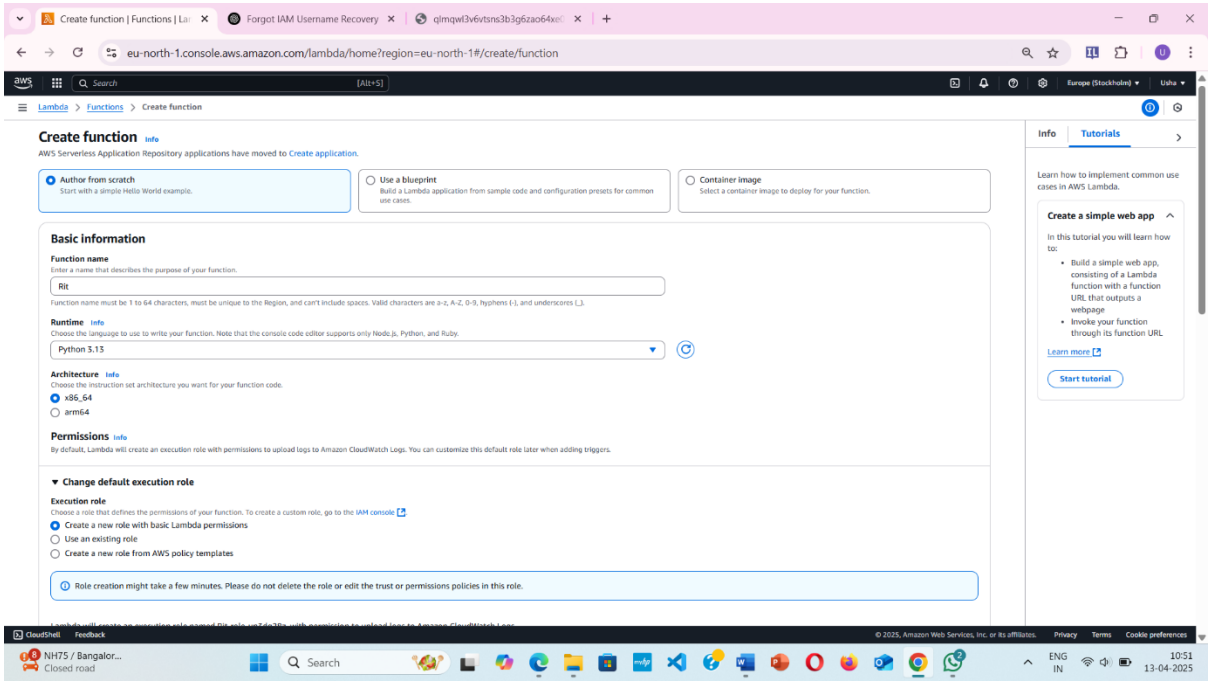
Introduction: API Gateway enables you to connect and access data, business logic, and functionality from backend services such as workloads running on Amazon Elastic Compute Cloud (Amazon EC2), code running on AWS Lambda, any web application, or real-time communication applications.

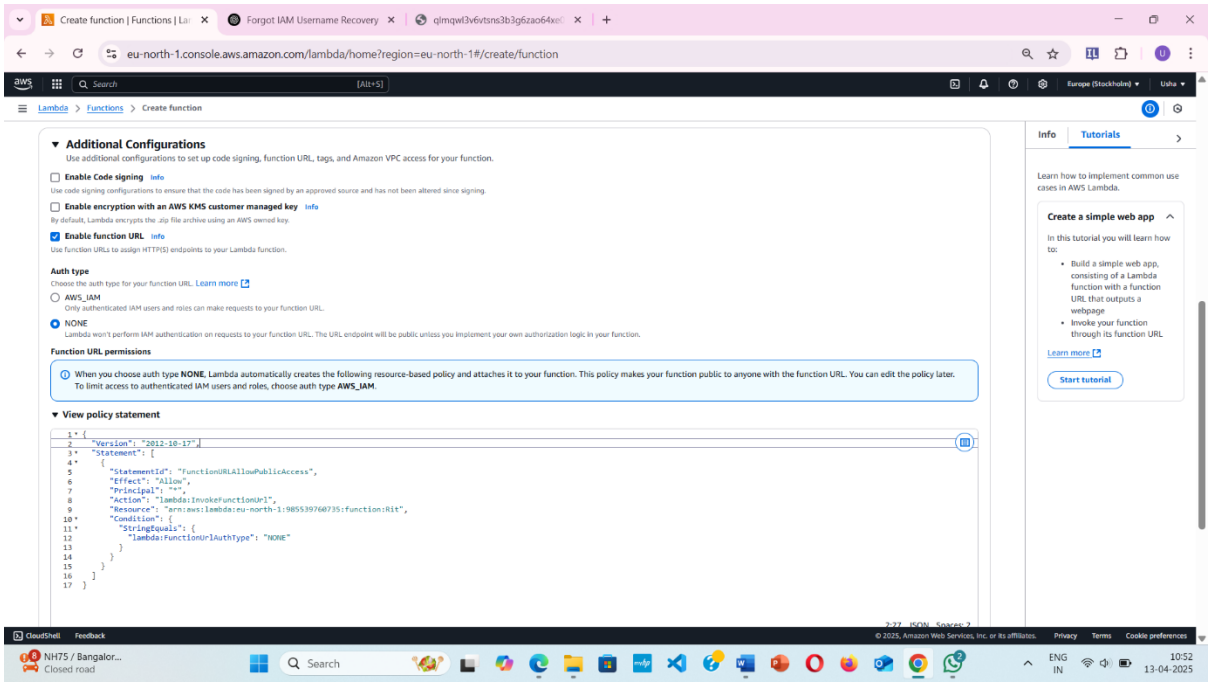
Procedure

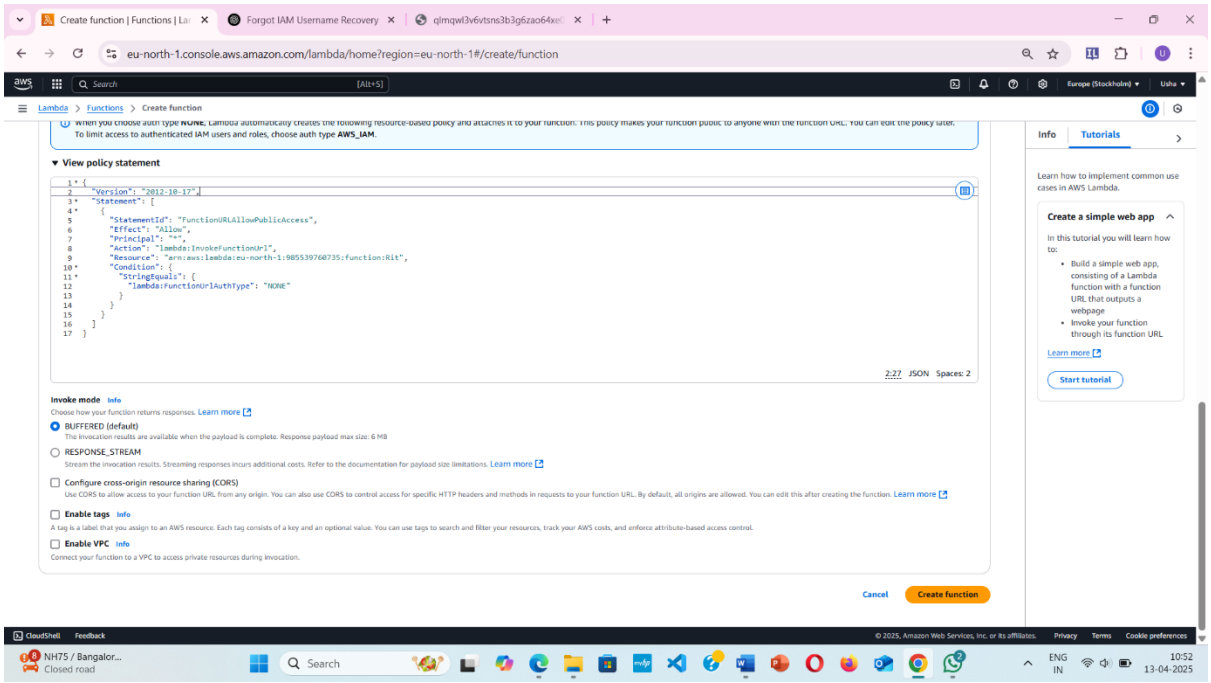
Open AWS Lambda



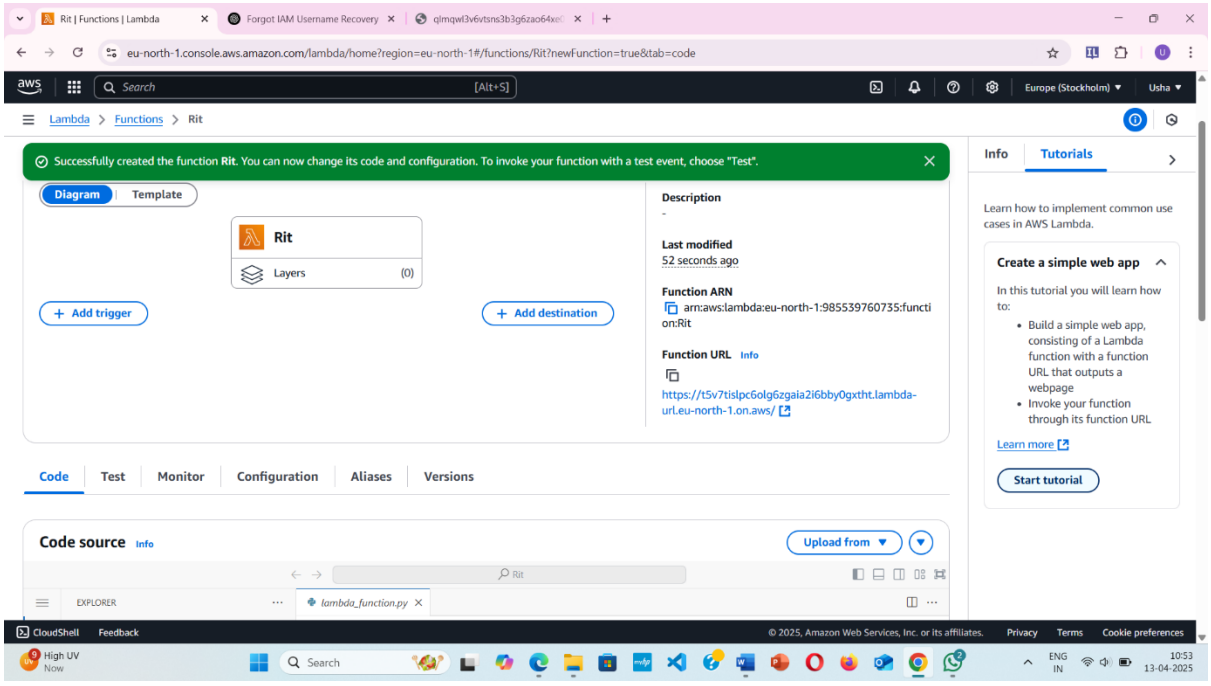
Create function

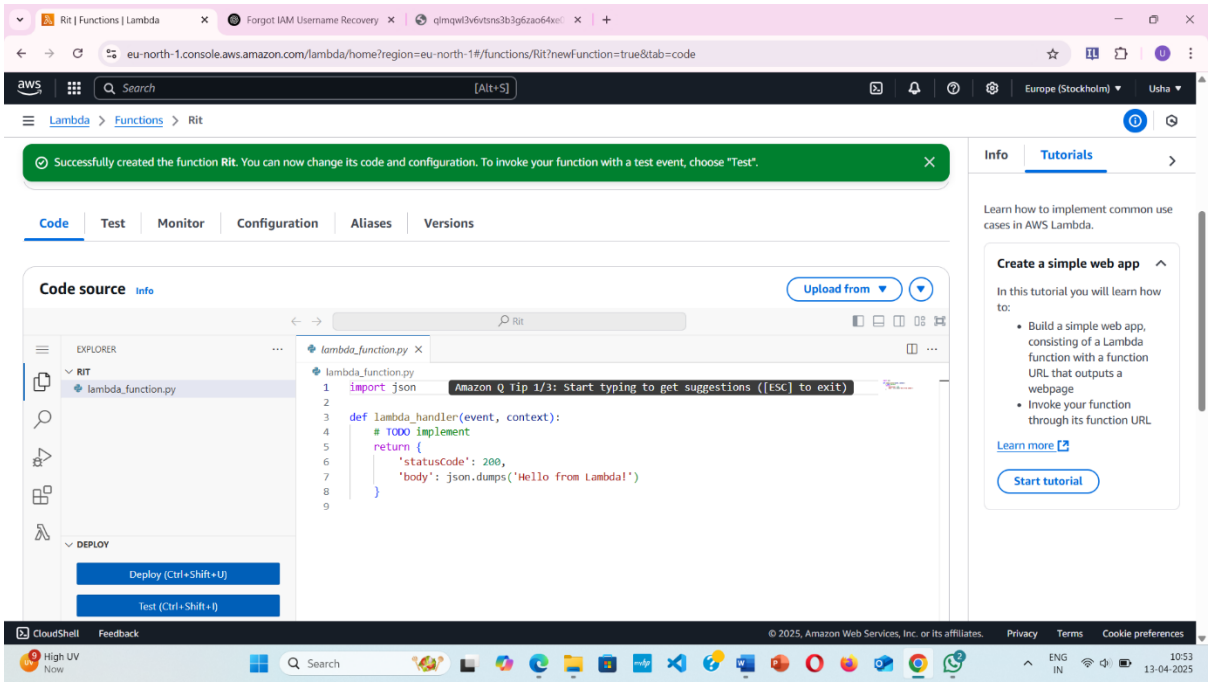




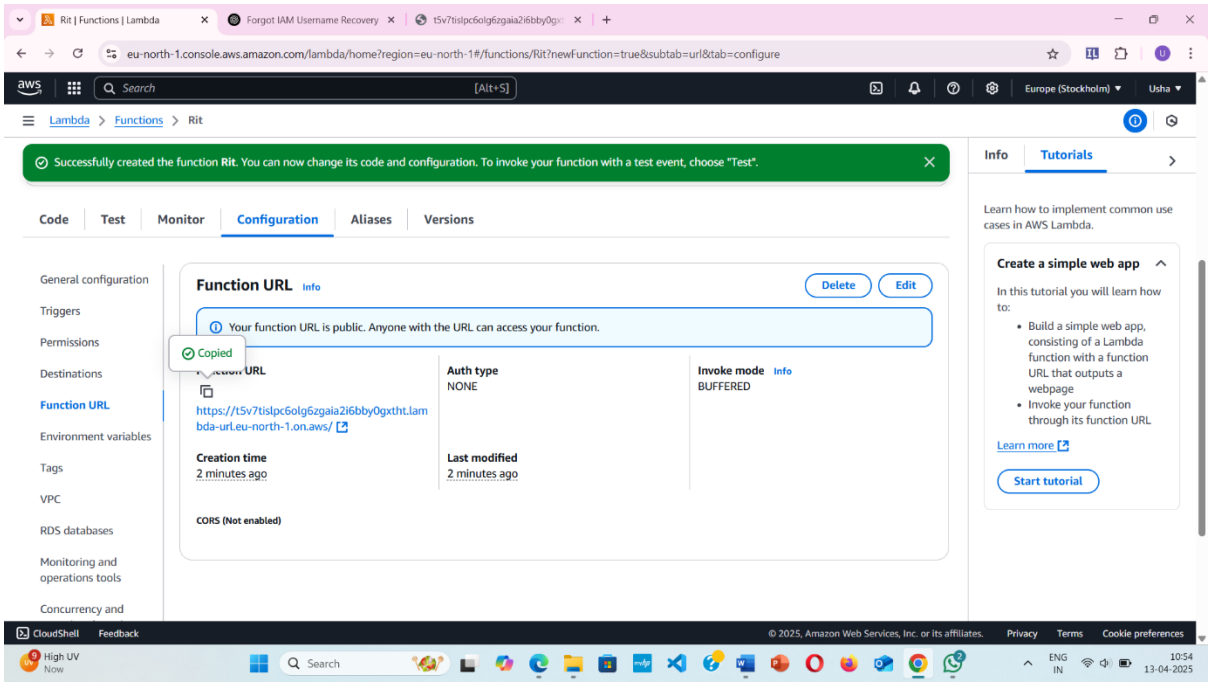


Successfully created the function





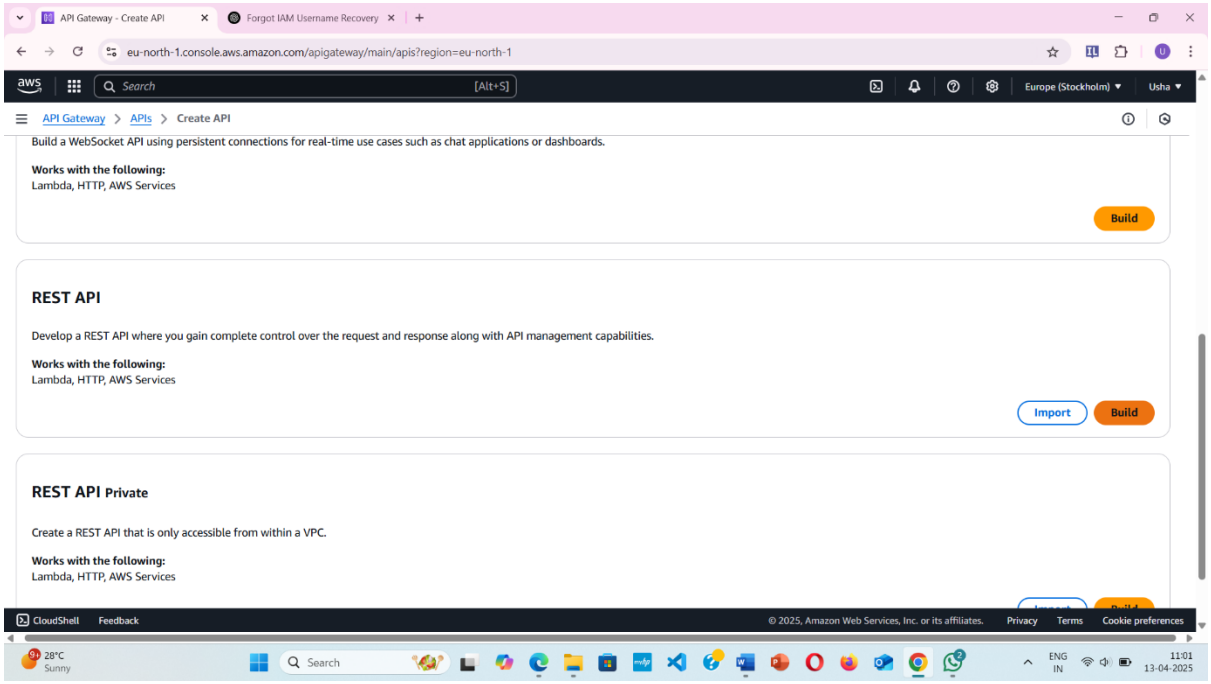
Configuration→ function URL→check URL runs



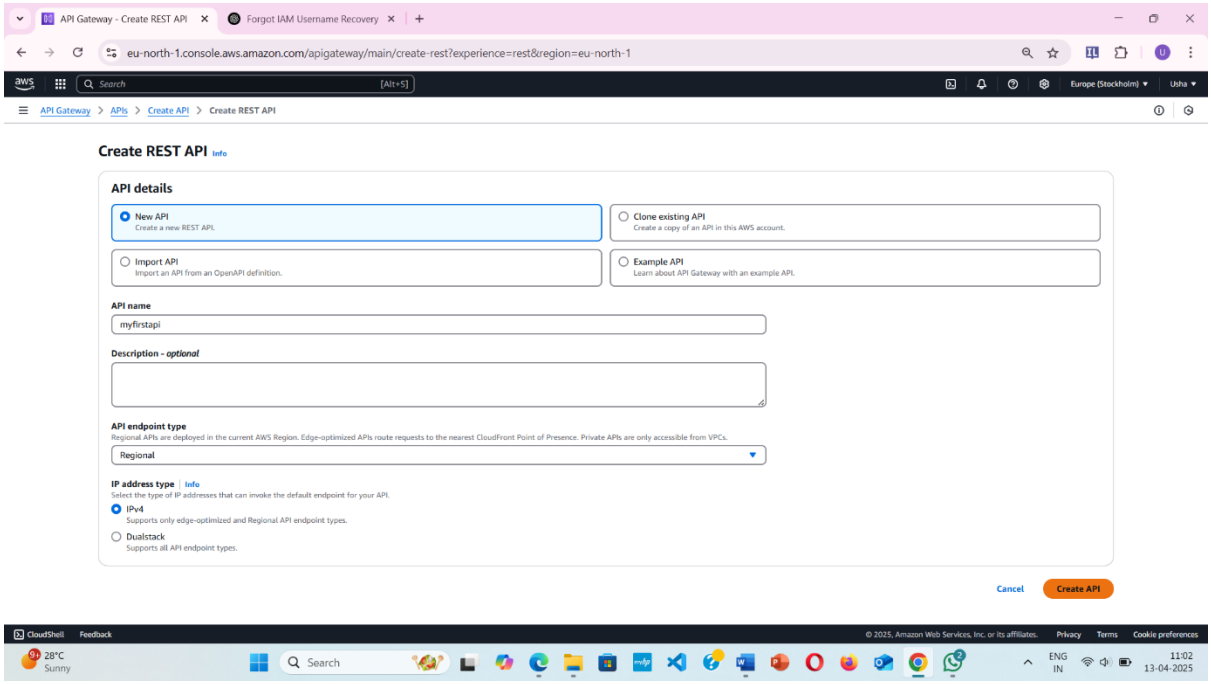
Lambda Code Samples

```
import json
print(json.dumps({"name": "John", "age": 30}))
print(json.dumps(["apple", "banan as"]))
print(json.dumps(("apple", "bananas")))
print(json.dumps("hello"))
print(json.dumps(42))
print(json.dumps(31.76))
print(json.dumps(True))
print(json.dumps(False))
print(json.dumps(None))
```

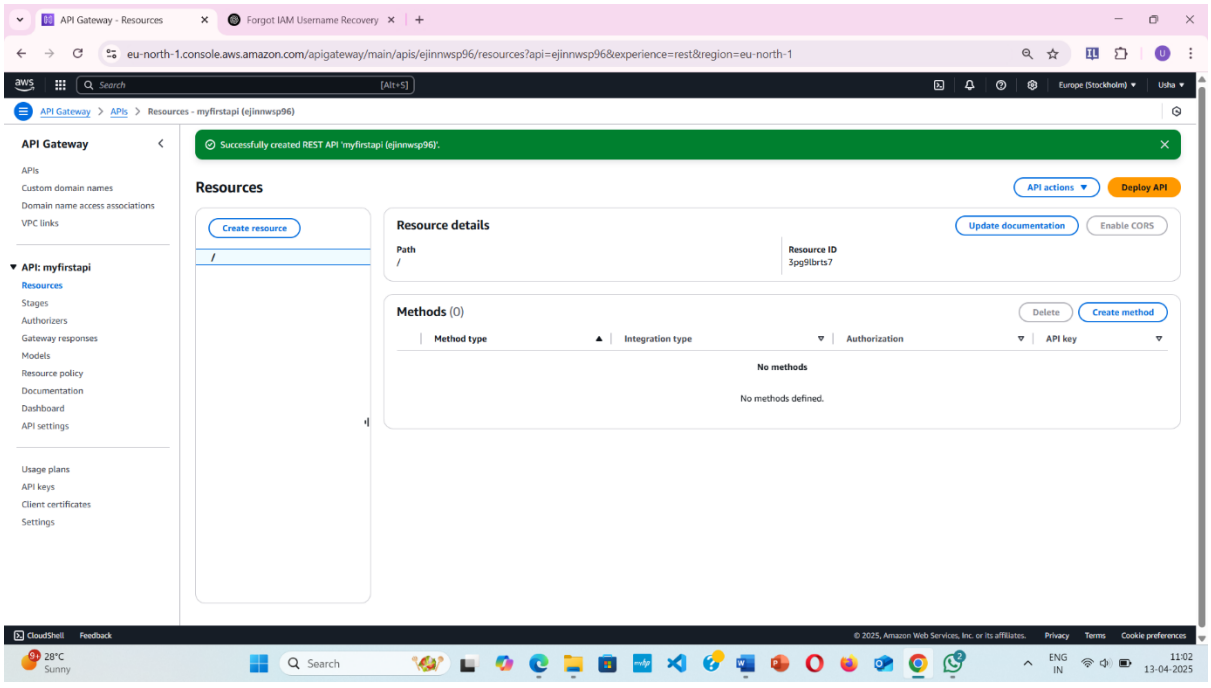
Go to API Gateway in Console And Select Rest API



Build Rest API

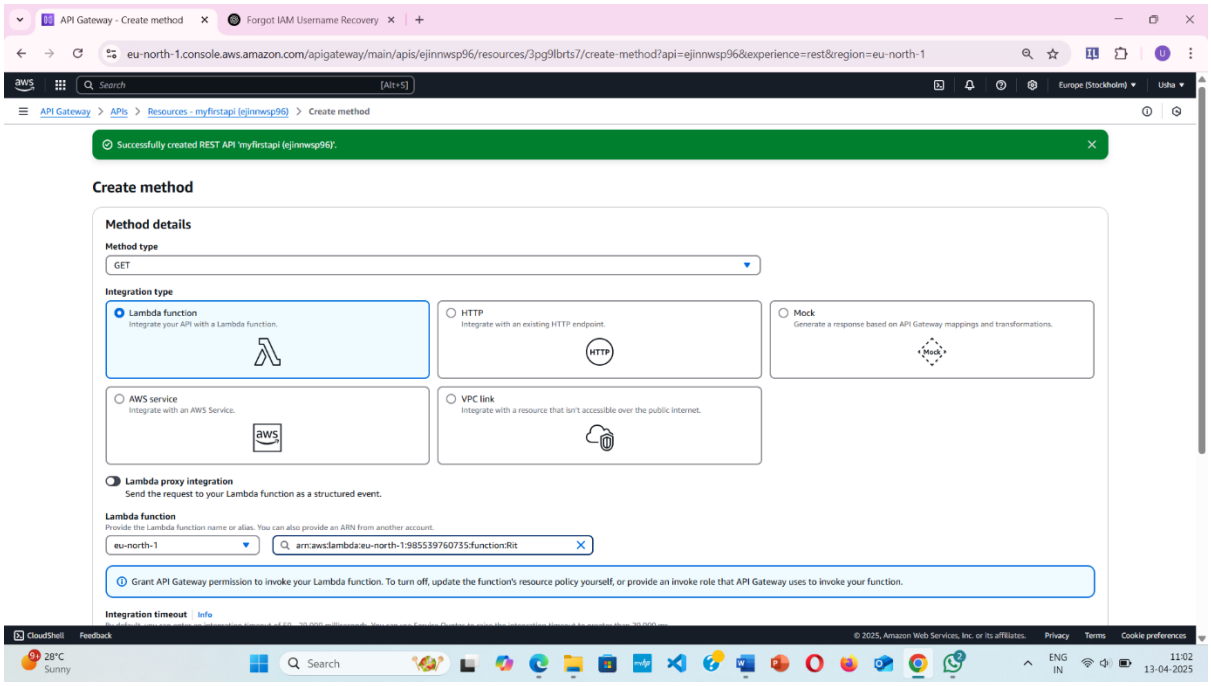


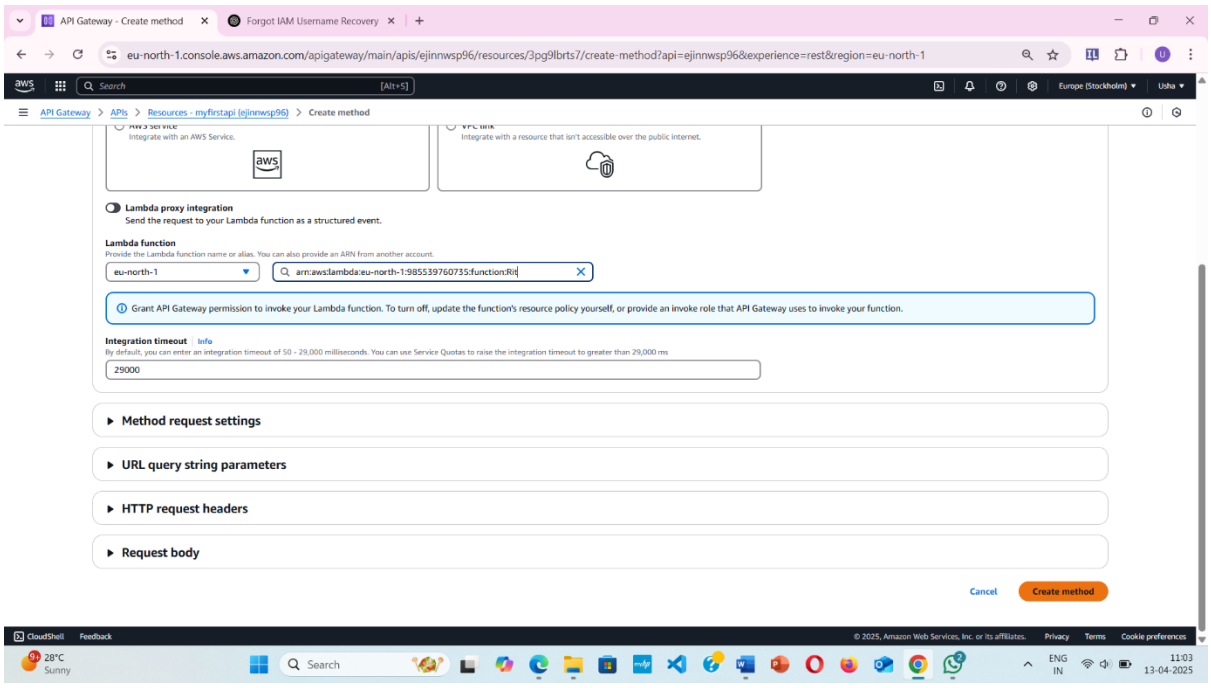
Successfully created the REST API



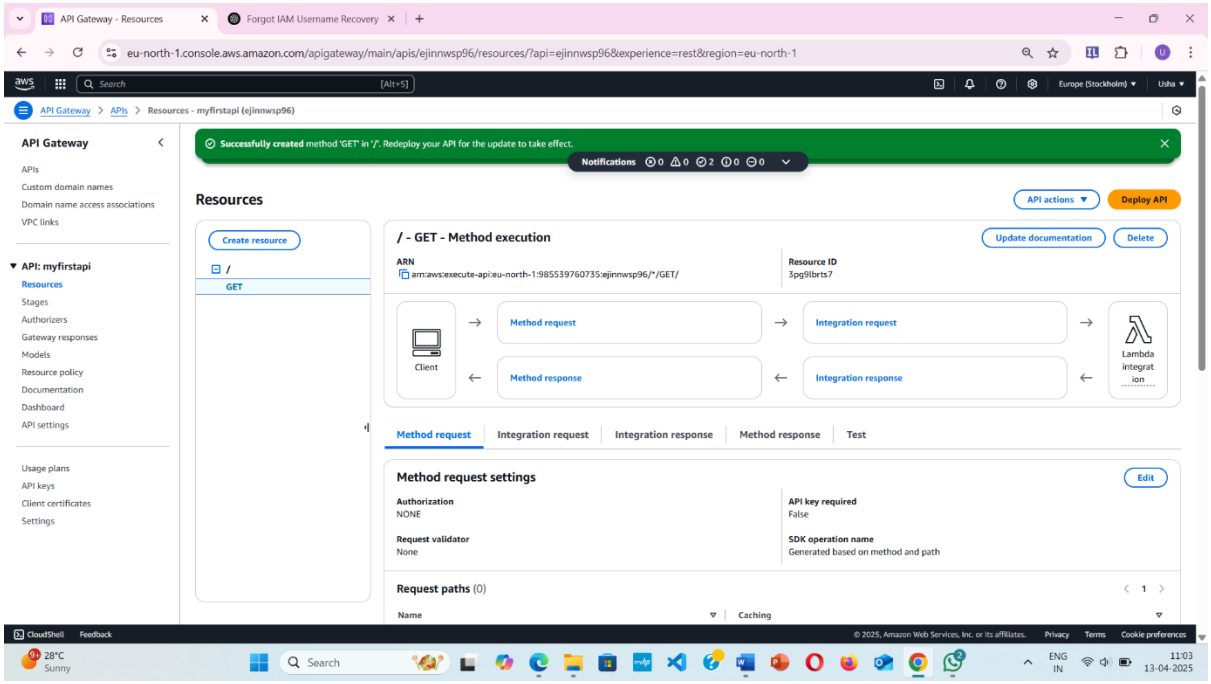
Create method And Select Lambda Integration

Select GET method



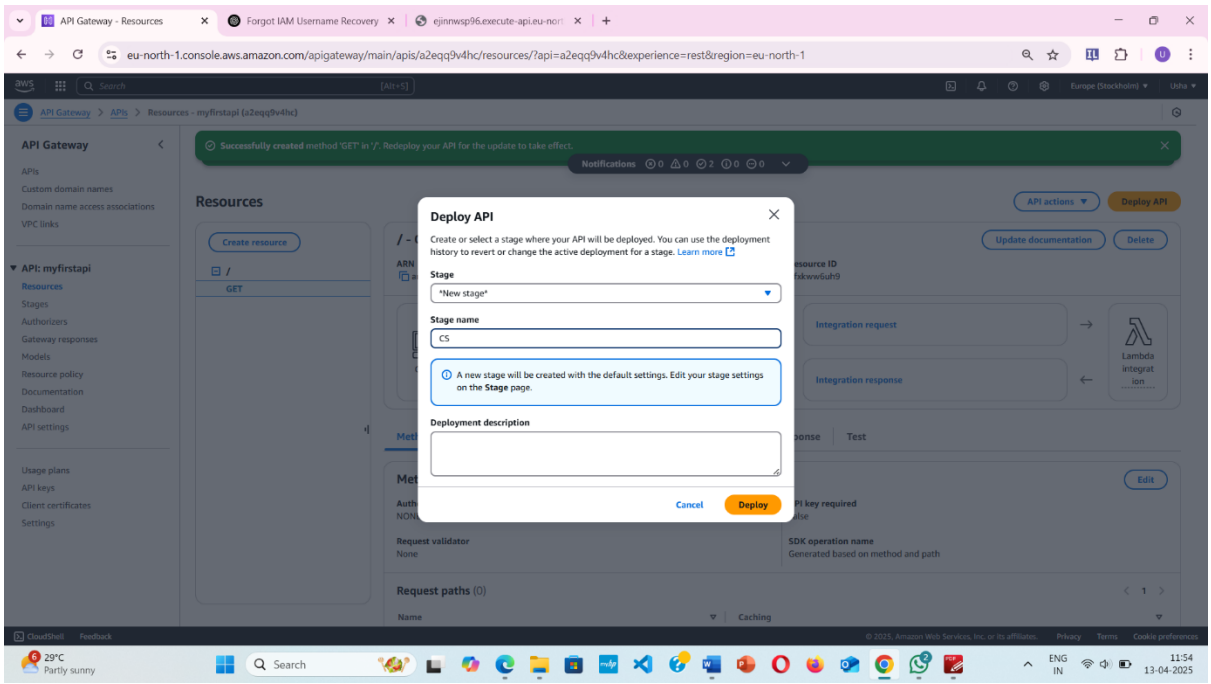


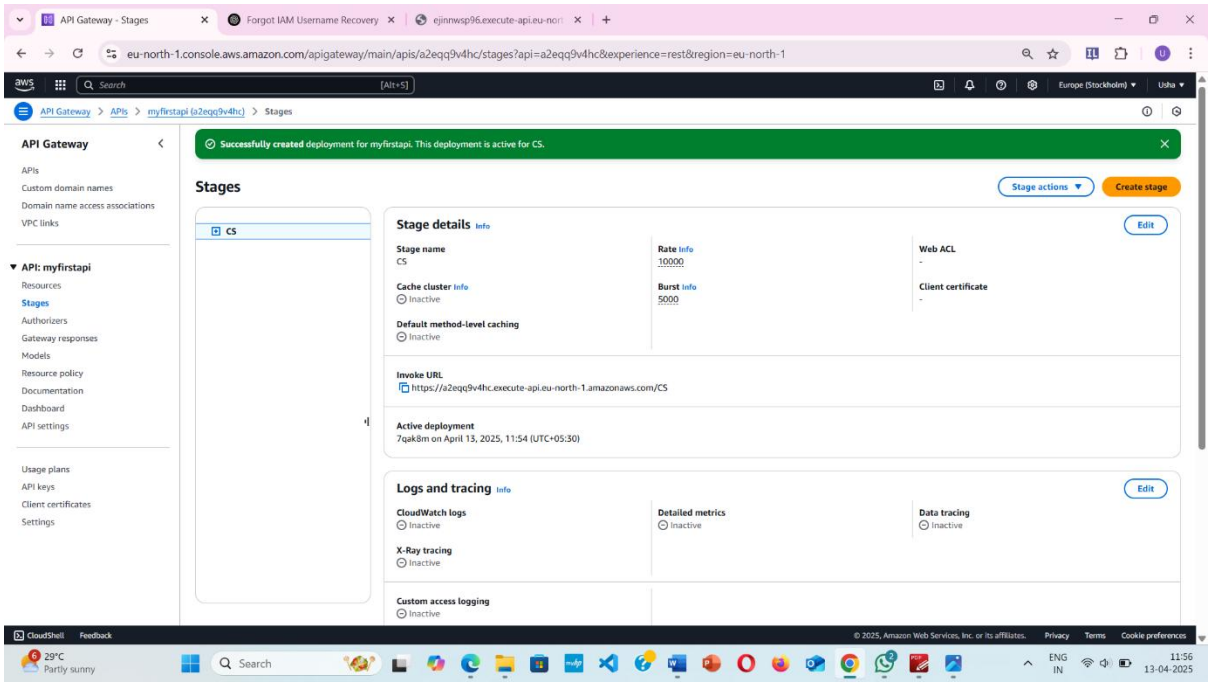
Successfully created method “GET”



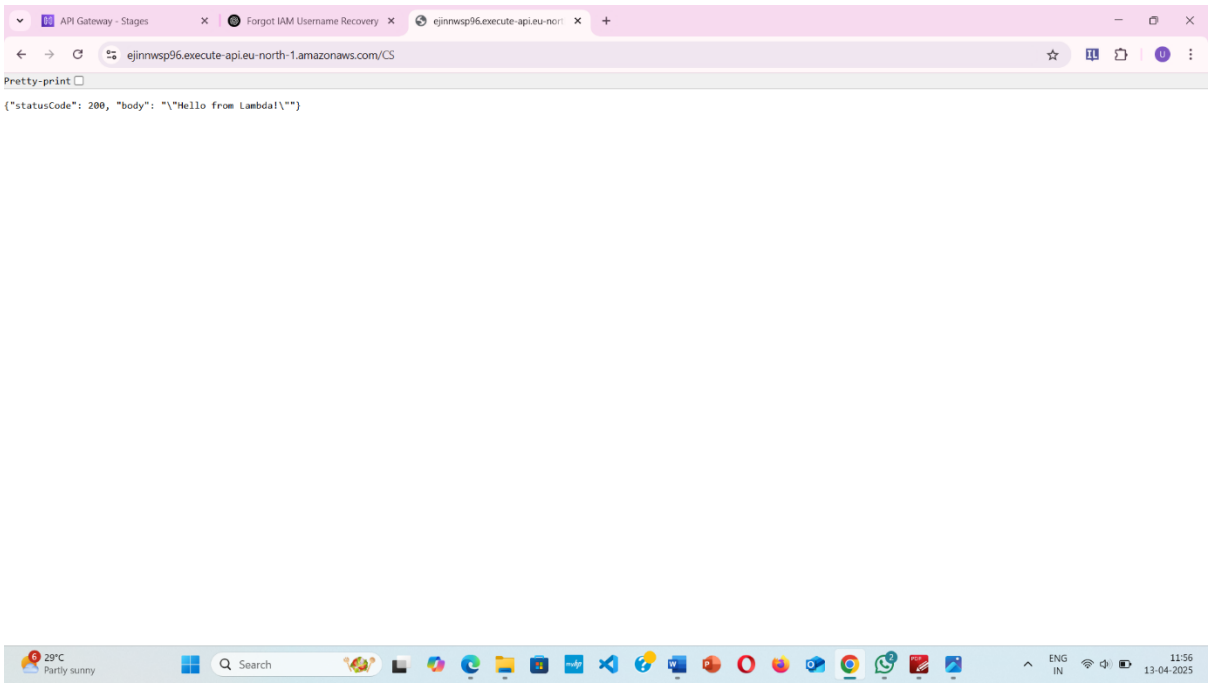
To Test API

Deploy API





Use URL and copy in new tab



Experiment No – 5

Creating Lambda Functions Using the AWS SDK for Python

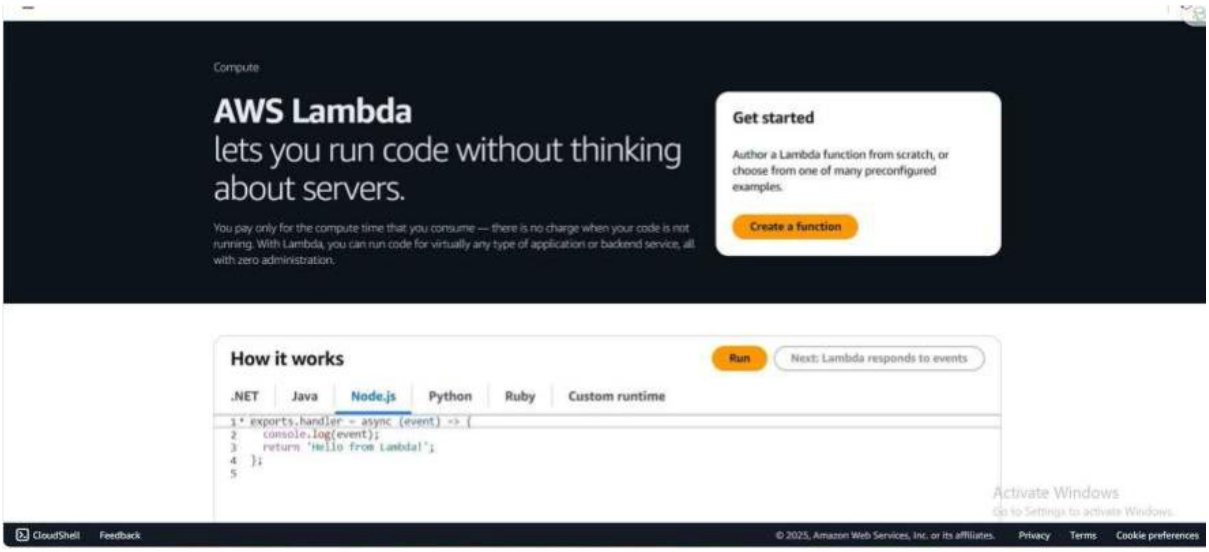
Objectives: Creating Lambda Functions Using the AWS SDK for Python

Apparatus Required: AWS Account

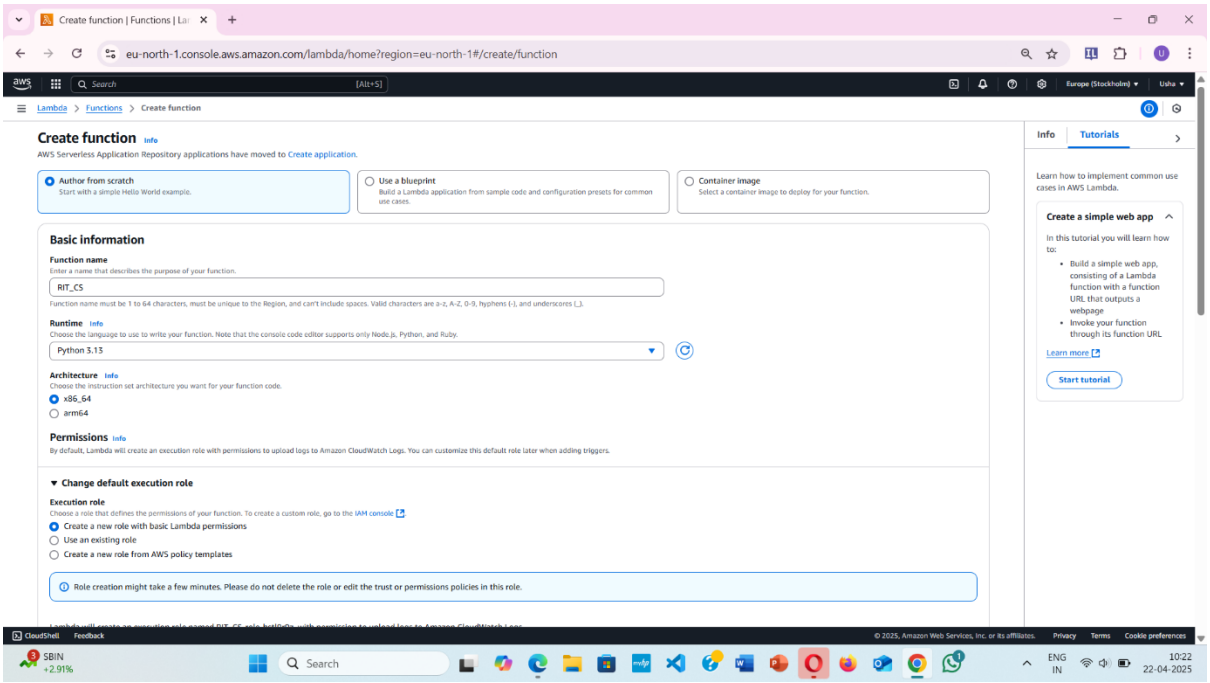
Pre-Requisite: AWS basic knowledge, basics of lambda functions and Python

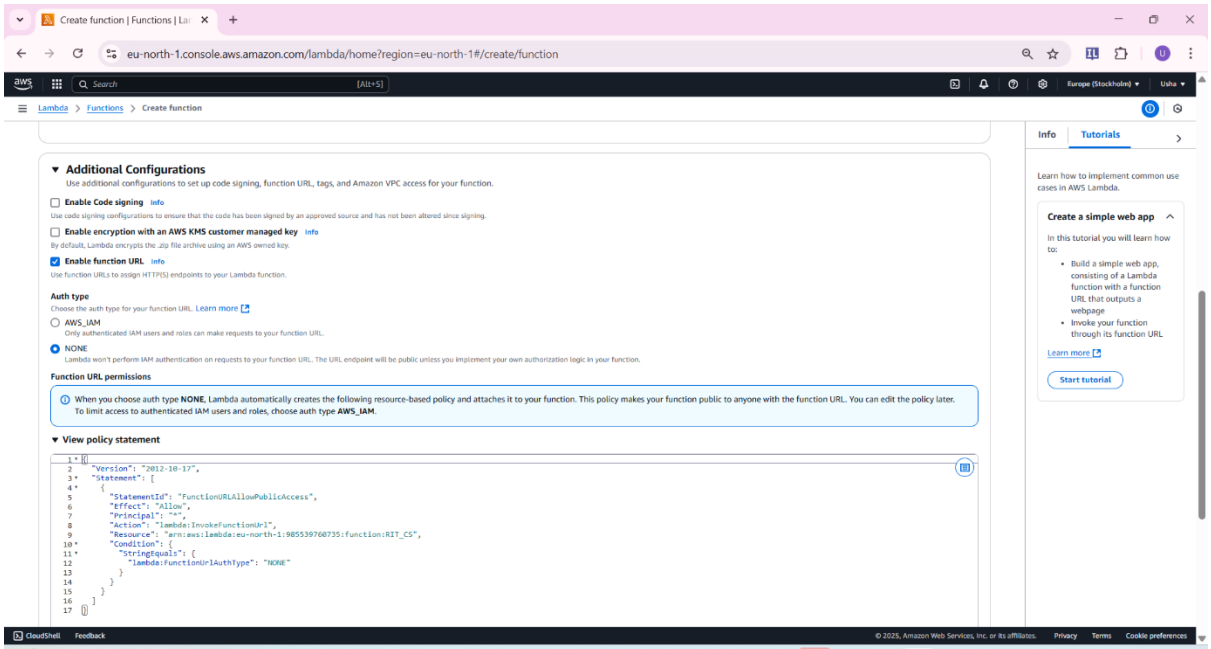
Introduction: An AWS Lambda function is a serverless function that runs code in response to events, such as user actions or system updates. You can use Lambda functions to create custom backend services, extend other AWS services, or trigger actions in other services.

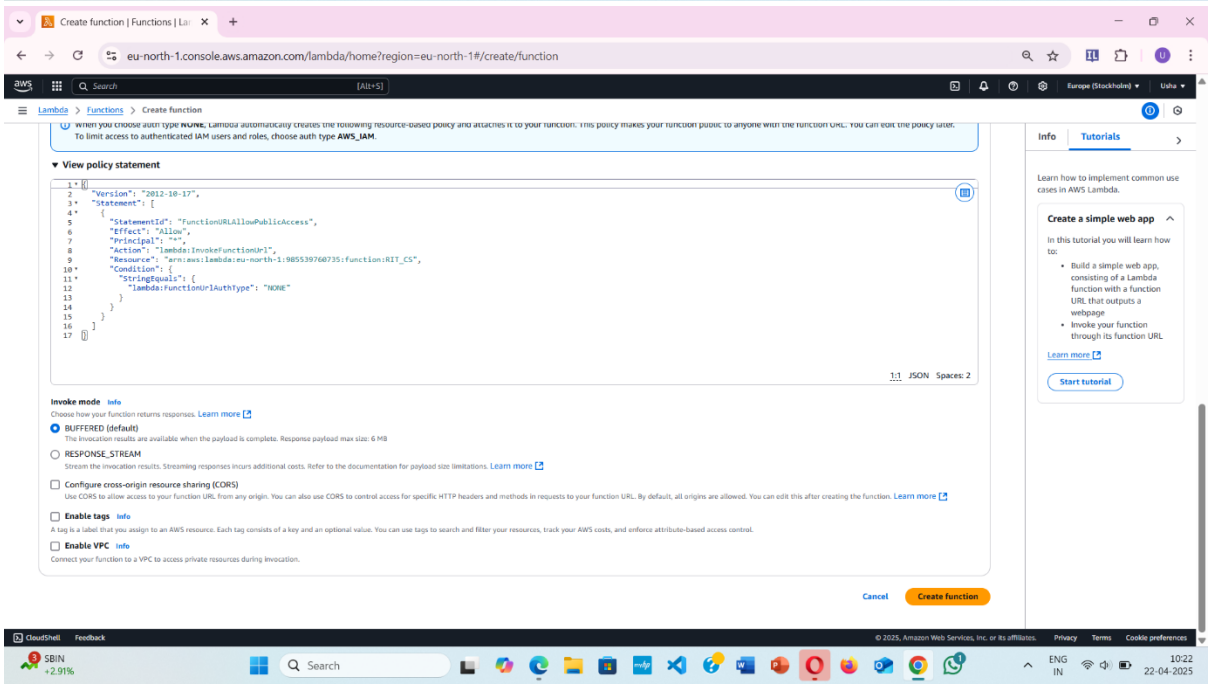
Procedure:

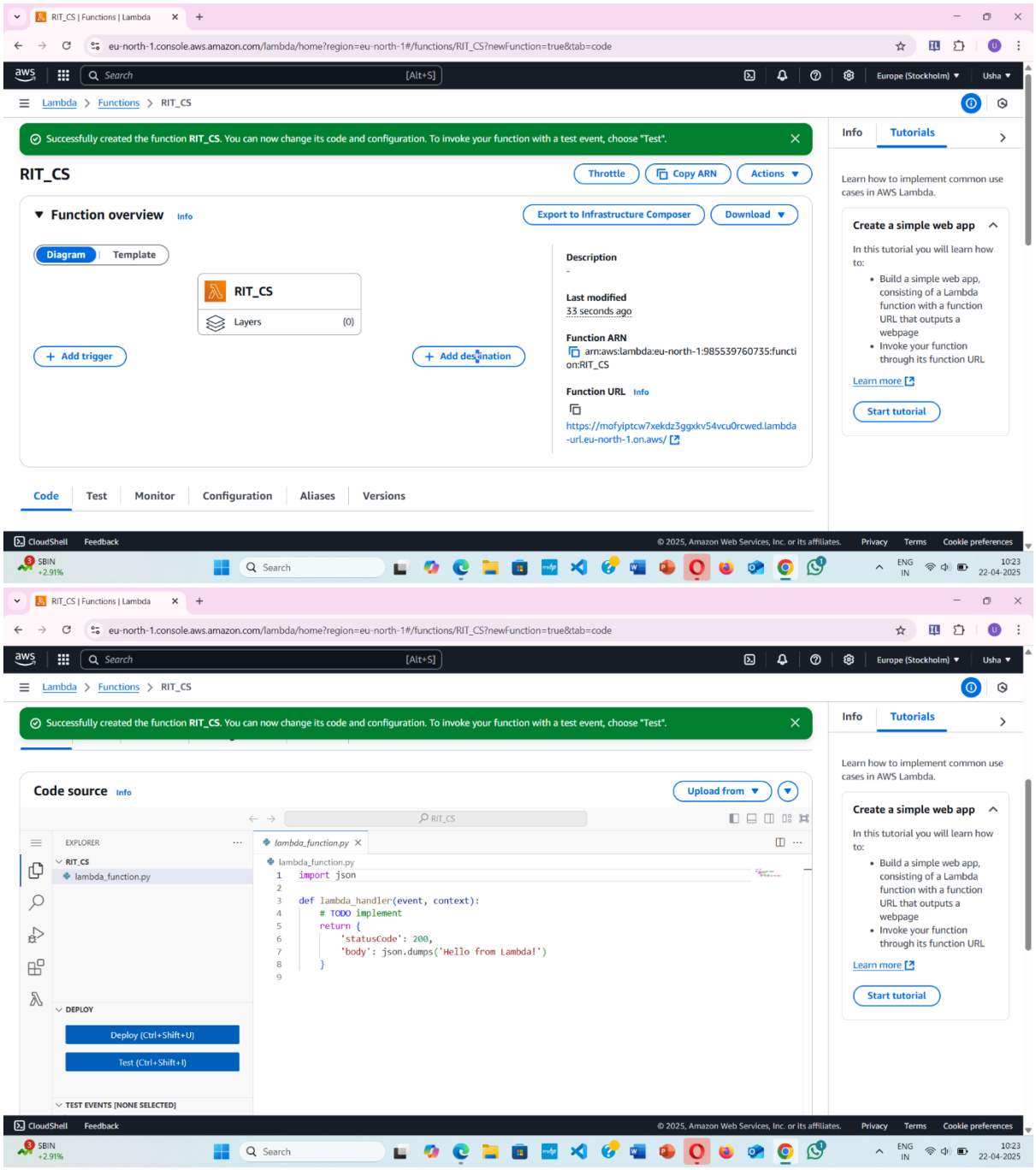


CREATE FUNCTION



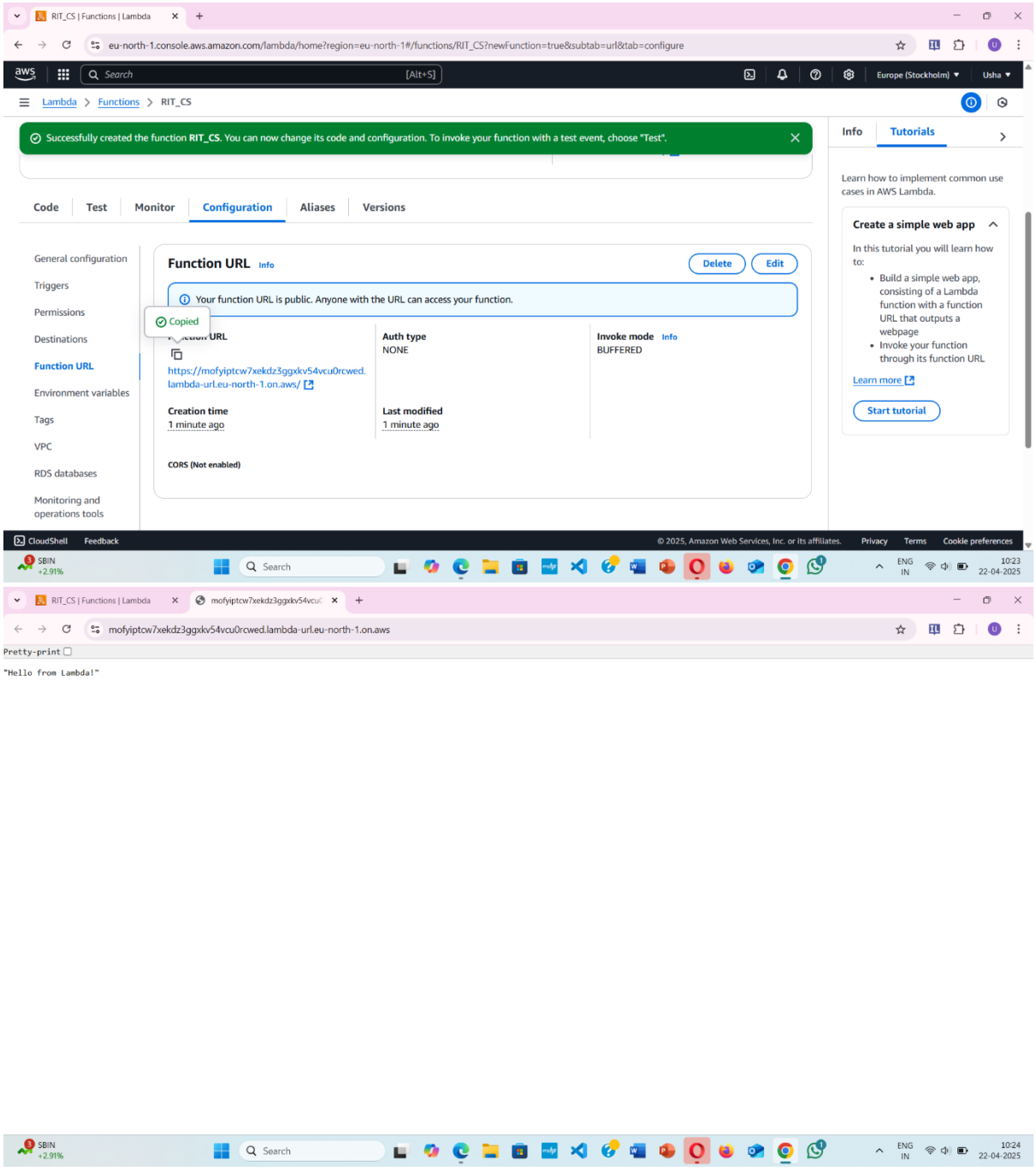

Create function | Functions | La
eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/create/function
Lambda > Functions > Create function
Additional Configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.
Enable Code signing Info
Use code signing configurations to ensure that the code has been signed by an approved source and has not been altered since signing.
Enable encryption with an AWS KMS customer managed key Info
By default, Lambda encrypts the .zip file archive using an AWS owned key.
Enable function URL Info
Use function URLs to assign HTTP(S) endpoints to your Lambda function.
Auth type
Choose the auth type for your function URL. Learn more
AWS_IAM
Only authenticated IAM users and roles can make requests to your function URL.
NONE
Lambda won't perform IAM authentication on requests to your function URL. The URL endpoint will be public unless you implement your own authorization logic in your function.
Function URL permissions
When you choose auth type NONE, Lambda automatically creates the following resource-based policy and attaches it to your function. This policy makes your function public to anyone with the function URL. You can edit the policy later. To limit access to authenticated IAM users and roles, choose auth type AWS_IAM.
View policy statement
{
"Version": "2012-10-17",
"Statement": [
{
"StatementId": "FunctionURLAllowPublicAccess",
"Effect": "Allow",
"Principal": "*",
"Action": "lambda:InvokeFunctionUrl",
"Resource": "arn:aws:lambda:eu-north-1:985539760735:function:RIT_CS",
"Condition": {
"StringEquals": {
"lambda:FunctionUrlAuthType": "NONE"
}
}
}
]
}
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
SBIN
+2.91%
Search
ENG
IN
10:22
22-04-2025


Create function | Functions | La
eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/create/function
Lambda > Functions > Create function
To limit access to authenticated IAM users and roles, choose auth type AWS_IAM.
View policy statement
{
"Version": "2012-10-17",
"Statement": [
{
"StatementId": "FunctionURLAllowPublicAccess",
"Effect": "Allow",
"Principal": "*",
"Action": "lambda:InvokeFunctionUrl",
"Resource": "arn:aws:lambda:eu-north-1:985539760735:function:RIT_CS",
"Condition": {
"StringEquals": {
"lambda:FunctionUrlAuthType": "NONE"
}
}
}
]
}
1:1 JSON Spaces: 2
Invoke mode Info
Choose how your function returns responses. Learn more
BUFFERED (default)
The invocation results are available when the payload is complete. Response payload max size: 6 MB
RESPONSE_STREAM
Stream the invocation results. Streaming responses incurs additional costs. Refer to the documentation for payload size limitations. Learn more
Configure cross-origin resource sharing (CORS)
Use CORS to allow access to your function URL from any origin. You can also use CORS to control access for specific HTTP headers and methods in requests to your function URL. By default, all origins are allowed. You can edit this after creating the function. Learn more
Enable tags Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources, track your AWS costs, and enforce attribute-based access control.
Enable VPC Info
Connect your function to a VPC to access private resources during invocation.
Cancel
Create function
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
SBIN
+2.91%
Search
ENG
IN
10:22
22-04-2025


RIT_CS | Functions | Lambda
eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/functions/RIT_CS?newFunction=true&tab=code
Search [Alt+S]
Europe (Stockholm)
Usha
Lambda > Functions > RIT_CS
Successfully created the function RIT_CS. You can now change its code and configuration. To invoke your function with a test event, choose "Test".
RIT_CS
Throttle
Copy ARN
Actions
Function overview Info
Export to Infrastructure Composer
Download
Diagram
Template
RIT_CS
Layers (0)
+ Add trigger
+ Add destination
Description
-
Last modified
33 seconds ago
Function ARN
arn:aws:lambda:eu-north-1:985539760735:function:RIT_CS
Function URL Info
https://mofyiptcw7xekdz3ggxkv54vcu0rcwed.lambda-url.eu-north-1.on.aws/
Code
Test
Monitor
Configuration
Aliases
Versions
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
SBIN +2.91%
Search
ENG IN
10:23 22-04-2025

RIT_CS | Functions | Lambda
eu-north-1.console.aws.amazon.com/lambda/home?region=eu-north-1#/functions/RIT_CS?newFunction=true&tab=code
Search [Alt+S]
Europe (Stockholm)
Usha
Lambda > Functions > RIT_CS
Successfully created the function RIT_CS. You can now change its code and configuration. To invoke your function with a test event, choose "Test".
Code source Info
Upload from
EXPLORER
RIT_CS
lambda_function.py
DEPLOY
Deploy (Ctrl+Shift+U)
Test (Ctrl+Shift+I)
TEST EVENTS [NONE SELECTED]
lambda_function.py
1 import json
2
3 def lambda_handler(event, context):
4 # TODO implement
5 return {
6 'statusCode': 200,
7 'body': json.dumps('Hello from Lambda!')
8 }
9
Info
Tutorials
Learn how to implement common use cases in AWS Lambda.
Create a simple web app
In this tutorial you will learn how to:
Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
Invoke your function through its function URL
Learn more
Start tutorial
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
SBIN +2.91%
Search
ENG IN
10:23 22-04-2025

CONFIGERATIONS →FUNCTION URL → CHECK URL RUNS



Experiment No – 06

Migrating a Web Application to Docker Containers

Objectives: Migrating a Web Application to Docker Containers

Apparatus Required: AWS Account

Pre-Requisite: AWS basics knowledge, Basics of docker

Introduction: Docker is a set of platform as a service (PaaS) products that use OS-level virtualization to deliver software in packages called containers.

Docker is a software platform that packages software into containers.

Docker images are read-only templates that contain instructions for creating a Container.

A Docker image is a snapshot or blueprint of the libraries and dependencies required inside a container for an application to run.

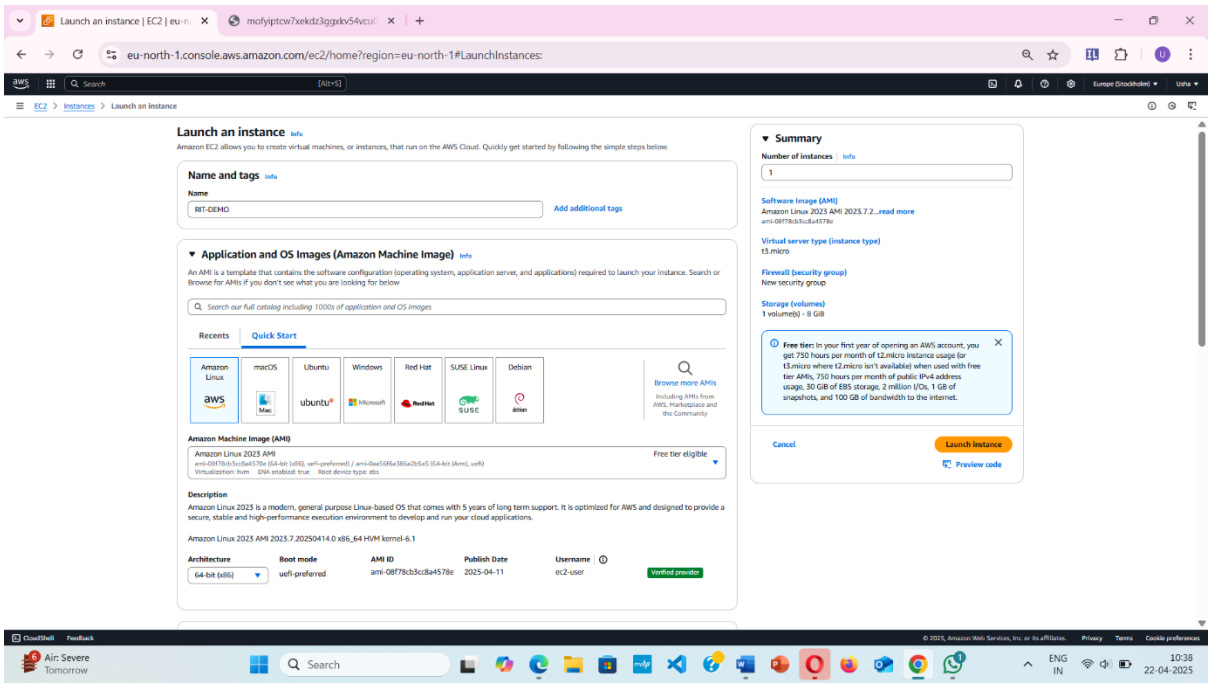
A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime,

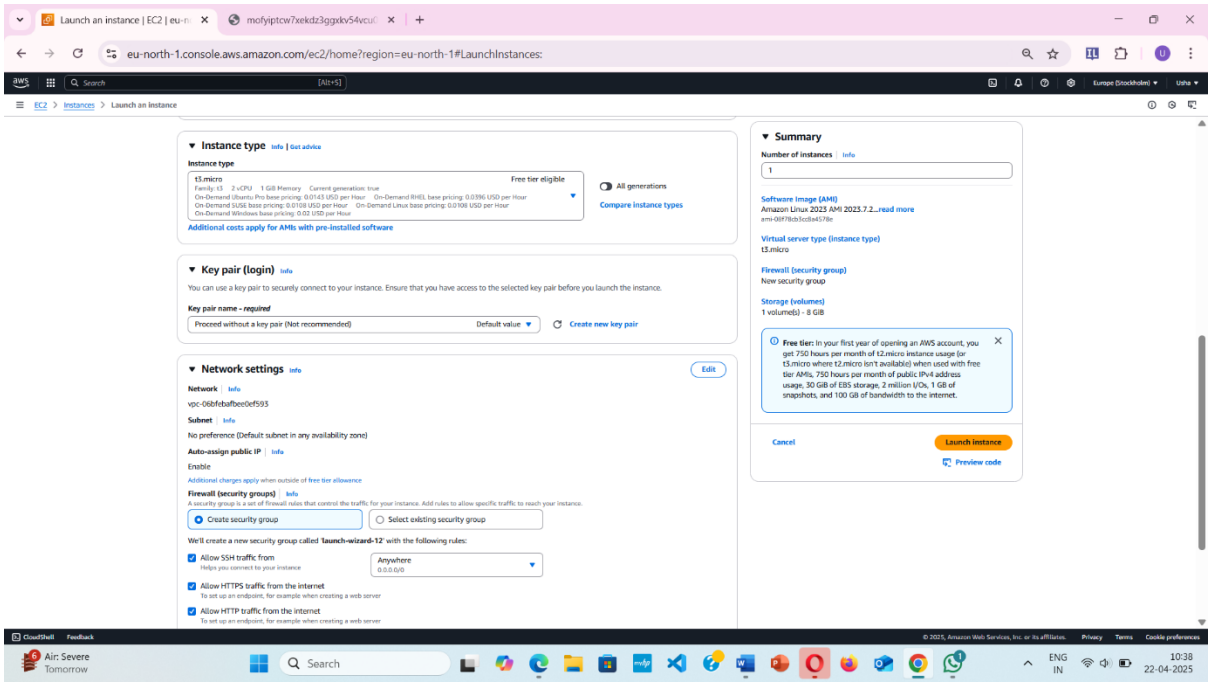
Procedure:

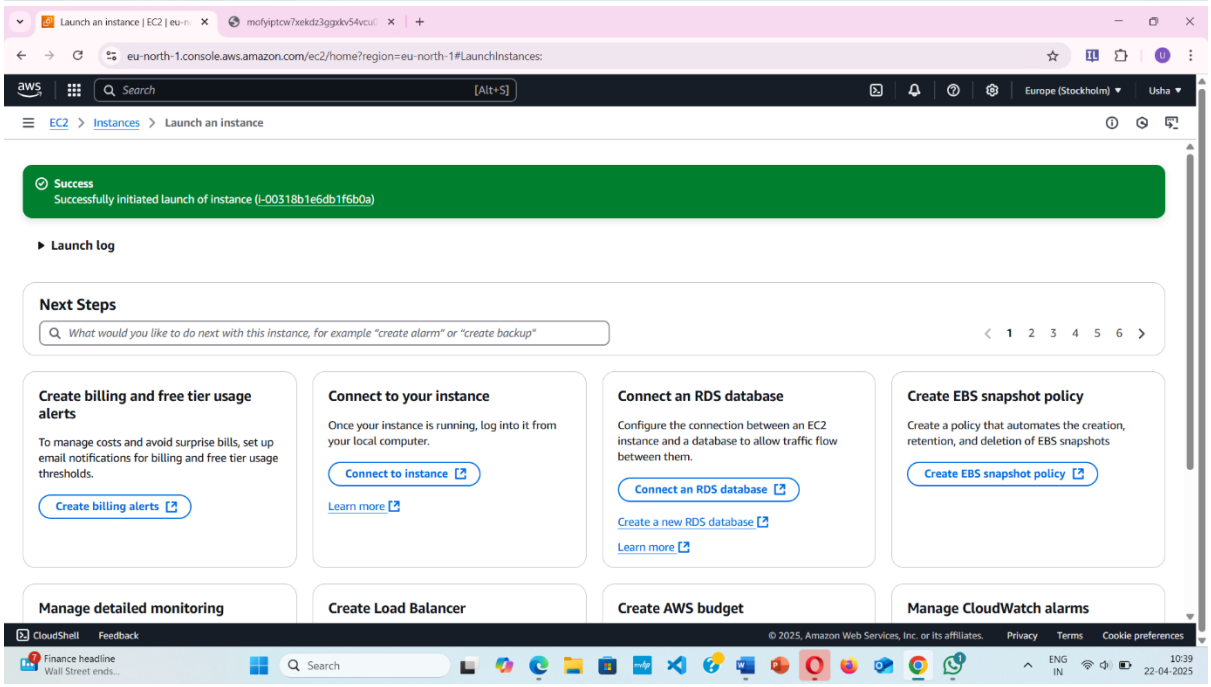
Create EC2 Instance

Linux Machine

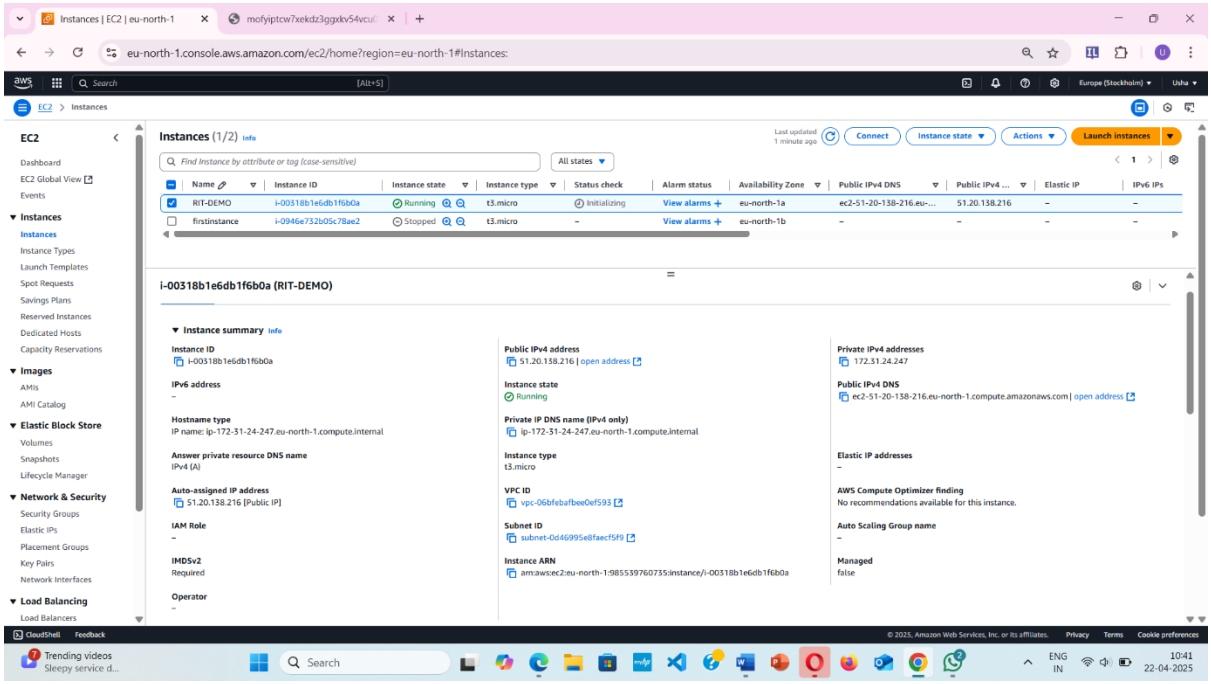
Enable HTTPS and H TTP



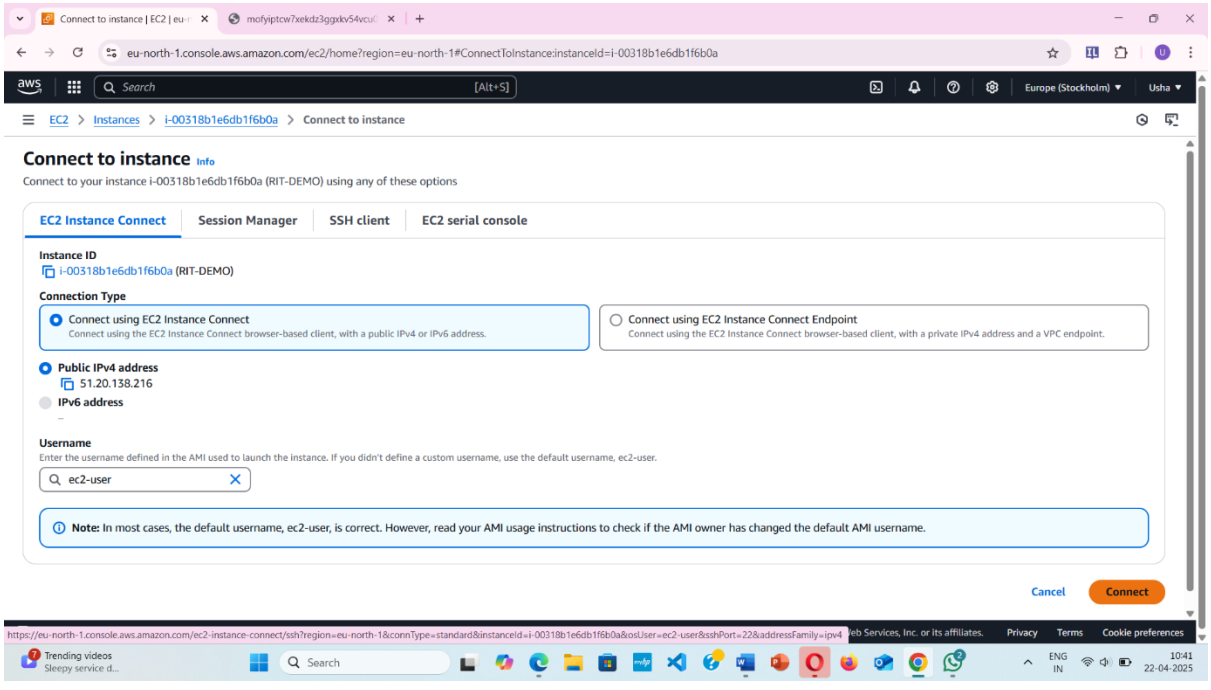

Launch an instance | EC2 | eu-north-1
eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#LaunchInstances:
EC2 > Instances > Launch an instance
Instance type
t3.micro
Free tier eligible
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Key pair (login)
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
Proceed without a key pair (Not recommended)
Default value
Create new key pair
Network settings
Edit
Network
vpc-06bfebafbee0ef593
Subnet
No preference (Default subnet in any availability zone)
Auto-assign public IP
Enable
Firewall (security groups)
Create security group
Select existing security group
We'll create a new security group called 'launch-wizard-12' with the following rules:
Allow SSH traffic from
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
Allow HTTP traffic from the internet
Summary
Number of instances
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.7.2...read more
Virtual server type (instance type)
t3.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Preview code
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Air: Severe
Tomorrow
Search
ENG
IN
10:38
22-04-2025


Launch an instance | EC2 | eu-north-1
eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#LaunchInstances:
Search
[Alt+S]
Europe (Stockholm)
Usha
EC2 > Instances > Launch an instance
Success
Successfully initiated launch of instance (i-00318b1e6db1f6b0a)
Launch log
Next Steps
What would you like to do next with this instance, for example "create alarm" or "create backup"
1 2 3 4 5 6
Create billing and free tier usage alerts
To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.
Create billing alerts
Connect to your instance
Once your instance is running, log into it from your local computer.
Connect to instance
Learn more
Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
Connect an RDS database
Create a new RDS database
Learn more
Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
Create EBS snapshot policy
Manage detailed monitoring
Create Load Balancer
Create AWS budget
Manage CloudWatch alarms
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Finance headline
Wall Street ends...
Search
ENG
IN
10:39
22-04-2025

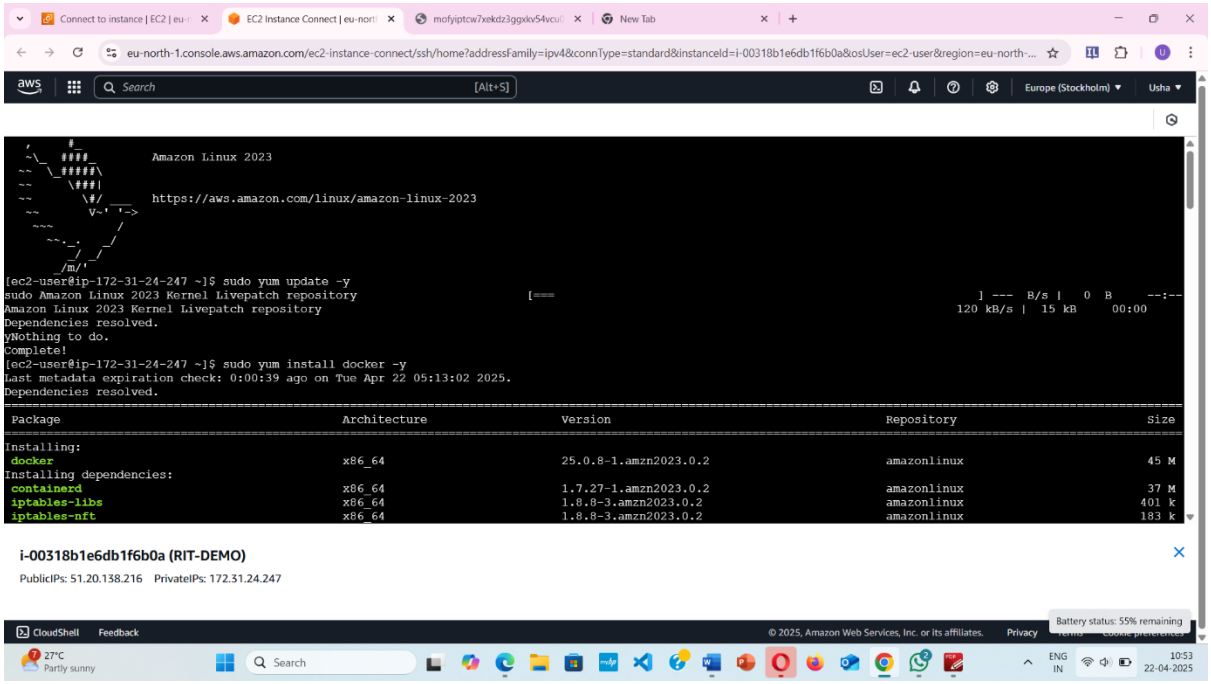
Go To Instance



Press connect



It will take you to Linux Machine



EXECUTE BELOW COMMANDS IN LINUX ENGINE

sudo yum update –y

sudo yum install docker –y

sudo service docker start

sudo service docker status

(ctrl Z)

sudo su

docker version

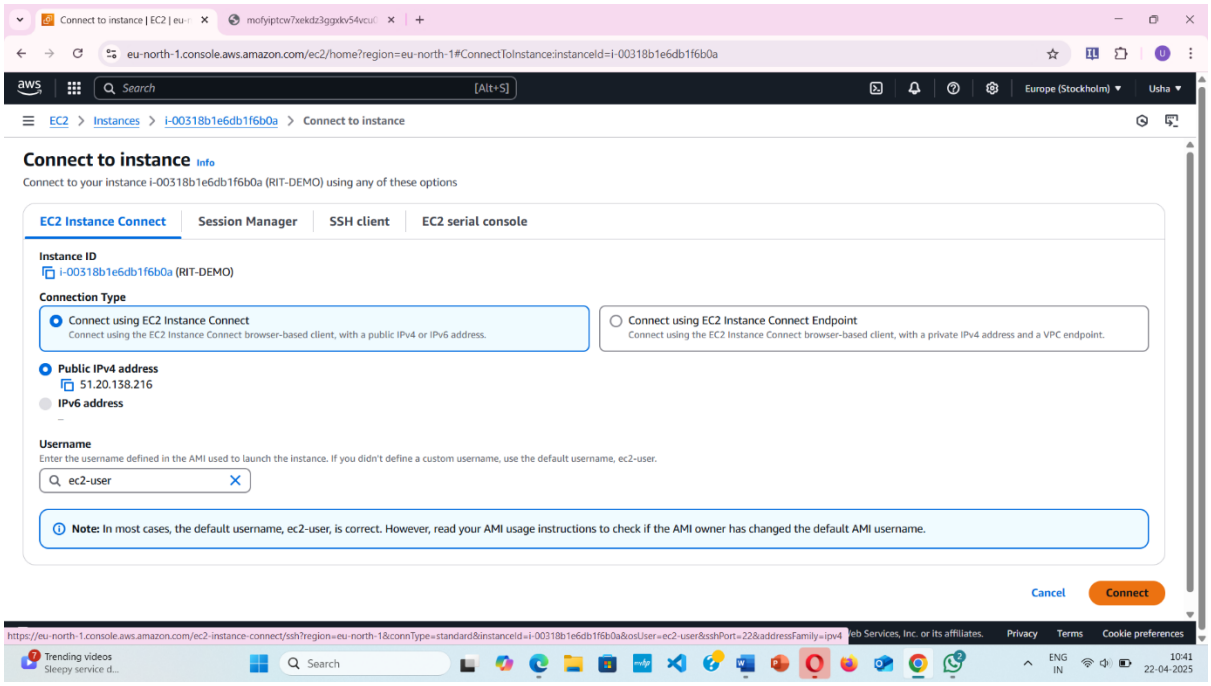
docker pull nginx

docker images

docker run -d -p 80:80 nginx

docker ps

after this go to ec2 instance → connect → copy public ip



Paste it in browser