

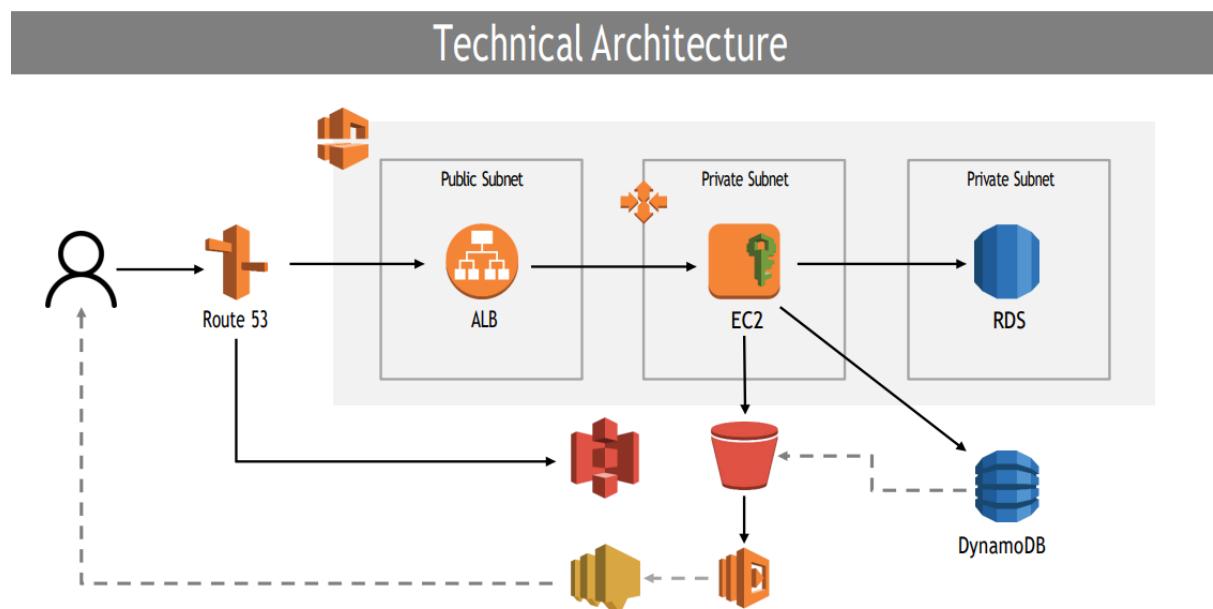
AWS Capstone Case study

Agenda

Lets create an application using a few services that have been discussed in this course.

Application

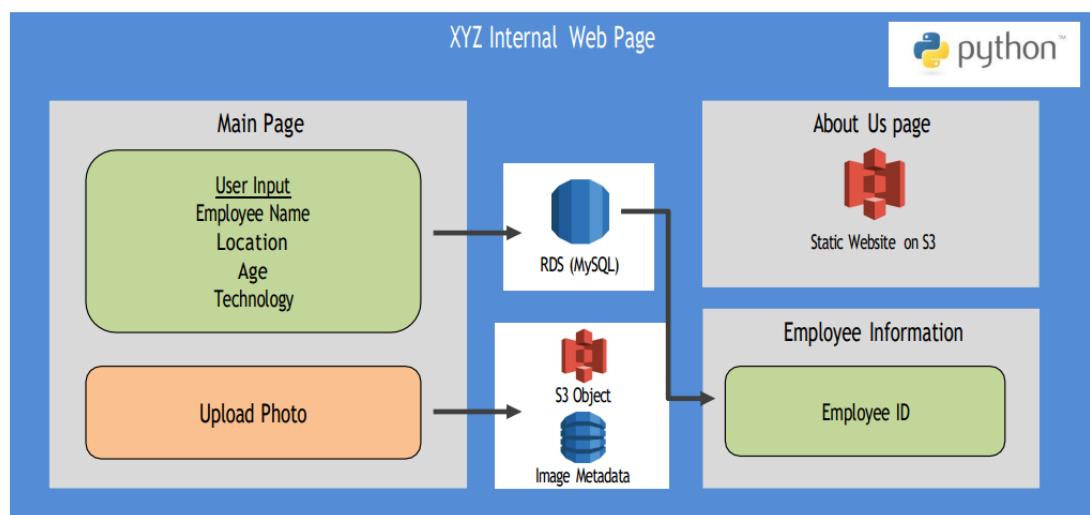
Employee profile of XYZ company – New employees input their information and upload photos. Existing employees can get their information.



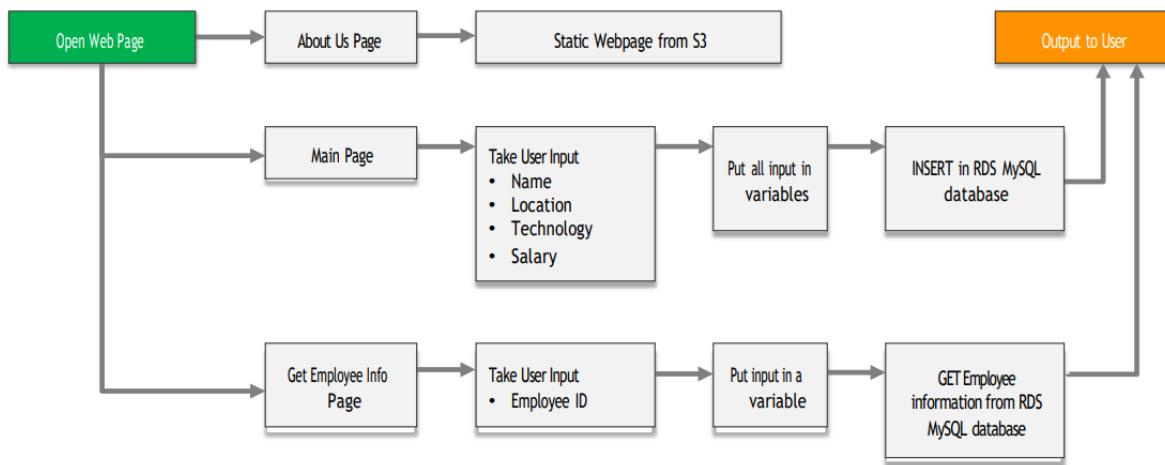
AWS Setup

- Setup VPC for Load Balancer, Application EC2 instance and RDS Database - one public and two private subnets.
- Create Load balancer and Auto Scaling Group.
- Create RDS DB instance and DynamoDB table.
- S3 Bucket
- Get domain name and map it with Load Balancer
- Create instance profile that has to be attached to the EC2 instances being launched. Instance profile should have permission to access RDS, DynamoDB and S3 bucket.
- Lambda function to get triggered when an object is uploaded to the bucket
- SNS Topic for Lambda and user email id subscription

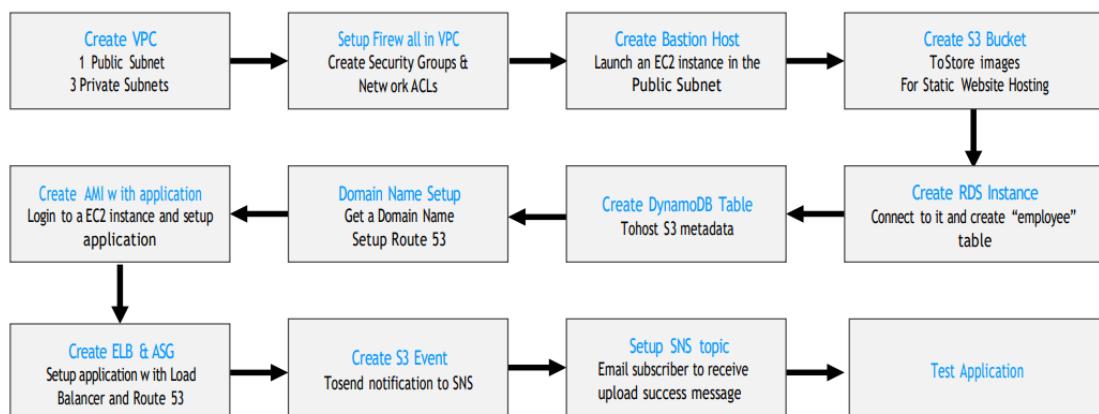
Application Architecture



Application Logic

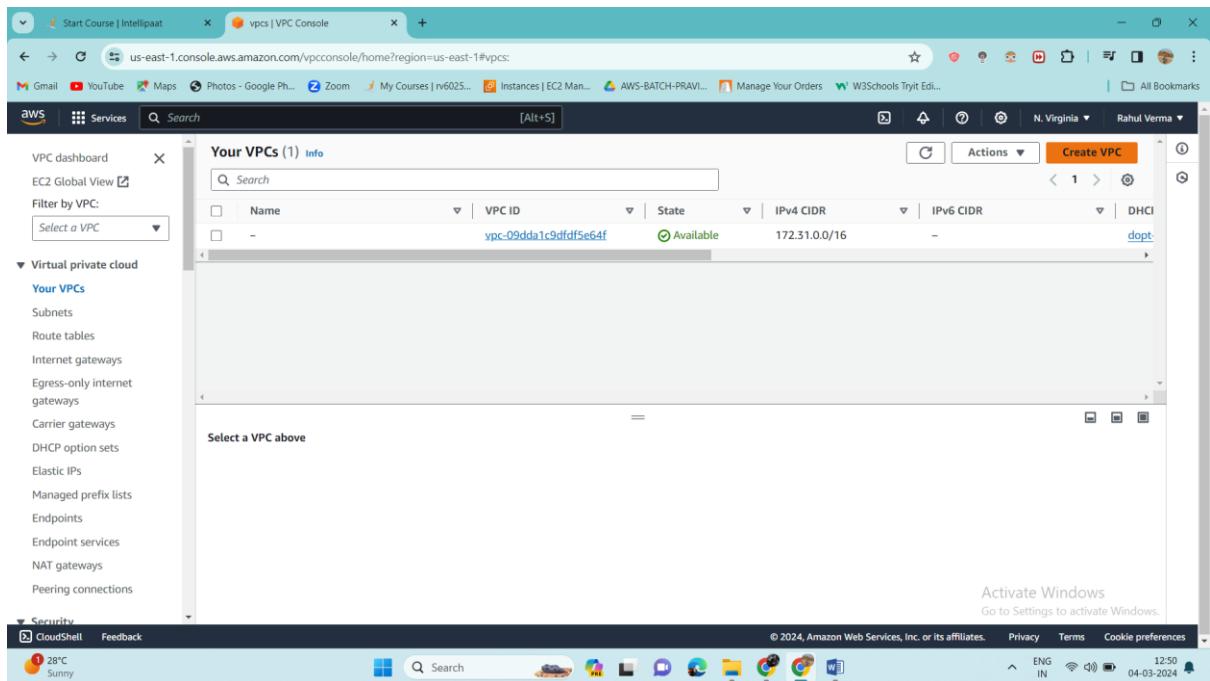


Steps



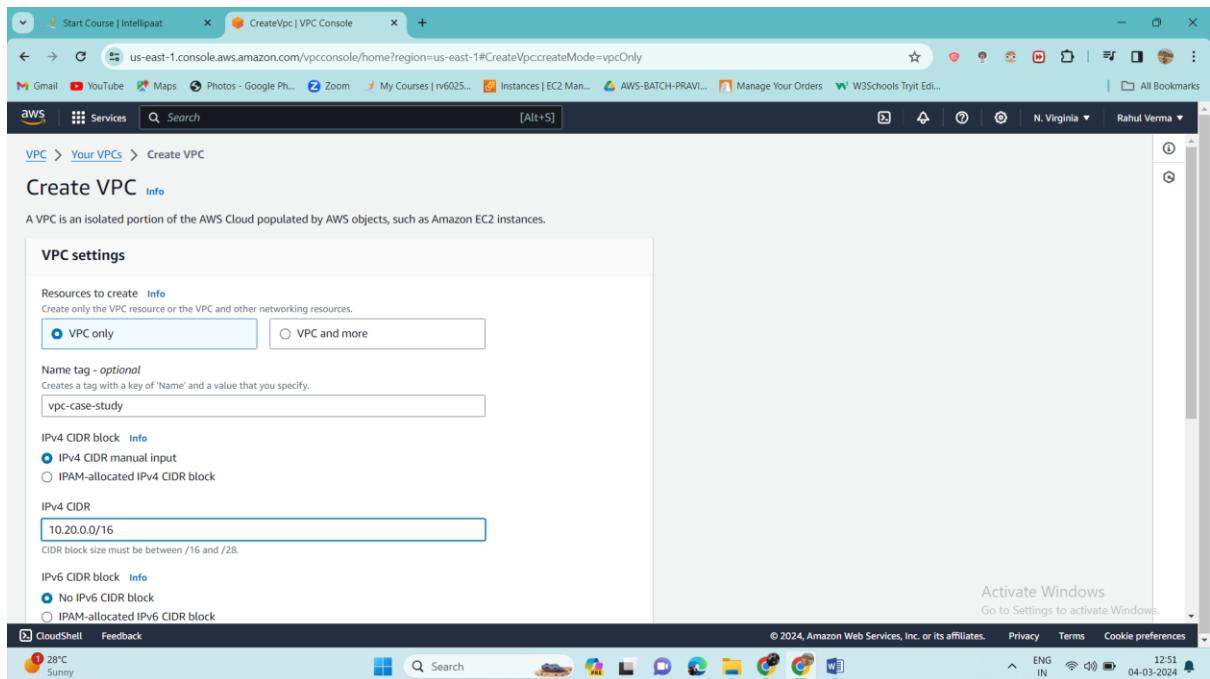
Activate Wincc
Go to Settings to

Let's create VPC according to the steps



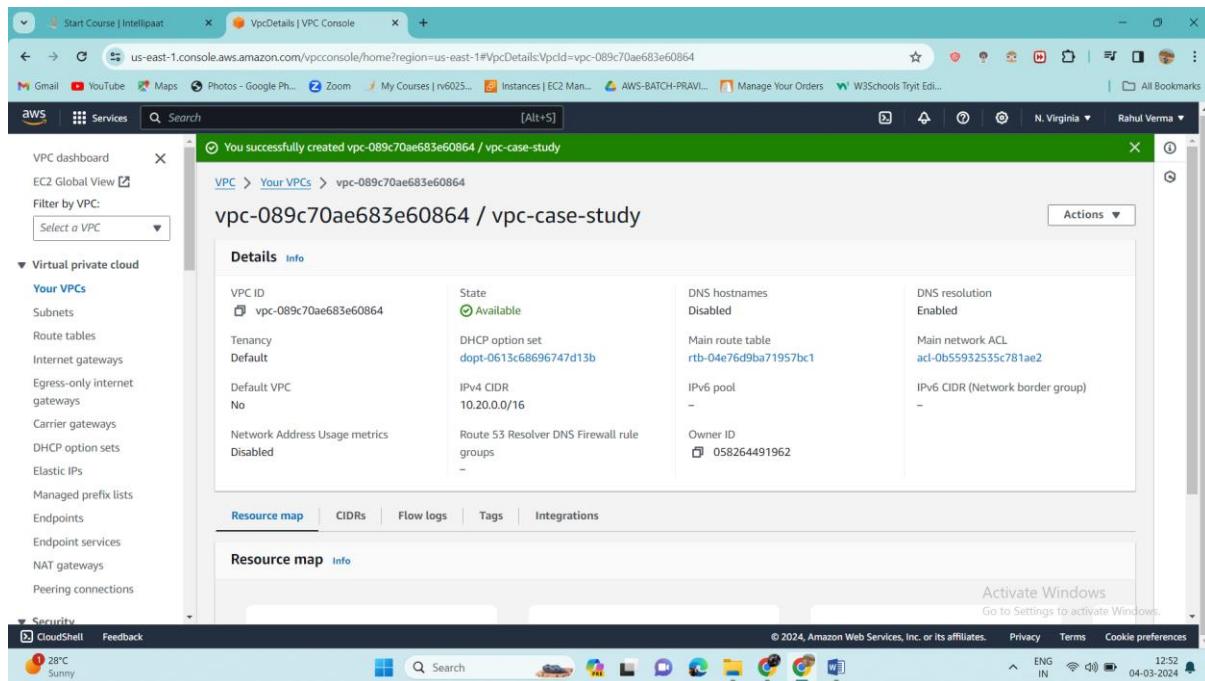
The screenshot shows the AWS VPC Console interface. On the left, a sidebar lists various VPC-related services: VPC dashboard, EC2 Global View, Filter by VPC, Virtual private cloud (with sub-options like Your VPCs, Subnets, Route tables, Internet gateways, Egress-only Internet gateways, Carrier gateways, DHCP option sets, Elastic IPs, Managed prefix lists, Endpoints, Endpoint services, NAT gateways, and Peering connections), and Security (CloudShell, Feedback). The main content area is titled "Your VPCs (1) Info" and displays a table with one row. The table columns are Name, VPC ID, State, IPv4 CIDR, IPv6 CIDR, and DHCP. The single entry is "vpc-09dd1c9df5e64f" with "Available" status, "172.31.0.0/16" IPv4 CIDR, and "-" IPv6 CIDR. A "Create VPC" button is located in the top right of the table area. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

VPC name- "vpc-case-study" and CIDR range- 10.20.0.0/16



The screenshot shows the "Create VPC" wizard in the AWS VPC Console. The current step is "VPC settings". The "Resources to create" section has "VPC only" selected. A "Name tag - optional" field contains "vpc-case-study". The "IPv4 CIDR block" section has "IPv4 CIDR manual input" selected and "10.20.0.0/16" entered. The "IPv6 CIDR block" section has "No IPv6 CIDR block" selected. The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

Our VPC is created now



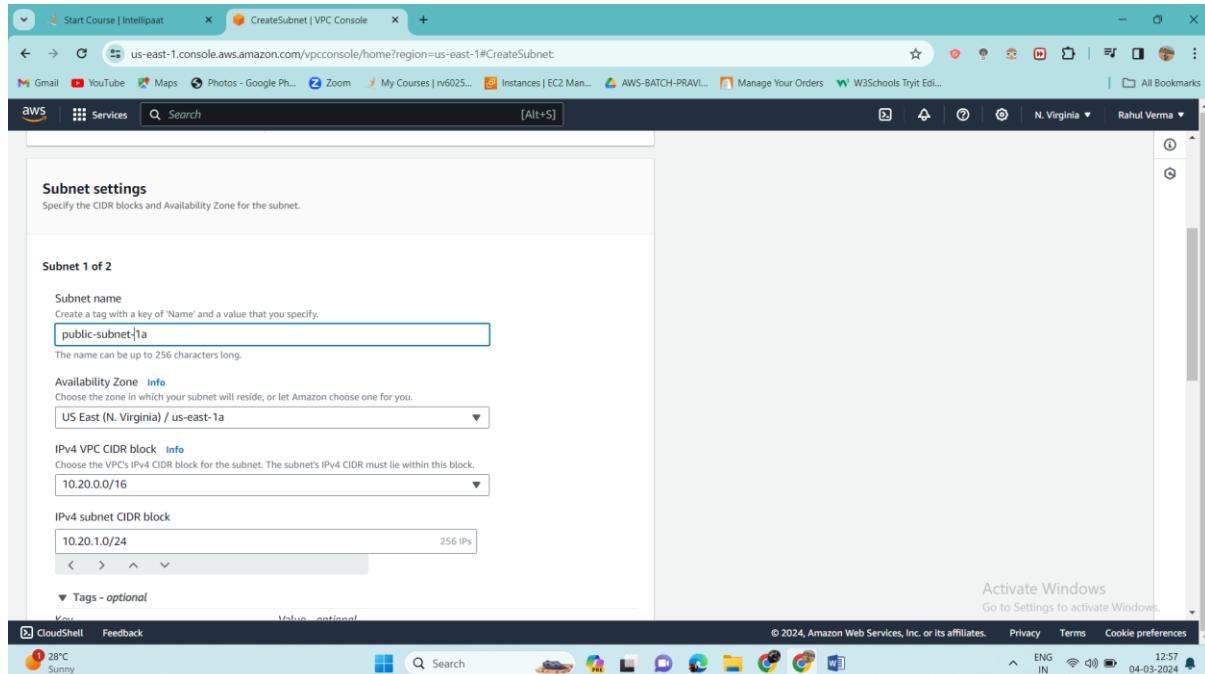
The screenshot shows the AWS VPC Console with a success message: "You successfully created **vpc-089c70ae683e60864** / **vpc-case-study**". The main pane displays the "Details" tab for the VPC, showing the following information:

VPC ID	State	DNS hostnames	DNS resolution
vpc-089c70ae683e60864	Available	Disabled	Enabled
Tenancy	DHCP option set	Main route table	Main network ACL
Default	dopt-0613c68696747d13b	rtb-04e76d9ba71957bc1	acl-0b55932535c781ae2
Default VPC	IPv4 CIDR	IPv6 pool	IPv6 CIDR (Network border group)
No	10.20.0.0/16	—	—
Network Address Usage metrics	Route 53 Resolver DNS Firewall rule groups	Owner ID	—
Disabled	—	058264491962	—

Below the details, there are tabs for "Resource map", "CIDRs", "Flow logs", "Tags", and "Integrations". The "Resource map" tab is selected. The bottom of the screen shows the AWS navigation bar and system status.

Now let's create subnets for our VPC

First I'm creating public subnet named as "**public-subnet-1a**" in us-east-1a availability zone with 10.20.1.0/24 CIDR range.

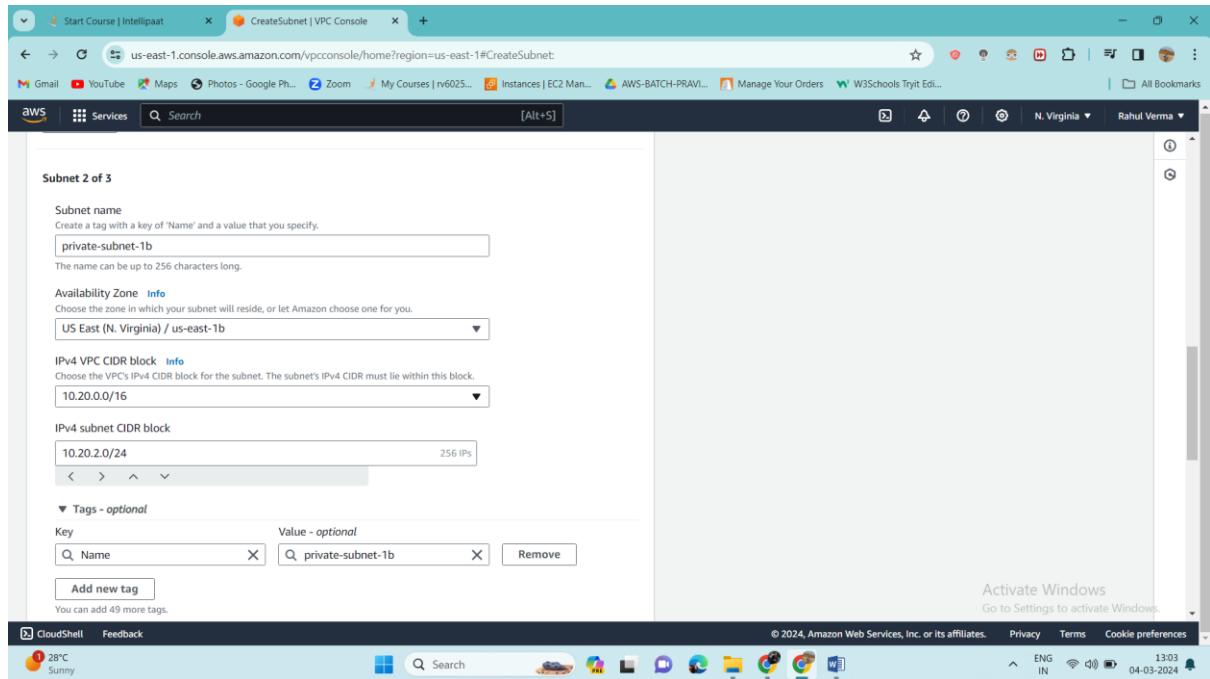


The screenshot shows the "CreateSubnet" page in the AWS VPC Console. The "Subnet settings" section is open, showing the configuration for "Subnet 1 of 2". The configuration includes:

- Subnet name:** public-subnet-1a
- Availability Zone:** US East (N. Virginia) / us-east-1a
- IPv4 VPC CIDR block:** 10.20.0.0/16
- IPv4 subnet CIDR block:** 10.20.1.0/24 (256 IPs)
- Tags - optional:** None

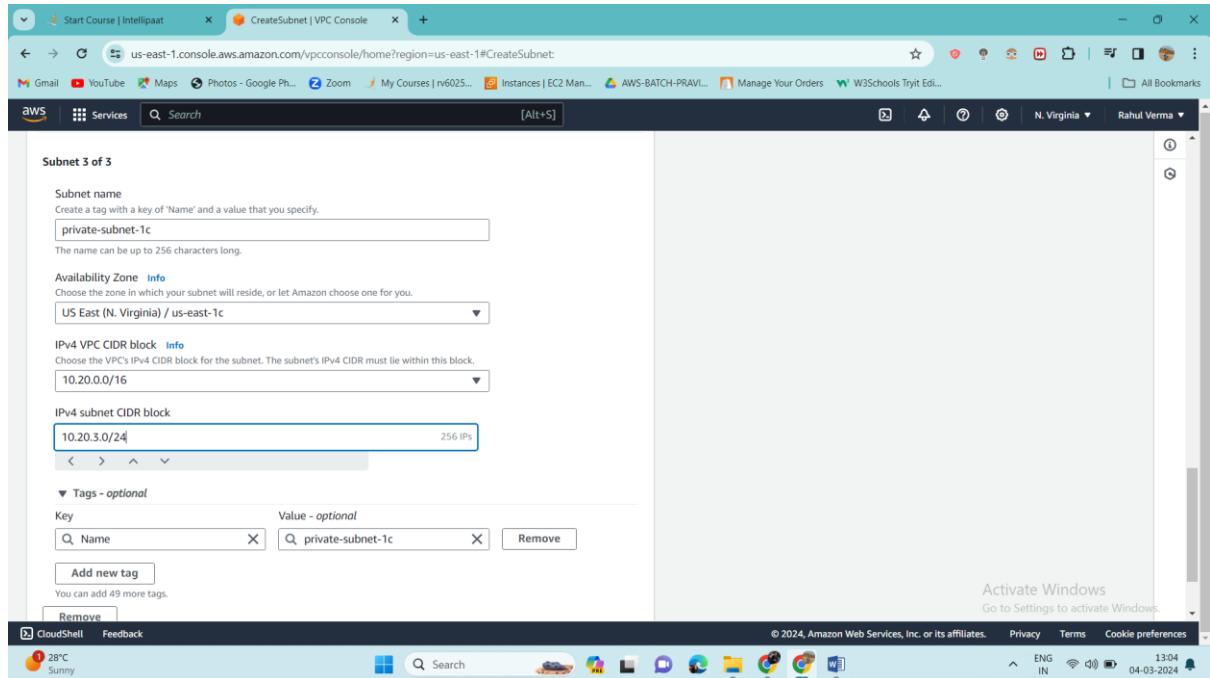
The bottom of the screen shows the AWS navigation bar and system status.

Now I'm creating 3 private subnets first private subnet will be named as "**private-subnet-1b**" in us-east-1b availability zone with 10.20.2.0/24 CIDR range.



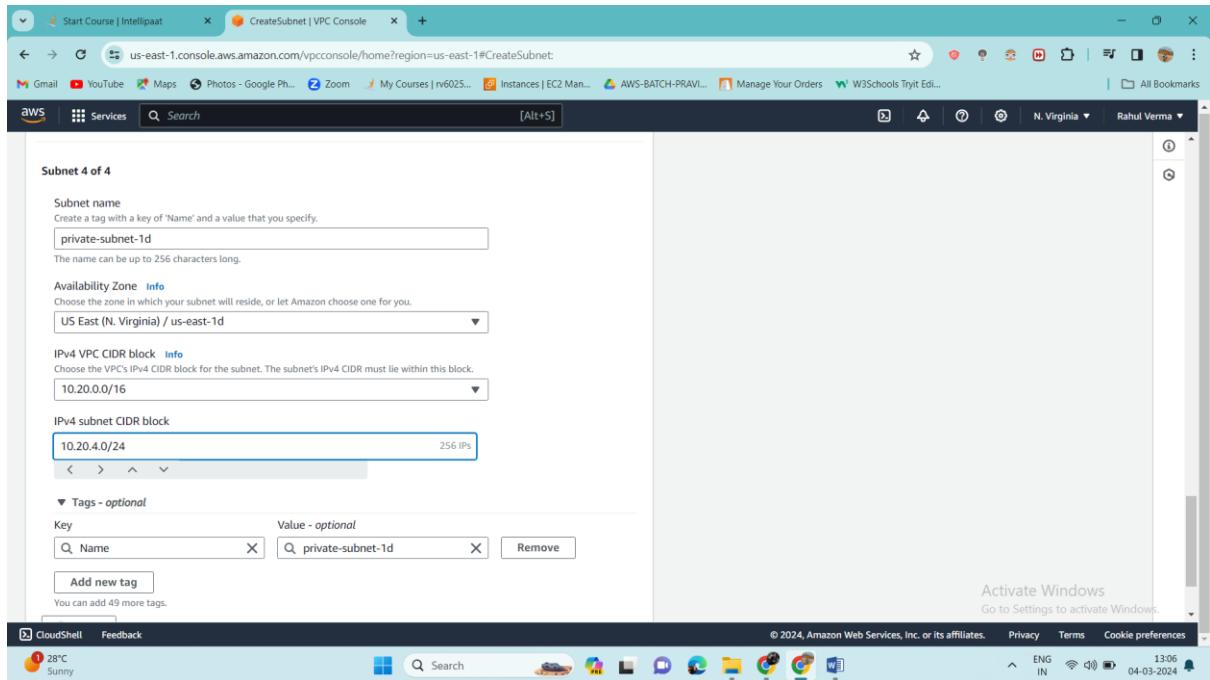
The screenshot shows the AWS VPC Console interface for creating a new subnet. The subnet is named "private-subnet-1b" and is located in the "US East (N. Virginia) / us-east-1b" availability zone. The IPv4 subnet CIDR block is set to 10.20.2.0/24. A tag named "Name" is added with the value "private-subnet-1b". The interface includes a "Tags - optional" section and a "Activate Windows" message at the bottom right.

Second subnet would be "**private-subnet-1c**" in us-east-1c availability zone with 10.20.3.0/24 CIDR range

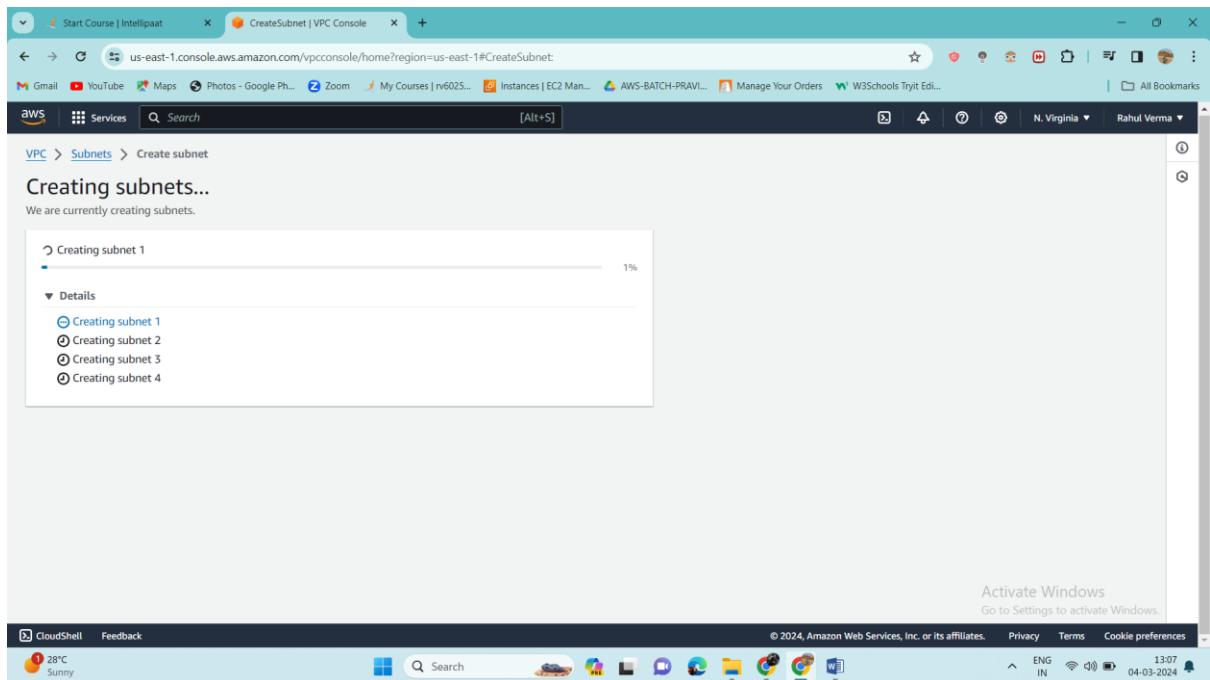


The screenshot shows the AWS VPC Console interface for creating a new subnet. The subnet is named "private-subnet-1c" and is located in the "US East (N. Virginia) / us-east-1c" availability zone. The IPv4 subnet CIDR block is set to 10.20.3.0/24. A tag named "Name" is added with the value "private-subnet-1c". The interface includes a "Tags - optional" section and a "Activate Windows" message at the bottom right.

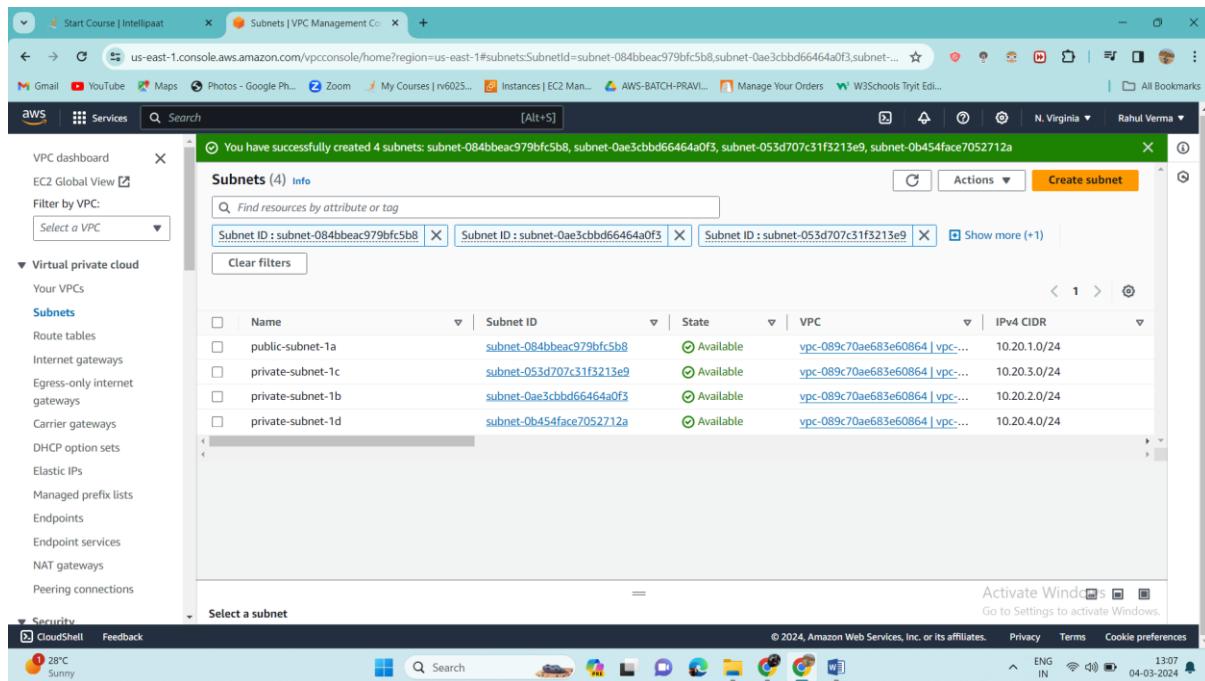
Last private subnet would be “**private-subnet-1d**” in us-east-1d availability zone with 10.20.4.0/24 CIDR range



Now let's click it on create button



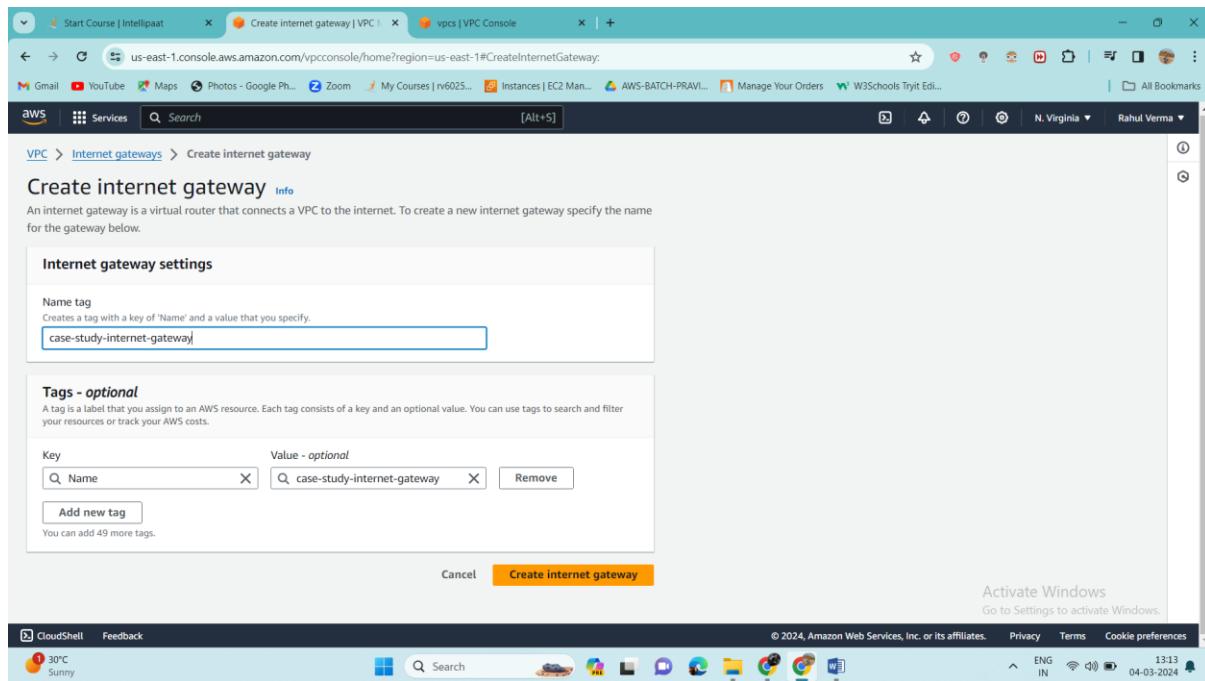
Our subnets are created now



The screenshot shows the AWS VPC Management Console with the 'Subnets' section selected. A message at the top indicates that four subnets have been successfully created. The table below lists the subnets with their details:

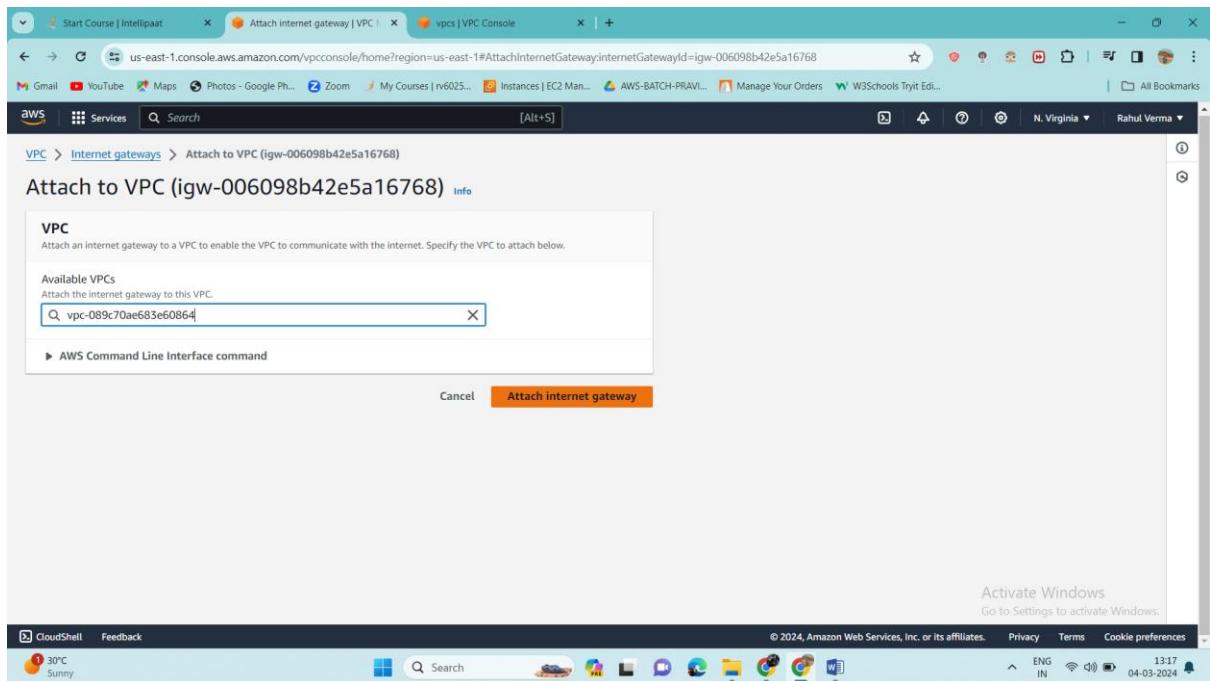
Name	Subnet ID	State	VPC	IPv4 CIDR
public-subnet-1a	subnet-084bbeac979bfc5b8	Available	vpc-089c70ae683e60864 vpc...	10.20.1.0/24
private-subnet-1c	subnet-055d707c31f5213e9	Available	vpc-089c70ae683e60864 vpc...	10.20.3.0/24
private-subnet-1b	subnet-0ae3cbb66464a0f3	Available	vpc-089c70ae683e60864 vpc...	10.20.2.0/24
private-subnet-1d	subnet-0b454face7052712a	Available	vpc-089c70ae683e60864 vpc...	10.20.4.0/24

Now im gonna create one internet gateway- “case-study-internet-gateway”



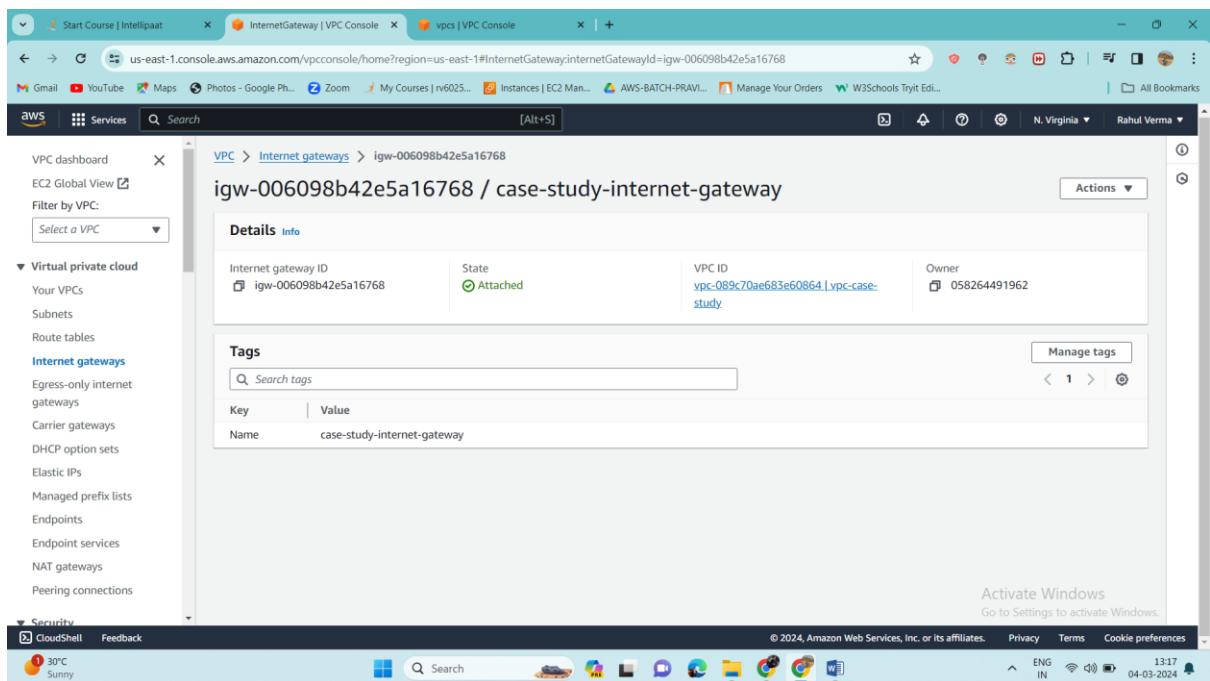
The screenshot shows the 'Create internet gateway' wizard. The first step, 'Internet gateway settings', is selected. It includes a 'Name tag' field containing 'case-study-internet-gateway' and a 'Tags - optional' section with a single tag 'Name: case-study-internet-gateway'. The 'Create internet gateway' button is at the bottom.

Now let's attach this internet gateway to our VPC



The screenshot shows the AWS VPC console with the 'Attach to VPC' dialog open. The URL in the browser is us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#AttachInternetGateway:internetGatewayId=igw-006098b42e5a16768. The dialog has a search bar with 'vpc-089c70ae683e60864' and an 'Attach internet gateway' button.

It's attached to our VPC now



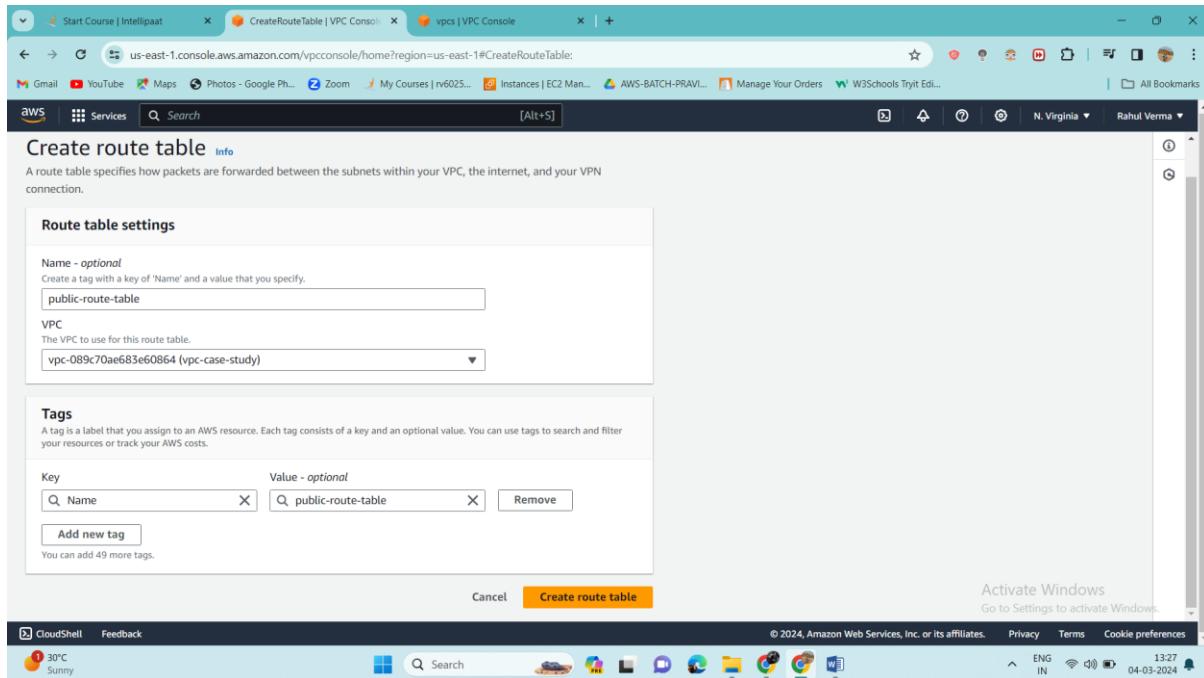
The screenshot shows the AWS VPC console with the details of the attached internet gateway. The URL in the browser is us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#InternetGateway:internetGatewayId=igw-006098b42e5a16768. The details table shows:

Internet gateway ID	igw-006098b42e5a16768	State	Attached
VPC ID	vpc-089c70ae683e60864	Owner	058264491962

The 'Tags' section shows a single tag: Name: case-study-internet-gateway.

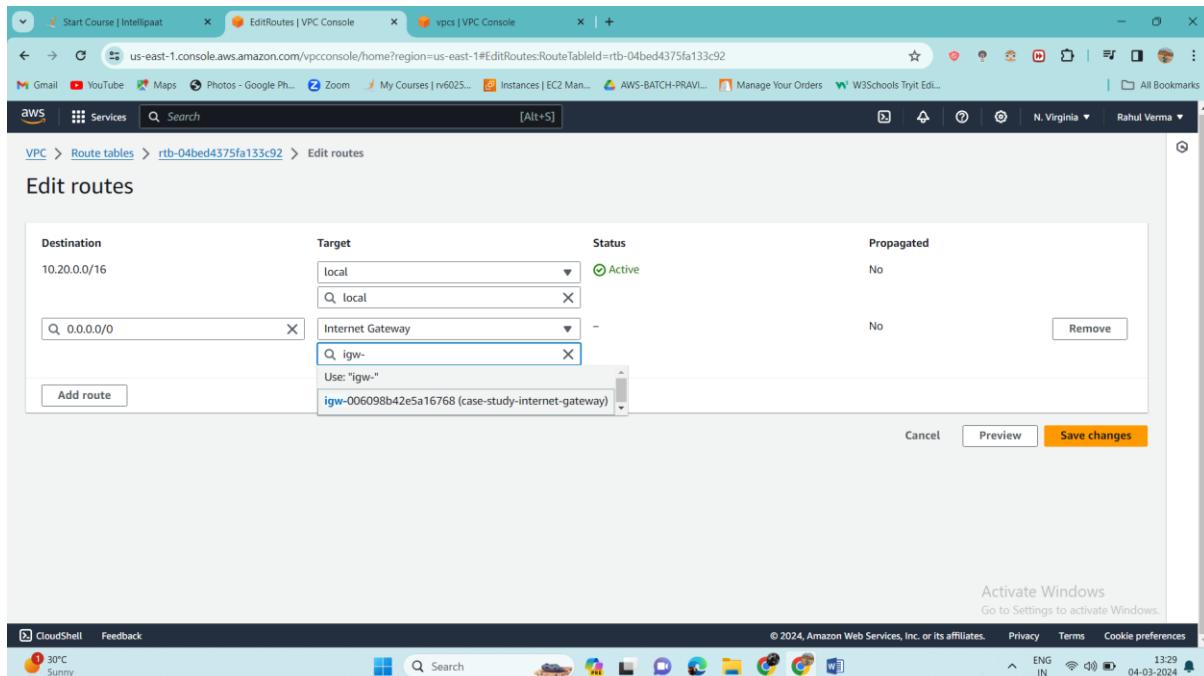
Now we are going to create Route tables

First will create “public-route-table”



The screenshot shows the 'Create route table' wizard in the AWS VPC console. The first step, 'Route table settings', is displayed. It includes fields for 'Name - optional' (set to 'public-route-table'), 'VPC' (set to 'vpc-089c70ae683e60864 (vpc-case-study)'), and 'Tags'. A single tag 'Name: public-route-table' is added. The 'Create route table' button is highlighted in orange at the bottom.

As it is public-route-table it should have internet gateway. So will go to edit routs and do the following changes

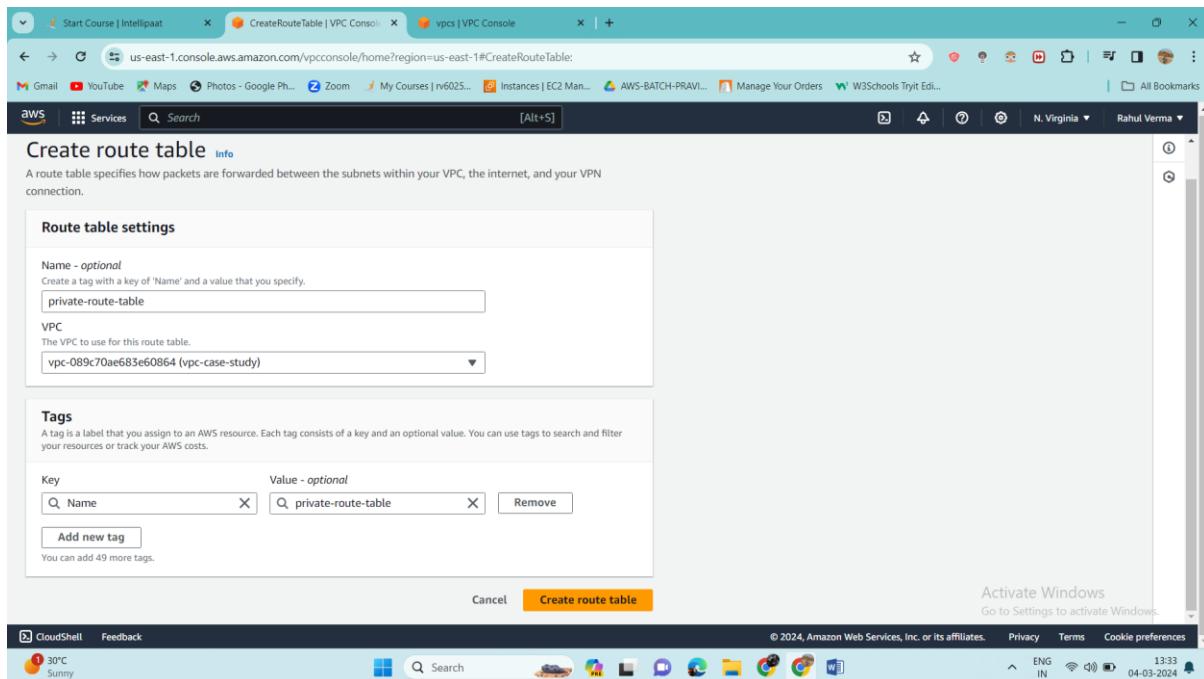


The screenshot shows the 'Edit routes' wizard in the AWS VPC console. It lists a single route for 'Destination 10.20.0.0/16' with 'Target' set to 'local' and 'Status' as 'Active'. A new route is being added for 'Destination 0.0.0.0/0' with 'Target' set to 'Internet Gateway' and 'Status' as 'Active'. The target dropdown shows 'igw-006098b42e5a16768 (case-study-internet-gateway)' selected. The 'Save changes' button is highlighted in orange at the bottom.

Now we will associate our public subnet to our route table. Just go to edit subnet association and select “public-subnet-1a”

And it's done

Now we will create “private-route-table”



Route table settings

Name - optional
Create a tag with a key of 'Name' and a value that you specify.

VPC
The VPC to use for this route table.

Tags

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 or track your AWS costs.

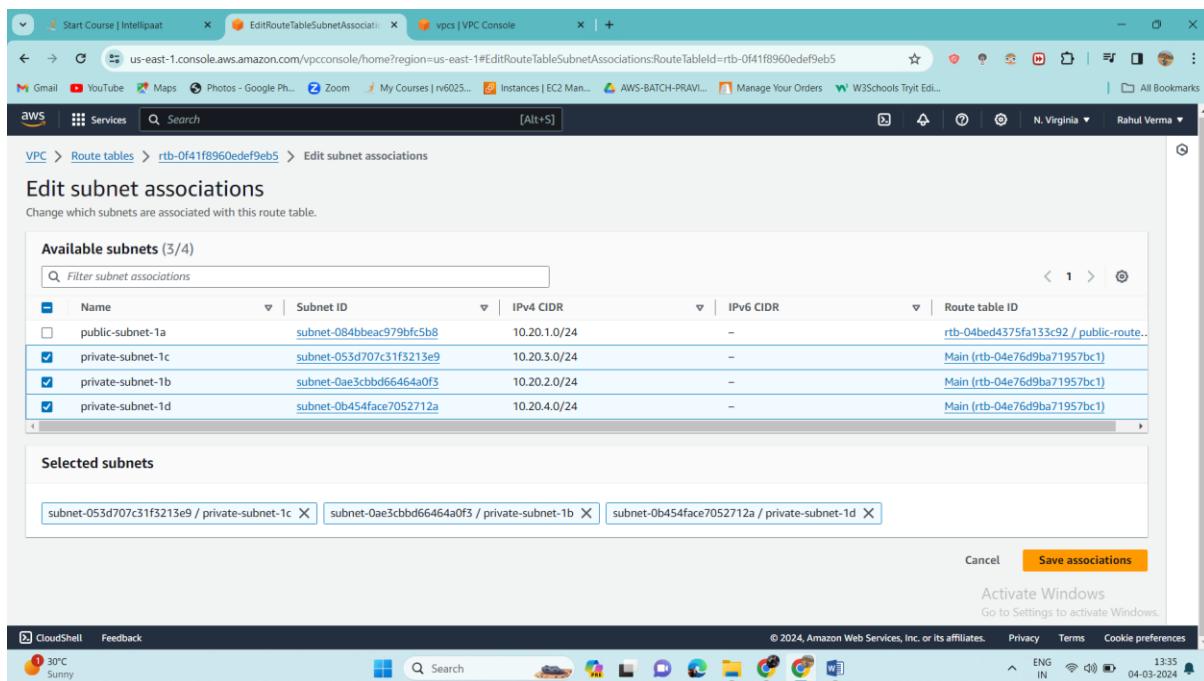
Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="private-route-table"/>

Add new tag

You can add 49 more tags.

Cancel **Create route table**

In this we don't want any internet access so we are going to add our private subnets only



Available subnets (3/4)

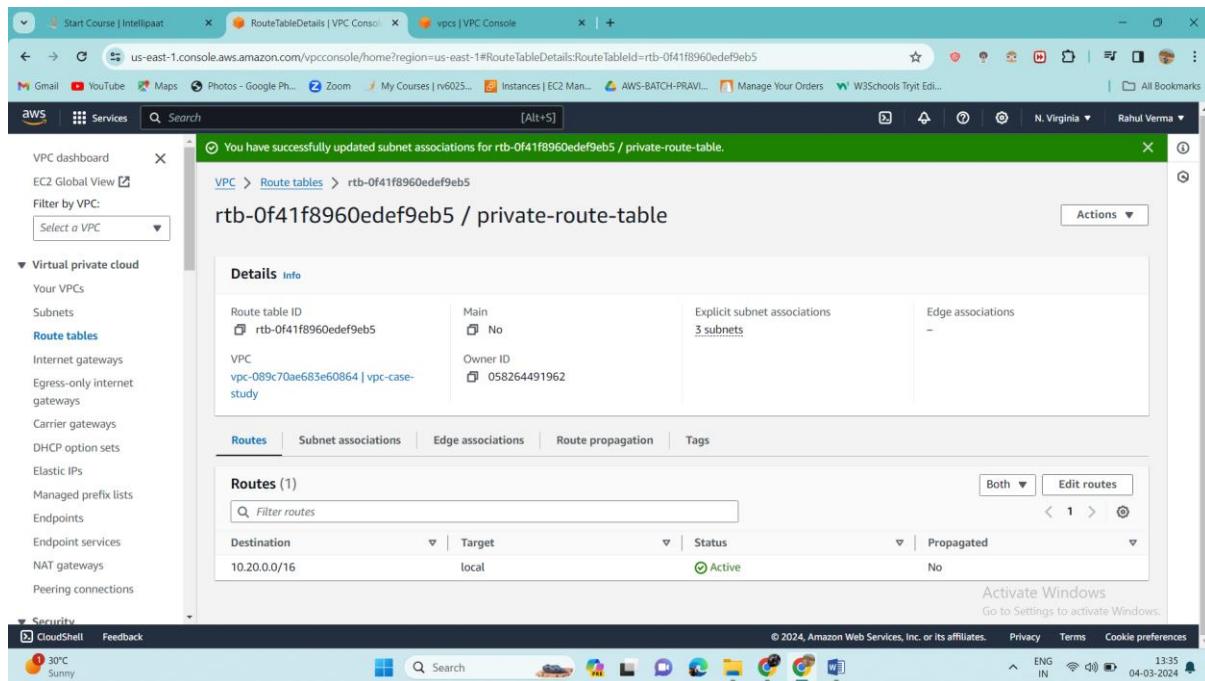
Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
public-subnet-1a	subnet-084bbeac979bfc5b8	10.20.1.0/24	-	rtb-04bed4375fa133c92 / public-route..
<input checked="" type="checkbox"/> private-subnet-1c	subnet-053d707c31f3213e9	10.20.3.0/24	-	Main (rtb-04e76d9ba71957bc1)
<input checked="" type="checkbox"/> private-subnet-1b	subnet-0ae3cbb66464a0f3	10.20.2.0/24	-	Main (rtb-04e76d9ba71957bc1)
<input checked="" type="checkbox"/> private-subnet-1d	subnet-0b454face7052712a	10.20.4.0/24	-	Main (rtb-04e76d9ba71957bc1)

Selected subnets

subnet-053d707c31f3213e9 / private-subnet-1c subnet-0ae3cbb66464a0f3 / private-subnet-1b subnet-0b454face7052712a / private-subnet-1d

Cancel **Save associations**

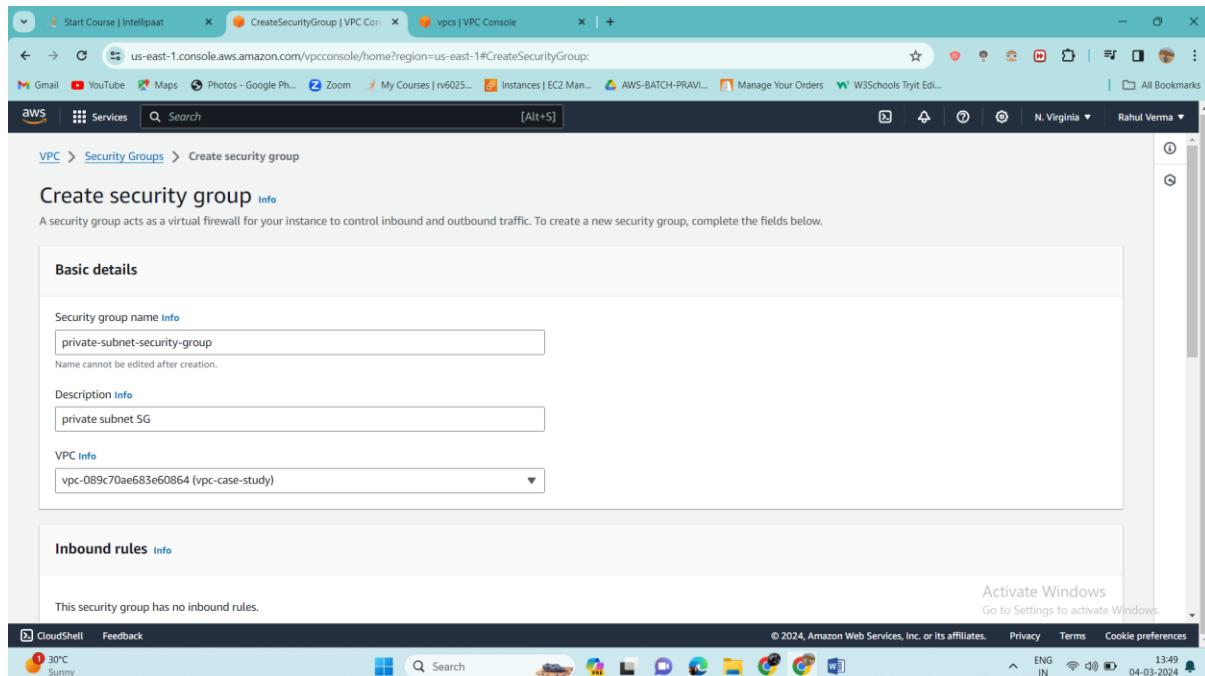
And it's done



The screenshot shows the AWS VPC Console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#RouteTableDetails>. The main content area displays a success message: "You have successfully updated subnet associations for rtb-0f41f8960edef9eb5 / private-route-table." Below this, the "rtb-0f41f8960edef9eb5 / private-route-table" details are shown, including the Route table ID (rtb-0f41f8960edef9eb5), Main status (No), VPC (vpc-089c70ae683e60864 | vpc-case-study), and Owner ID (058264491962). The "Routes" tab is selected, showing one route entry: Destination 10.20.0.0/16, Target local, Status Active, and Propagated No. The bottom of the screen shows the AWS navigation bar and a status bar indicating 30°C, Sunny, and the date 04-03-2024.

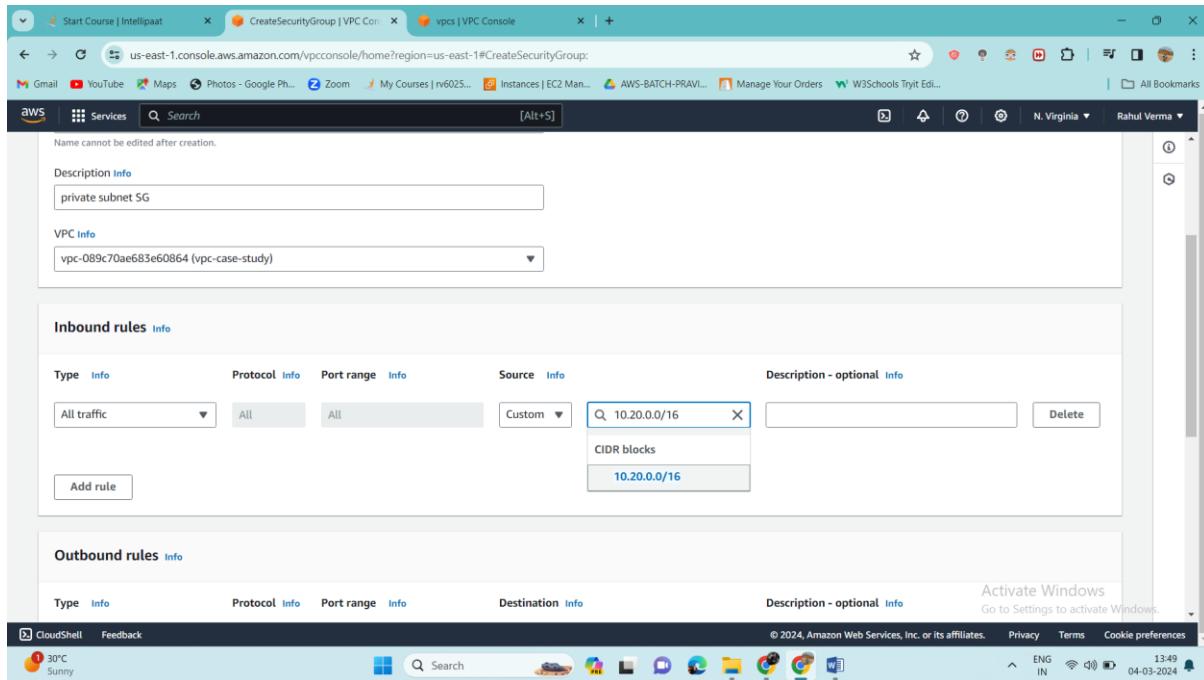
Now our VPC setup is done and now will create security firewall

So first im going to create one security group from this we are going to launch our private-subnet instances.

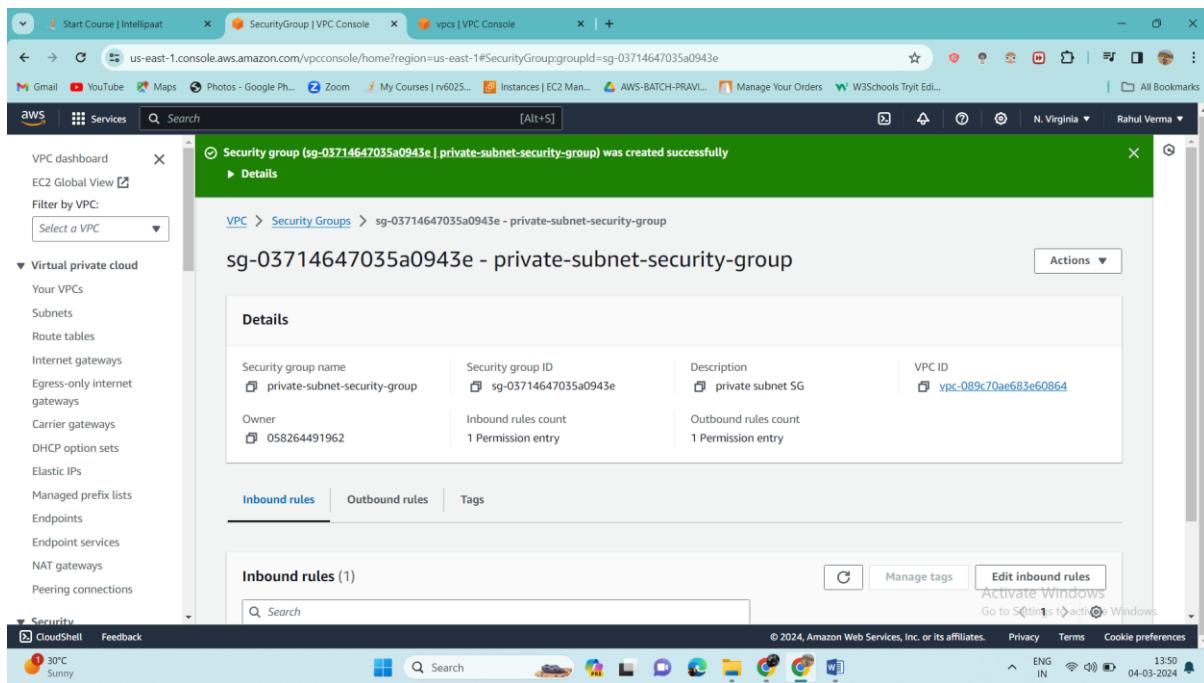


The screenshot shows the AWS VPC Console with the URL <https://us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#CreateSecurityGroup>. The "Create security group" page is displayed, with the sub-header "Create security group info". A note states: "A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below." The "Basic details" section contains fields for "Security group name" (private-subnet-security-group) and "Description" (private subnet SG). The "VPC Info" dropdown is set to "vpc-089c70ae683e60864 (vpc-case-study)". The "Inbound rules" section notes: "This security group has no inbound rules." The bottom of the screen shows the AWS navigation bar and a status bar indicating 30°C, Sunny, and the date 04-03-2024.

And in inbound rules we will allow all traffic to our VPC CIDR so that will be able to connect within our VPC.



The screenshot shows the AWS VPC Console with the 'CreateSecurityGroup' page. The security group is named 'private subnet SG' and is associated with VPC 'vpc-089c70ae683e60864'. The inbound rules table shows a single rule allowing 'All traffic' from 'Custom' source '10.20.0.0/16'. The outbound rules table is empty.

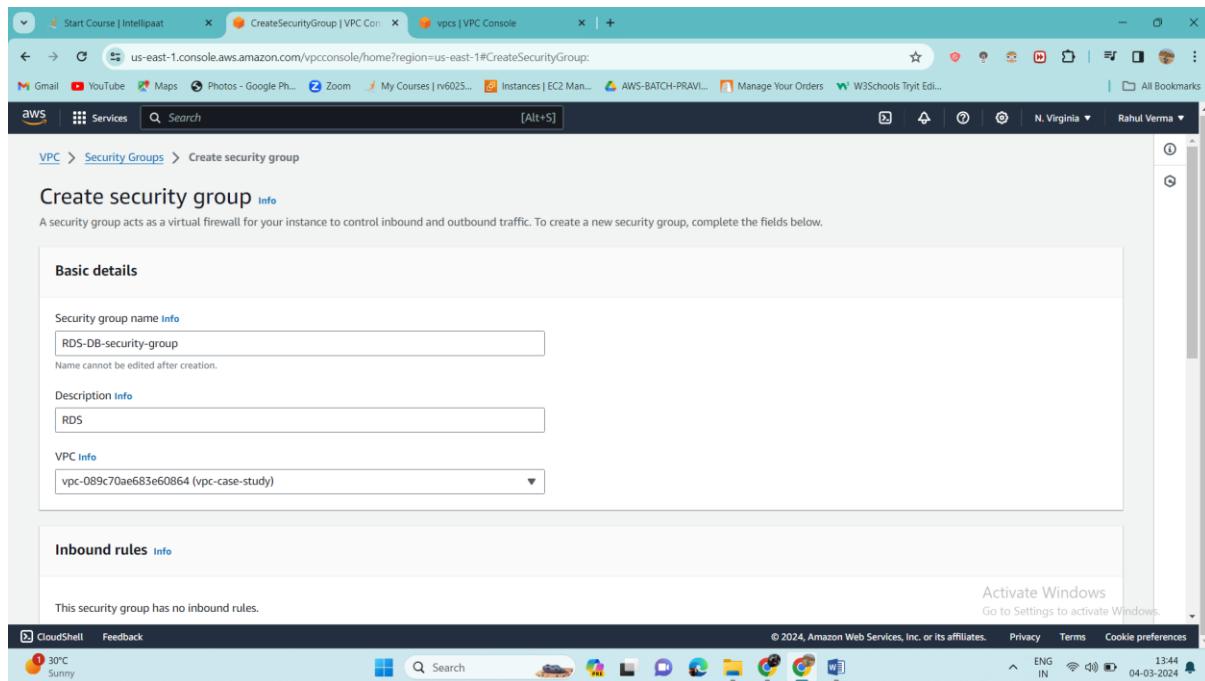


The screenshot shows the AWS VPC Console with the 'SecurityGroup' page. A success message indicates the security group was created successfully. The details page for 'sg-03714647035a0943e - private-subnet-security-group' shows the following information:

Security group name	Security group ID	Description	VPC ID
private-subnet-security-group	sg-03714647035a0943e	private subnet SG	vpc-089c70ae683e60864

The inbound rules section shows one rule allowing traffic from 10.20.0.0/16. The outbound rules section shows one rule allowing traffic to 0.0.0.0/0.

Now next will create security group for our RDS



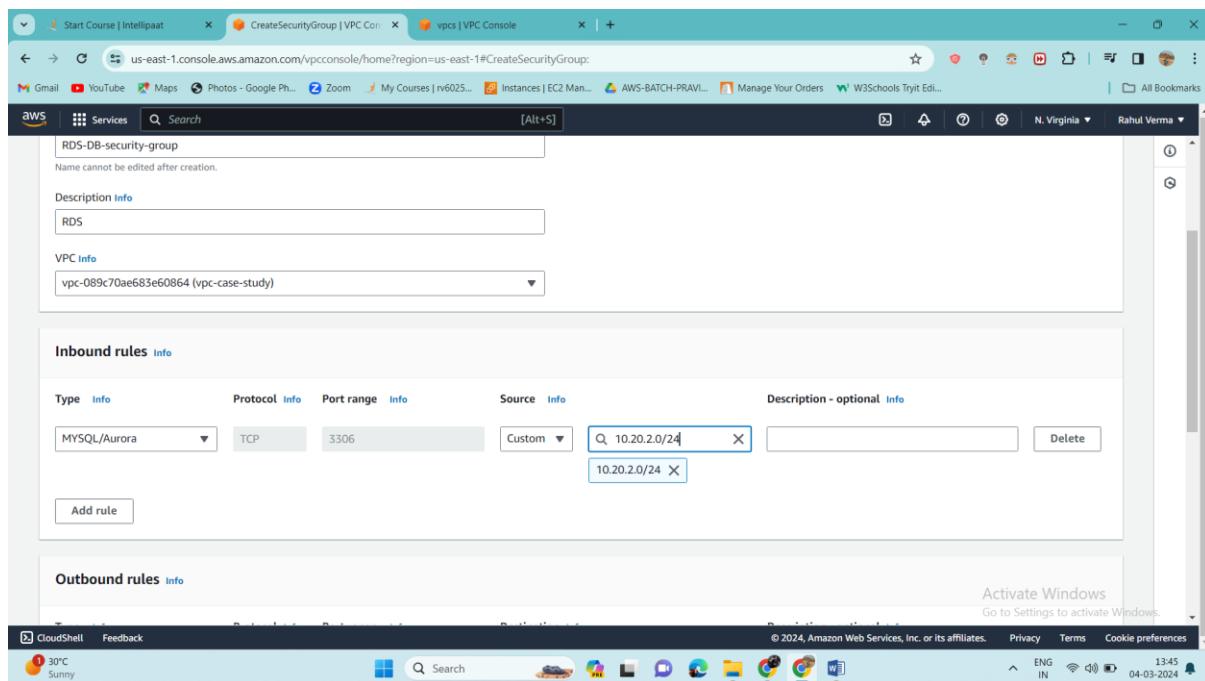
Basic details

Security group name [Info](#)
RDS-DB-security-group

Description [Info](#)
RDS

VPC Info
vpc-089c70ae683e60864 (vpc-case-study)

And in inbound rules will select MySQL/Aurora for 10.20.20./24 CIDR range



Inbound rules [Info](#)

This security group has no inbound rules.

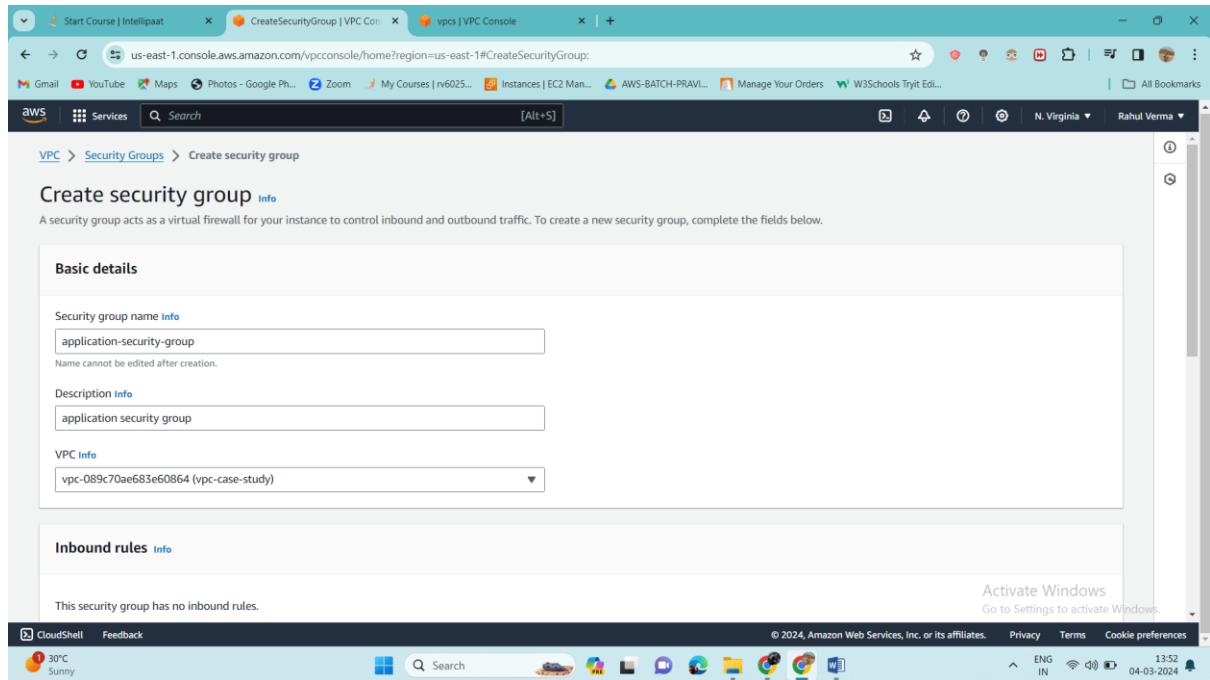
Type [Info](#) Protocol [Info](#) Port range [Info](#) Source [Info](#) Description - optional [Info](#)

MySQL/Aurora [▼](#) TCP [▼](#) 3306 [Custom \[▼\]\(#\)](#) [X](#) [X](#) [Delete](#)

Add rule

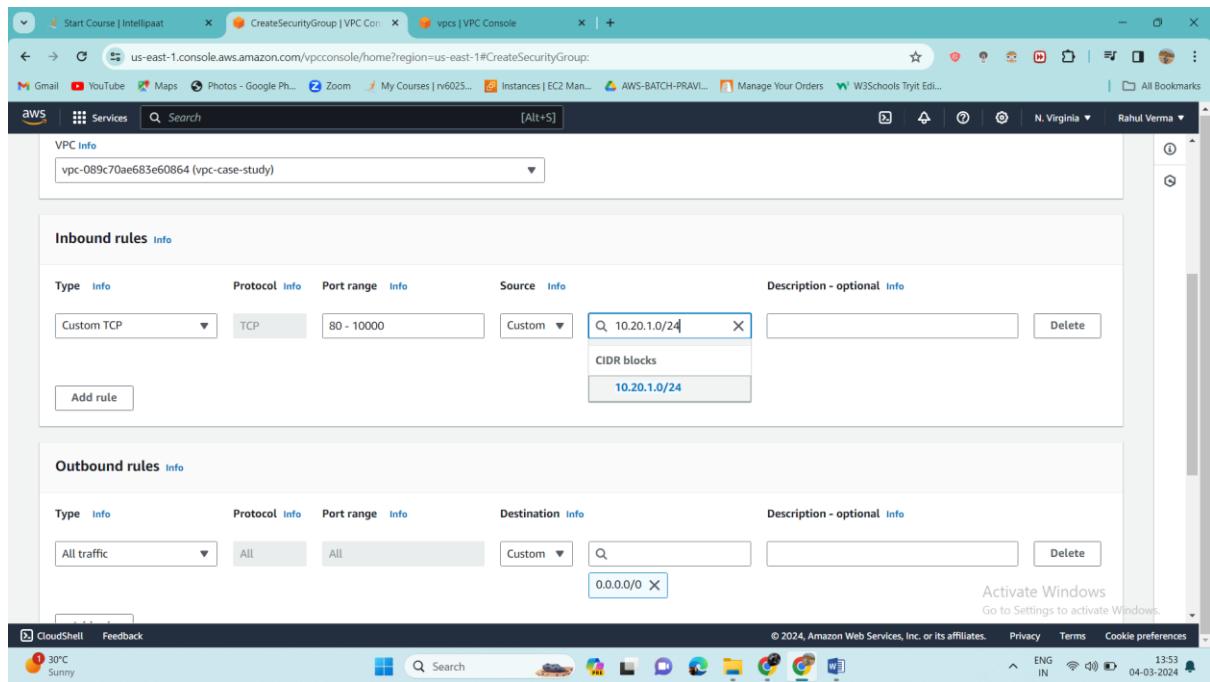
Outbound rules [Info](#)

Now will create one security group for our application subnet where we are going to install our application.



The screenshot shows the 'Create security group' page in the AWS VPC console. In the 'Basic details' section, the security group name is 'application-security-group' and the description is 'application security group'. The VPC is set to 'vpc-089c70ae683e60864 (vpc-case-study)'. In the 'Inbound rules' section, it is noted that the security group has no inbound rules.

And in inbound rules port range 80-10000 and private subnet range only

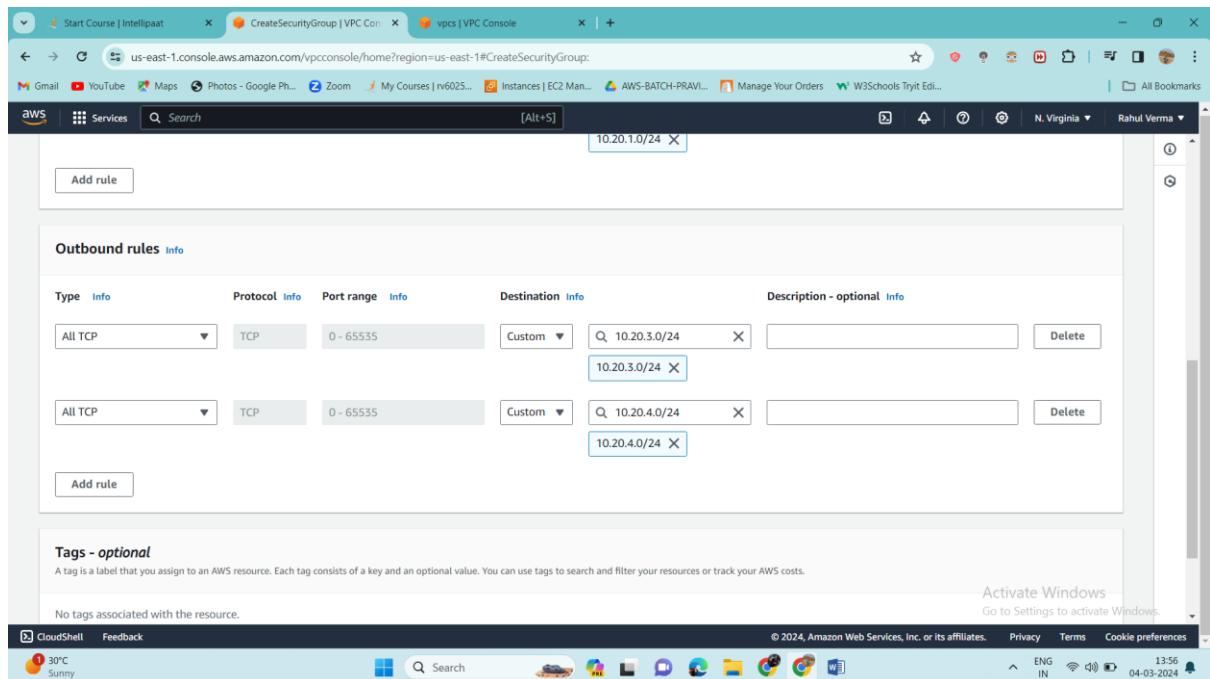


The screenshot shows the 'Create security group' page in the AWS VPC console. In the 'Inbound rules' section, a new rule is being configured with the following details:

Type	Protocol	Port range	Source	Description - optional
Custom TCP	TCP	80 - 10000	Custom	10.20.1.0/24

The 'Source' dropdown shows 'CIDR blocks' with '10.20.1.0/24' selected. The 'Outbound rules' section shows a rule for 'All traffic' with a destination of '0.0.0.0/0'.

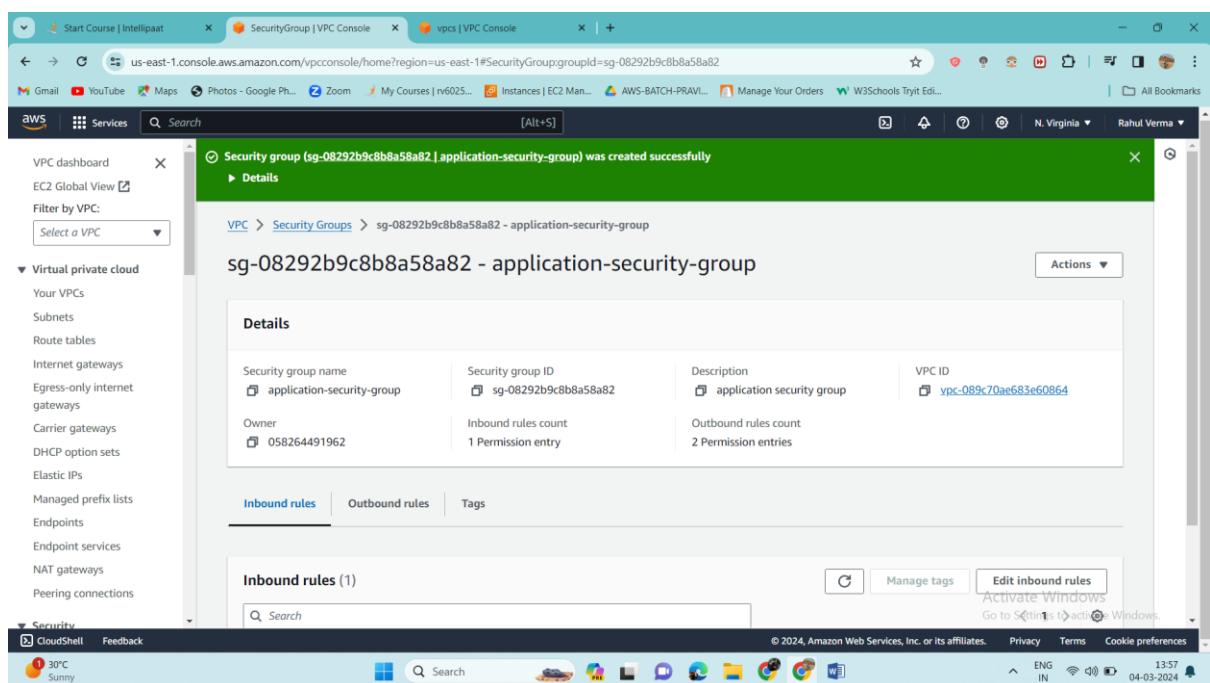
And lastly in our outbound rules All TCP to our Database subnet



The screenshot shows the AWS VPC Console with the 'Outbound rules' section. There are two entries:

Type	Protocol	Port range	Destination	Description
All TCP	TCP	0 - 65535	Custom (10.20.3.0/24)	10.20.3.0/24
All TCP	TCP	0 - 65535	Custom (10.20.4.0/24)	10.20.4.0/24

Below the table is a 'Tags - optional' section with a note: '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 or track your AWS costs.' The status bar at the bottom shows 'CloudShell Feedback' and the date '04-03-2024'.

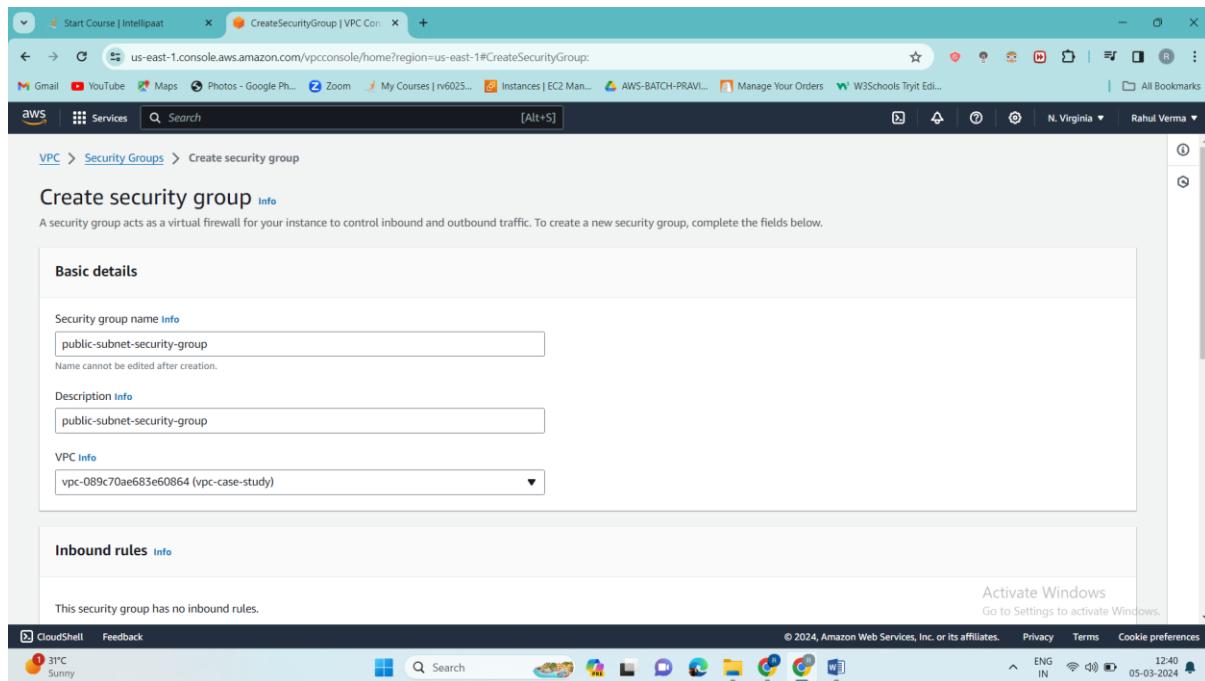


The screenshot shows the AWS VPC Console with a success message: 'Security group (sg-08292b9c8b8a58a82 | application-security-group) was created successfully'. The 'Details' section shows:

Security group name	Security group ID	Description	VPC ID
application-security-group	sg-08292b9c8b8a58a82	application security group	vpc-089c70ae683e60864
Owner	Inbound rules count	Outbound rules count	
058264491962	1 Permission entry	2 Permission entries	

The 'Inbound rules (1)' section shows one rule with a 'Manage tags' and 'Edit inbound rules' button. The status bar at the bottom shows 'CloudShell Feedback' and the date '04-03-2024'.

Now we have to create security group for our web tier which is public subnet

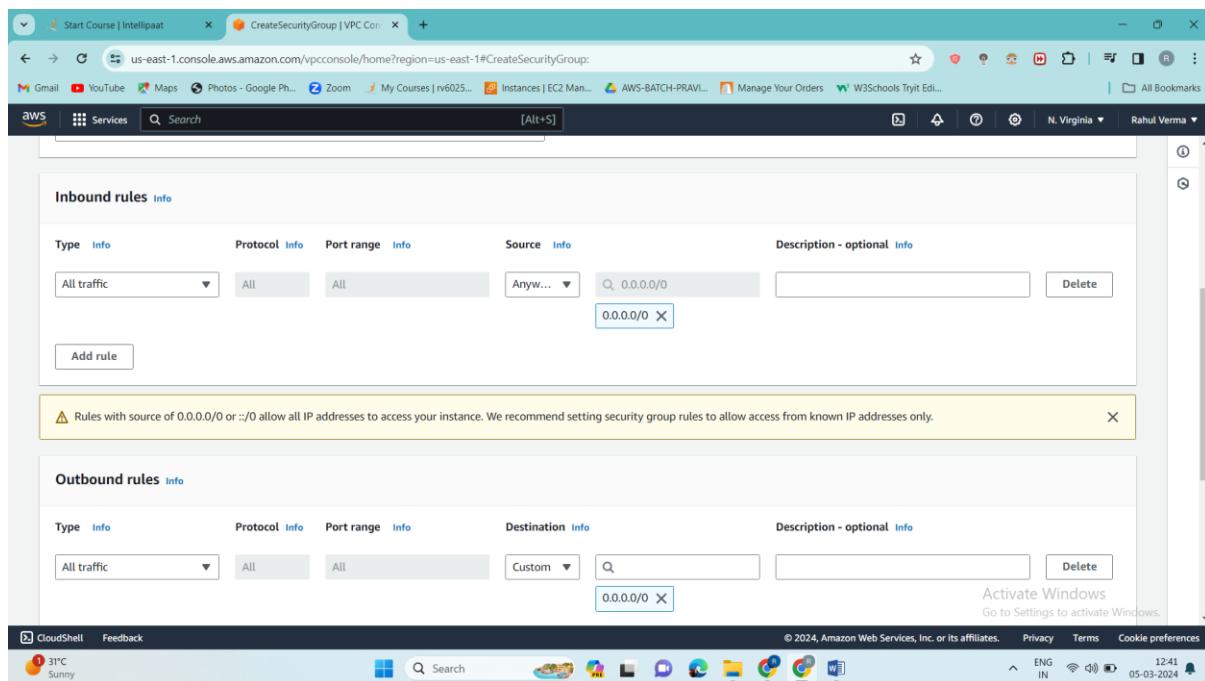


The screenshot shows the 'Create security group' page in the AWS VPC console. The 'Basic details' section is filled with the following information:

- Security group name:** public-subnet-security-group
- Description:** public-subnet-security-group
- VPC:** vpc-089c70ae683e60864 (vpc-case-study)

The 'Inbound rules' section shows a note: "This security group has no inbound rules." Below this, there is a warning message: "⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." The page includes standard AWS navigation and status bars at the bottom.

Inbound rule will change it to All traffic anywhere

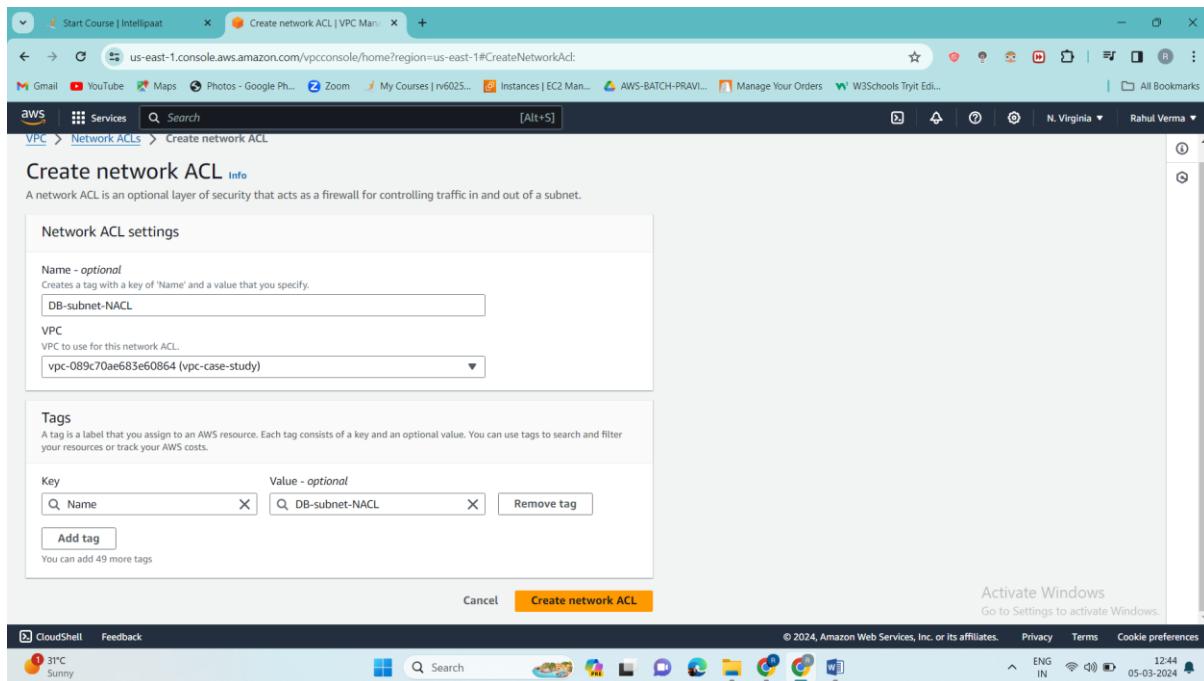


The screenshot shows the 'Create security group' page with the 'Inbound rules' and 'Outbound rules' sections visible. The 'Inbound rules' section is configured with the following settings:

Type	Protocol	Port range	Source	Description - optional
All traffic	All	All	Any... 0.0.0.0/0	

A warning message is displayed: "⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." The 'Outbound rules' section is also visible, showing a single rule with 'All traffic' selected. The page includes standard AWS navigation and status bars at the bottom.

Now we are going to create Network ACL- "DB-subnet-NACL"



Network ACL settings

Name - *optional*
Creates a tag with a key of 'Name' and a value that you specify.

DB-subnet-NACL

VPC
VPC to use for this network ACL.

vpc-089c70ae683e60864 (vpc-case-study)

Tags

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 or track your AWS costs.

Key Value - *optional*

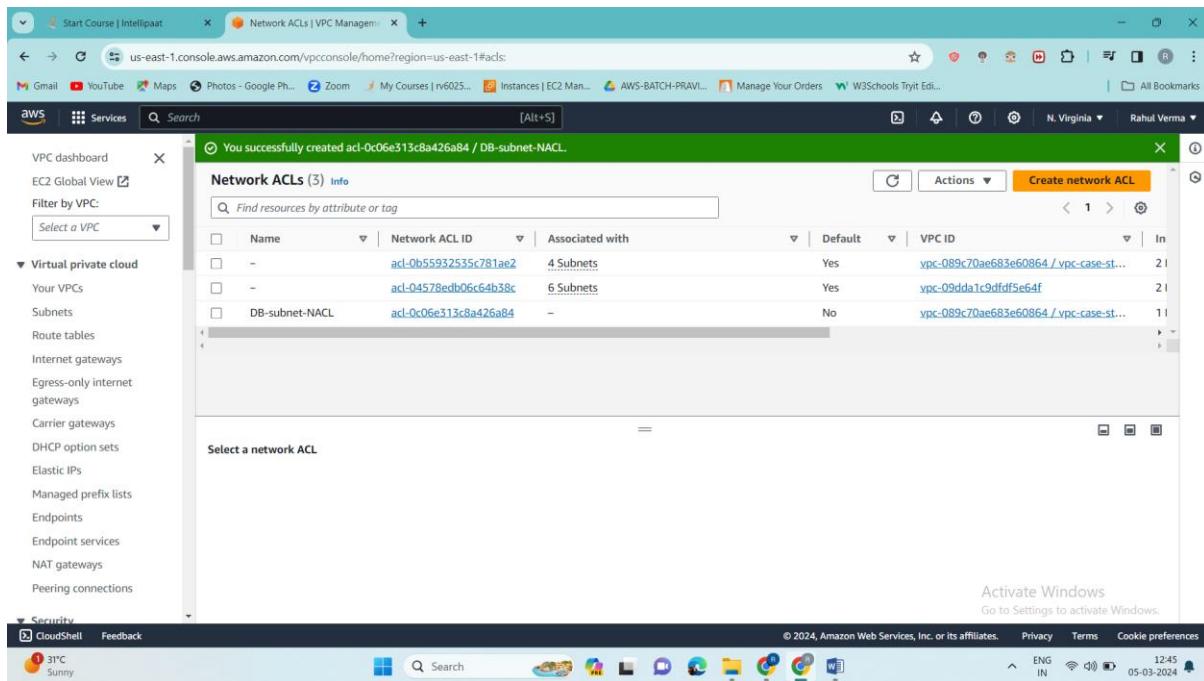
Q Name Q DB-subnet-NACL X Remove tag

Add tag

You can add 49 more tags

Create network ACL

Created



You successfully created acl-0c06e313c8a426a84 / DB-subnet-NACL

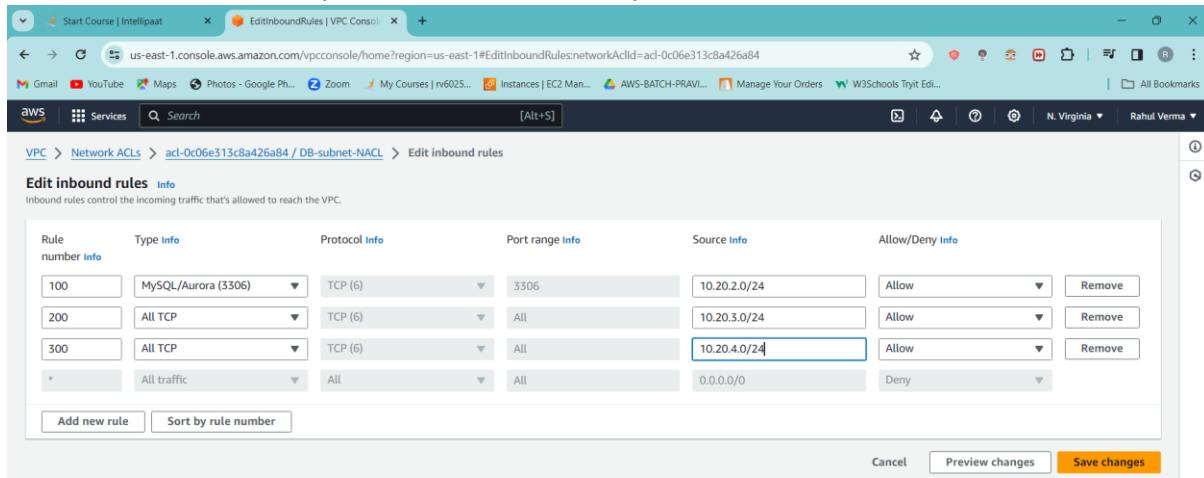
Name	Network ACL ID	Associated with	Default	VPC ID	In
-	acl-0b55932535c781ae2	4 Subnets	Yes	vpc-089c70ae683e60864 / vpc-case-st...	2 1
-	acl-04578edb06c64b38c	6 Subnets	Yes	vpc-09dd1c9df5e64f	2 1
DB-subnet-NACL	acl-0c06e313c8a426a84	-	No	vpc-089c70ae683e60864 / vpc-case-st...	1 1

Select a network ACL

Create network ACL

In it's inbound & will allow MySQL/Aurora to 10.20.2.0/24 CIDR private subnet 1b

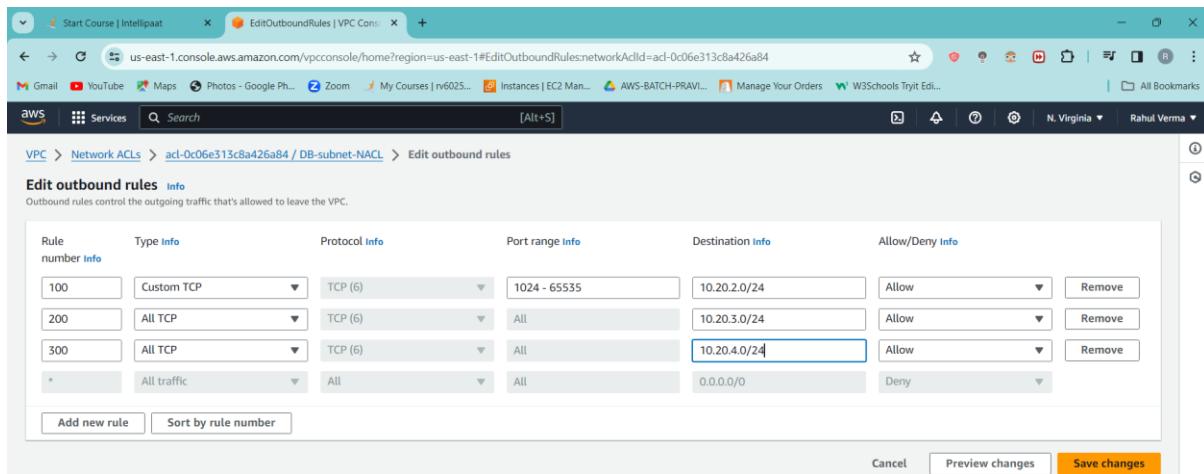
And we have added other private subnet so that they can communicate each other.



Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 31°C Sunny ENG IN 12:55 05-03-2024

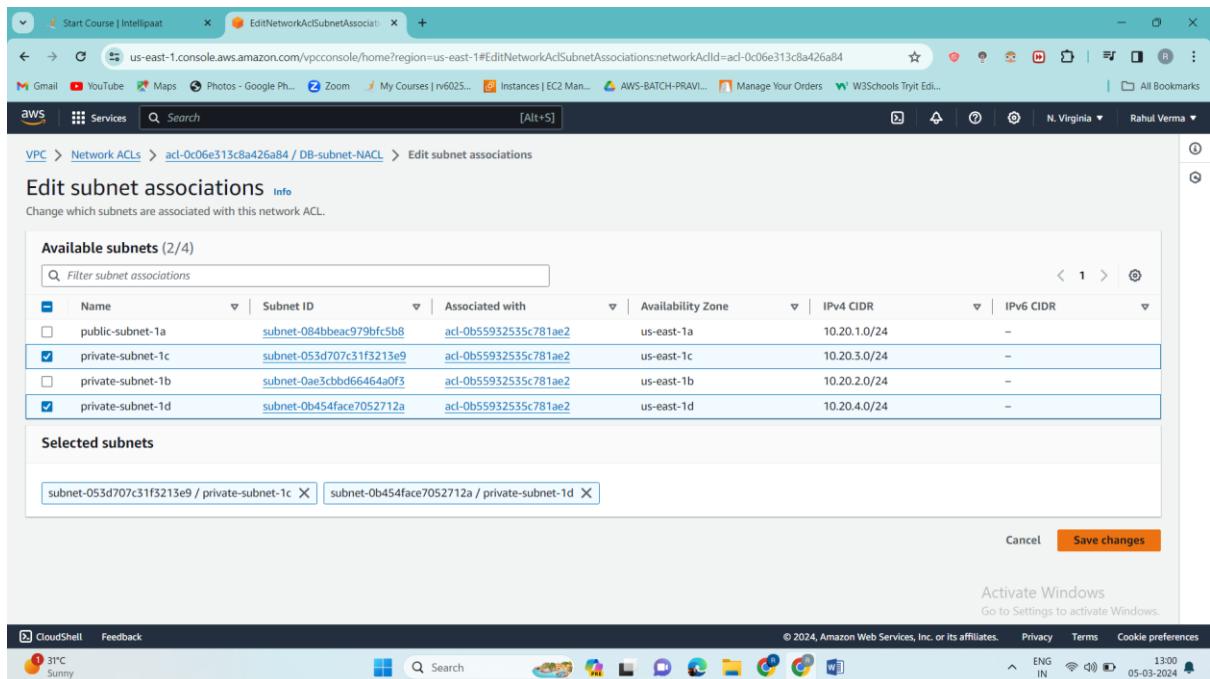
And similarly in outbound rule custom where port range is 1024-65535 with same CIDR range 10.20.2.0/24. And then will add private subnets.



Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 31°C Sunny ENG IN 12:58 05-03-2024

Now let's associate subnets to our Db-subnet-nacl



Available subnets (2/4)

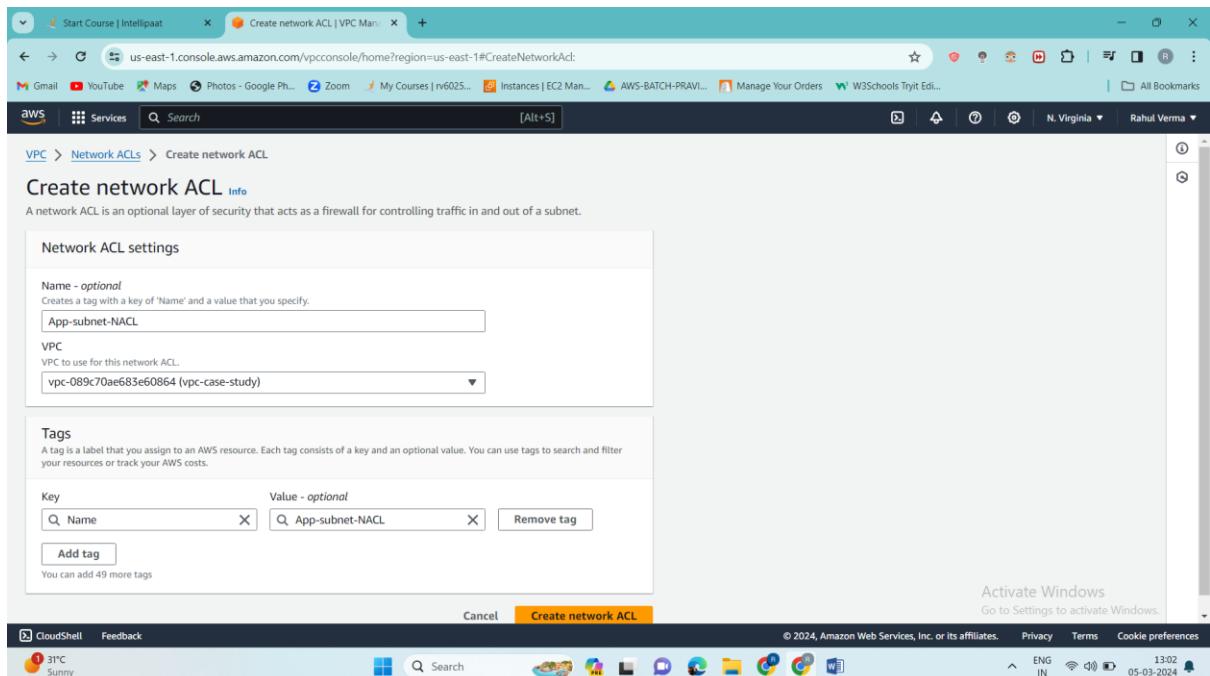
Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
public-subnet-1a	subnet-084bbeac979bfc5b8	acl-0b55932535c781ae2	us-east-1a	10.20.1.0/24	-
<input checked="" type="checkbox"/> private-subnet-1c	subnet-053d707c31f3213e9	acl-0b55932535c781ae2	us-east-1c	10.20.3.0/24	-
<input type="checkbox"/> private-subnet-1b	subnet-0ae3cbb66464a0f3	acl-0b55932535c781ae2	us-east-1b	10.20.2.0/24	-
<input checked="" type="checkbox"/> private-subnet-1d	subnet-0b454face7052712a	acl-0b55932535c781ae2	us-east-1d	10.20.4.0/24	-

Selected subnets

subnet-053d707c31f3213e9 / private-subnet-1c subnet-0b454face7052712a / private-subnet-1d

Cancel

And now will create one NACL for our App subnet



Network ACL settings

Name - optional
Creates a tag with a key of 'Name' and a value that you specify.

VPC
VPC to use for this network ACL.

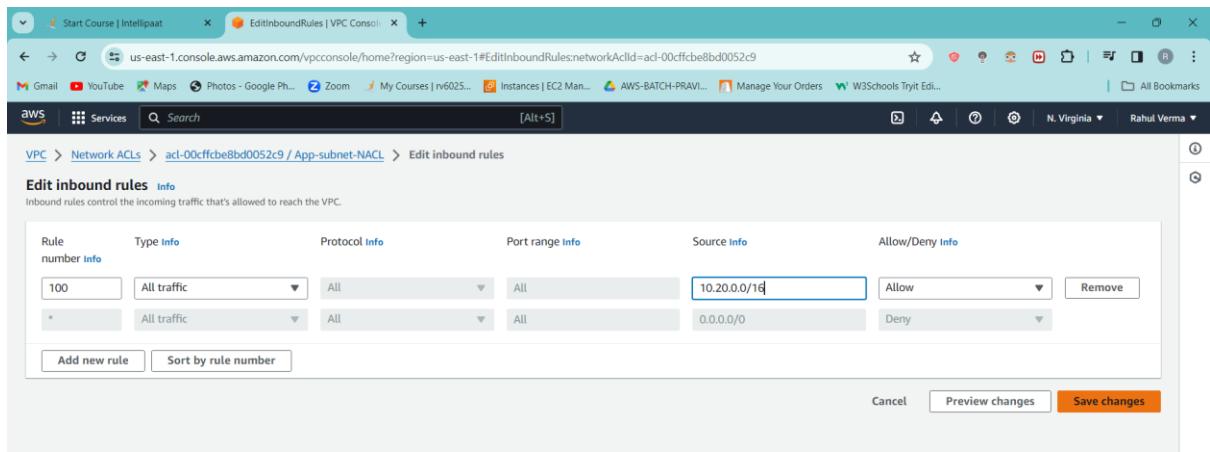
Tags
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="App-subnet-NAACL"/>

You can add 49 more tags

Create network ACL

Now will edit it's inbound rule will allow all traffic to our VPC

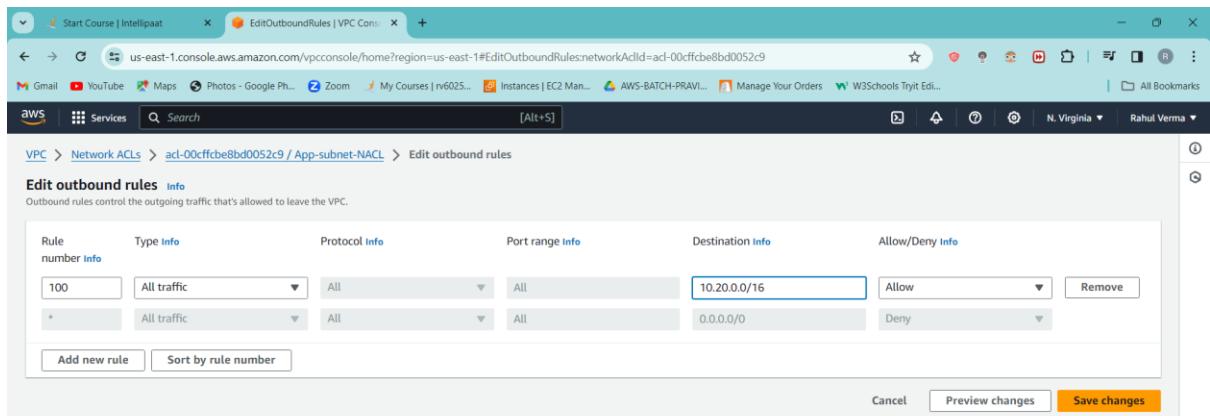


Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Similarly for outbound rule also

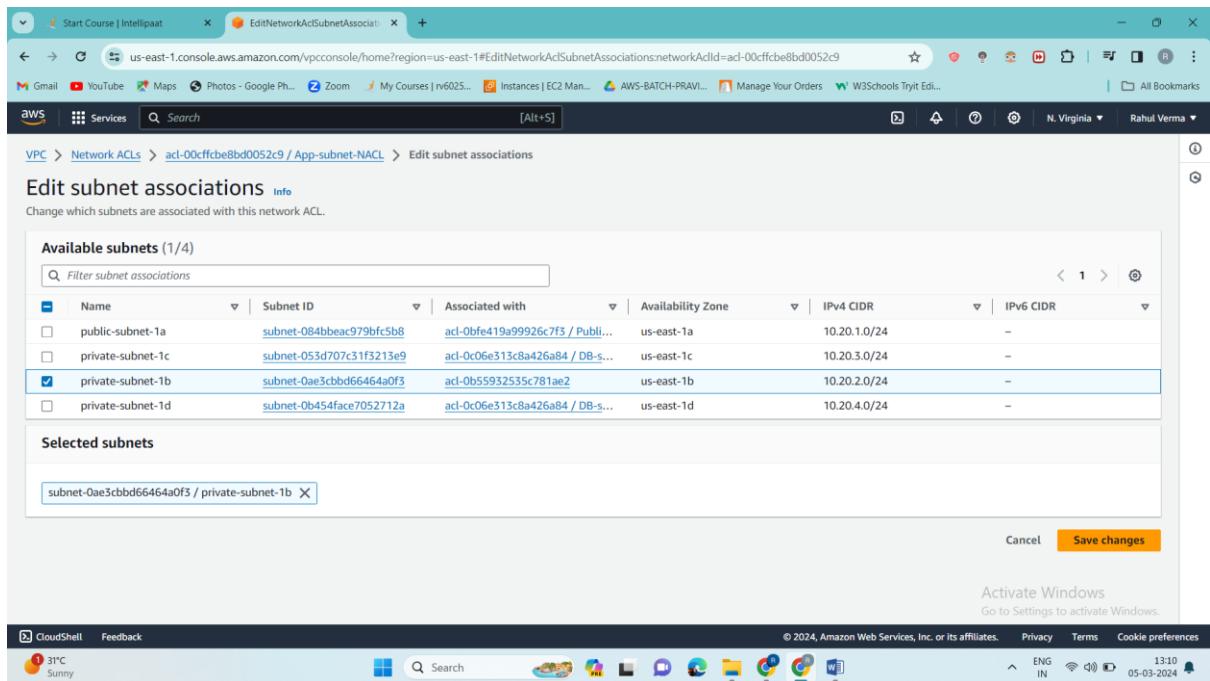


Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Will associate subnet now to our app-subnet-nacl



Available subnets (1/4)

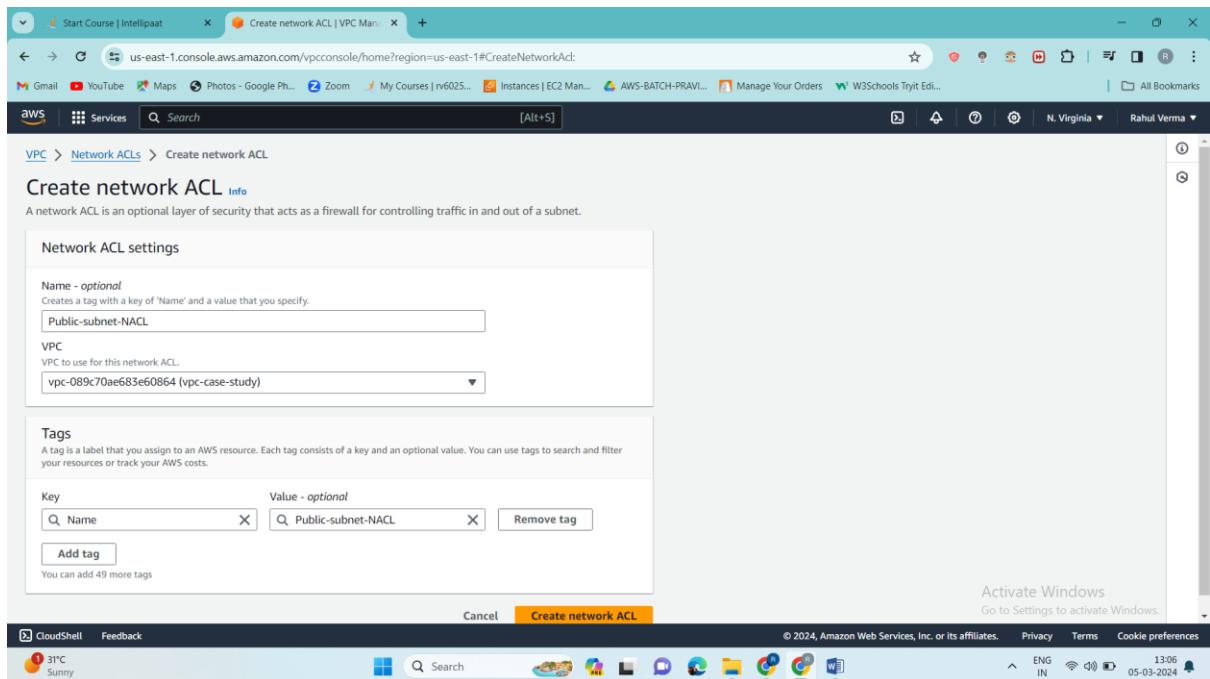
Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
public-subnet-1a	subnet-084bbaec979bfc5b8	acl-0bfe419a99926c7f3 / Public...	us-east-1a	10.20.1.0/24	-
private-subnet-1c	subnet-053d707c31f5213e9	acl-0c06e313c8a426a84 / DB-S...	us-east-1c	10.20.3.0/24	-
<input checked="" type="checkbox"/> private-subnet-1b	<input checked="" type="checkbox"/> subnet-0ae3cbb66464a0f3	<input checked="" type="checkbox"/> acl-0b5932535c781ae2	<input checked="" type="checkbox"/> us-east-1b	<input checked="" type="checkbox"/> 10.20.2.0/24	<input checked="" type="checkbox"/> -
private-subnet-1d	subnet-0b454face7052712a	acl-0c06e313c8a426a84 / DB-S...	us-east-1d	10.20.4.0/24	-

Selected subnets

subnet-0ae3cbb66464a0f3 / private-subnet-1b

Cancel **Save changes**

Now we are going to create one Public subnet NACL



Network ACL settings

Name - optional
Creates a tag with a key of 'Name' and a value that you specify.

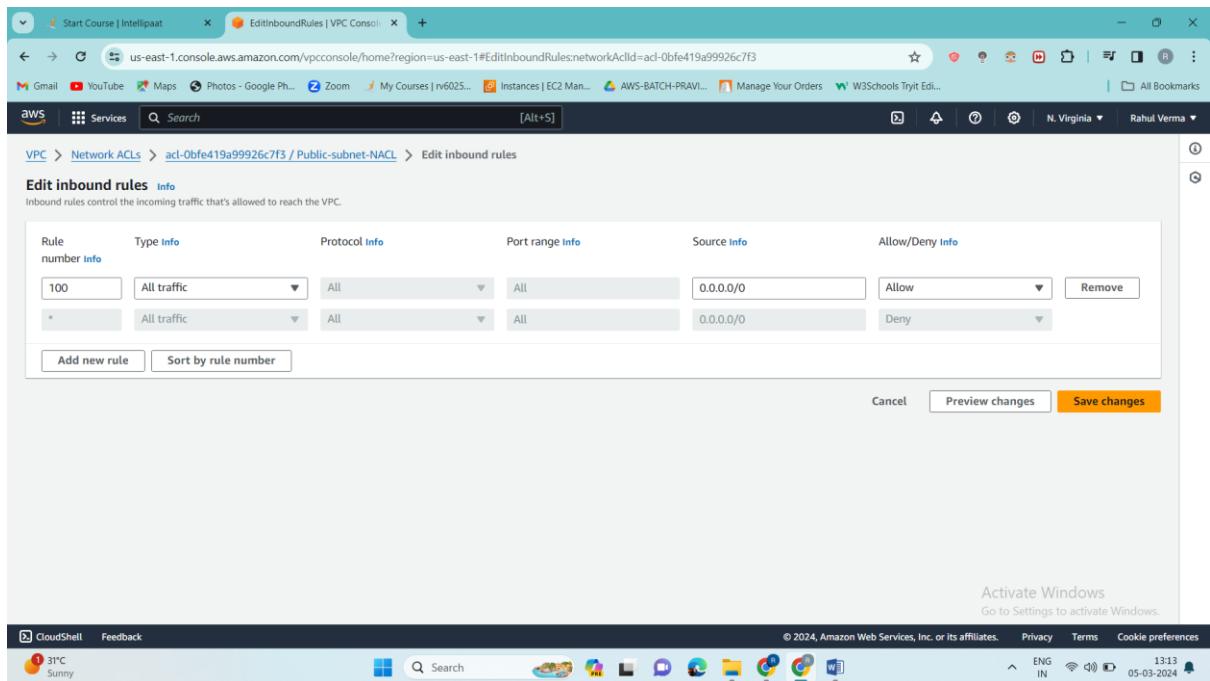
VPC
VPC to use for this network ACL.

Tags
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 or track your AWS costs.

Key Value - optional Remove tag
Add tag
You can add 49 more tags

Cancel **Create network ACL**

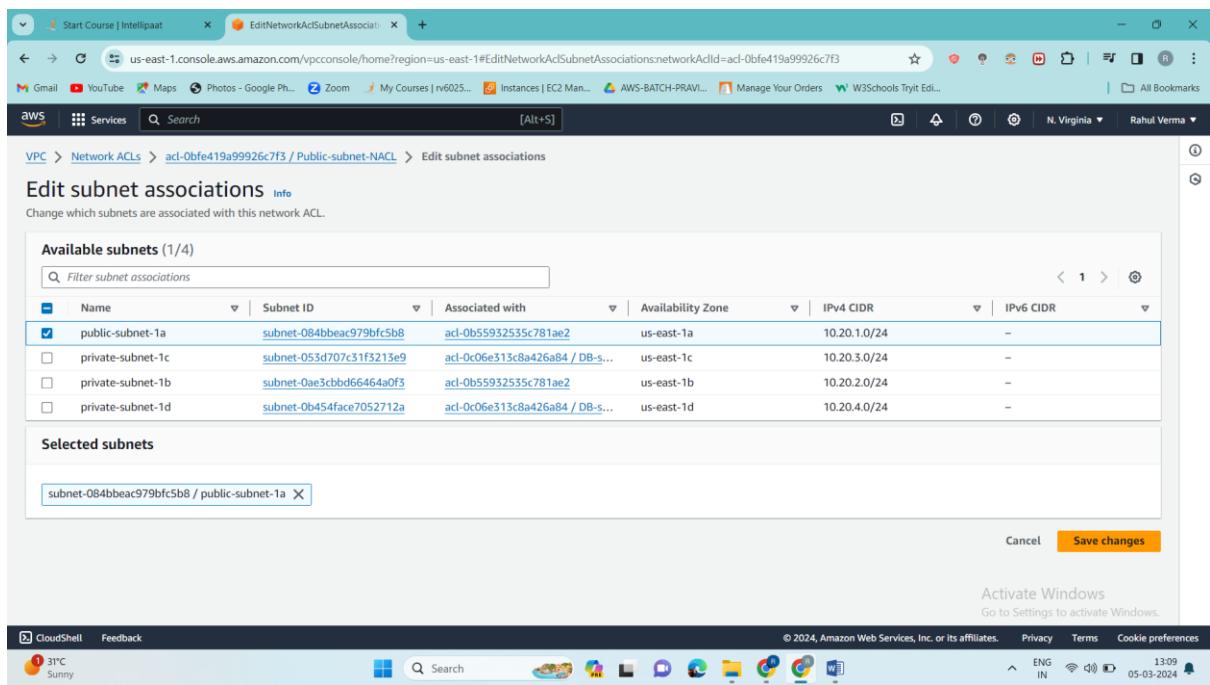
And inbound and outbound is same All traffic anywhere



Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback 31°C Sunny Search ENG IN 13:13 05-03-2024

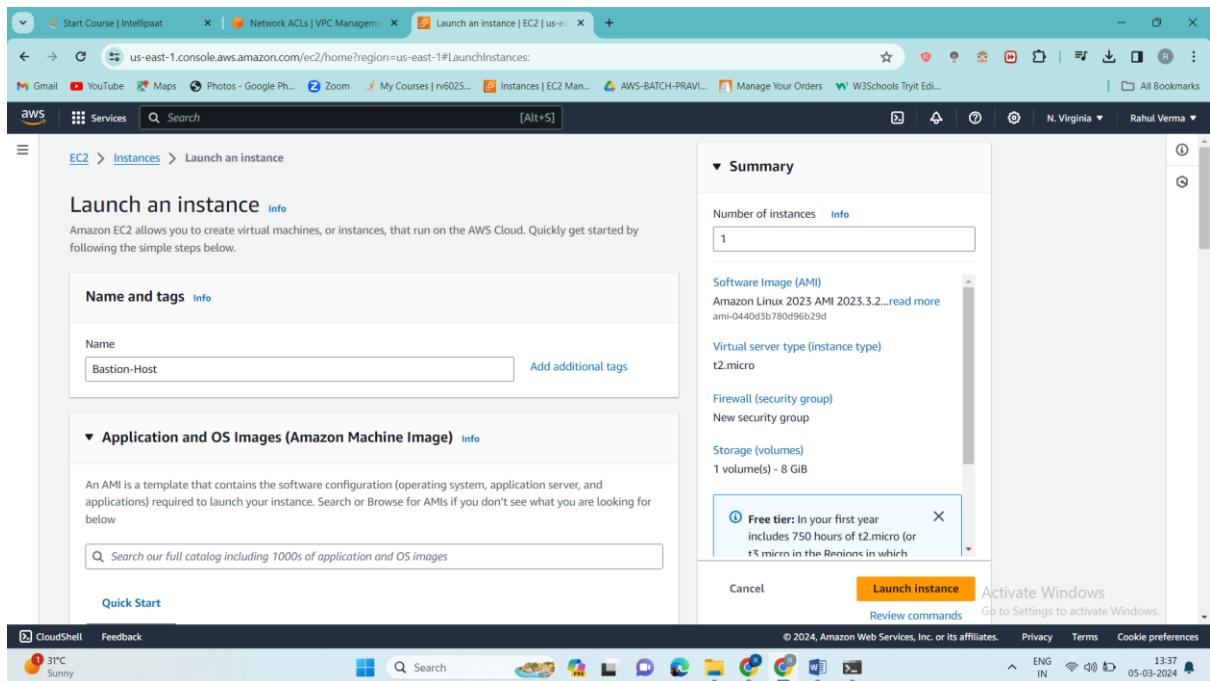
We just need to associate it to public-subnet-nacl



Activate Windows
Go to Settings to activate Windows.

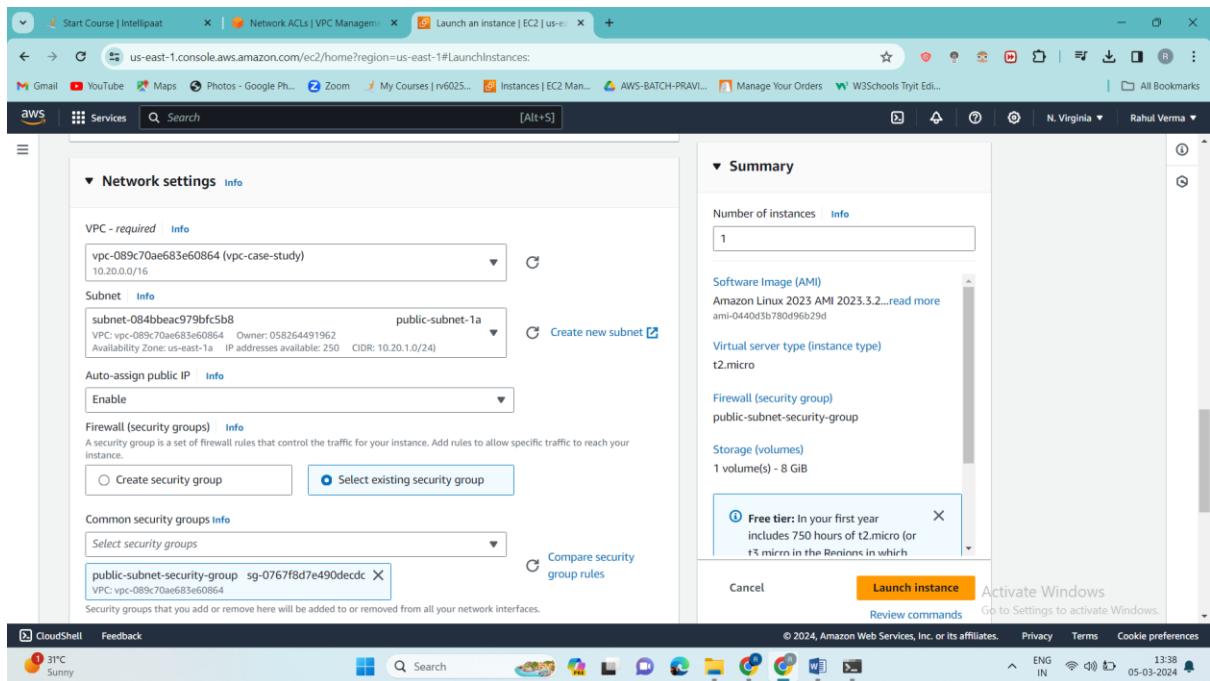
CloudShell Feedback 31°C Sunny Search ENG IN 13:09 05-03-2024

Now will create ec2 instance for “Bastion host step”



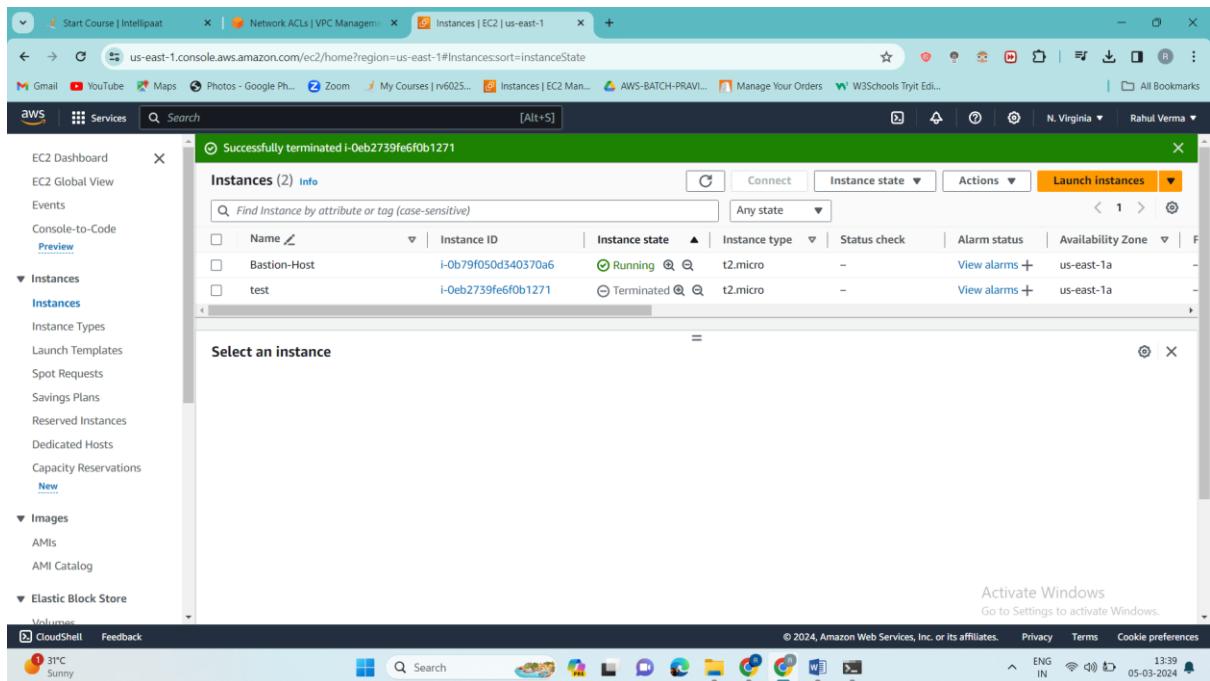
The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The 'Name and tags' section has 'Bastion-Host' entered in the 'Name' field. The 'Software Image (AMI)' is set to 'Amazon Linux 2023 AMI 2023.3.2...'. The 'Virtual server type (instance type)' is 't2.micro'. The 'Firewall (security group)' is 'New security group'. The 'Storage (volumes)' is '1 volume(s) - 8 GiB'. A 'Free tier' information box is visible. The 'Launch instance' button is highlighted in orange.

Will select our VPC and public subnet and public subnet security group



The screenshot shows the 'Network settings' section of the 'Launch an instance' wizard. The 'VPC - required' dropdown is set to 'vpc-089c70ae683e60864 (vpc-case-study)'. The 'Subnet' dropdown is set to 'subnet-084bbeac979bfc5b8 public-subnet-1a'. The 'Auto-assign public IP' dropdown is set to 'Enable'. The 'Firewall (security groups)' dropdown has 'Select existing security group' selected. The 'Common security groups' dropdown shows 'public-subnet-security-group sg-0767f8d7e490dec'. A 'Free tier' information box is visible. The 'Launch instance' button is highlighted in orange.

Our instance is created

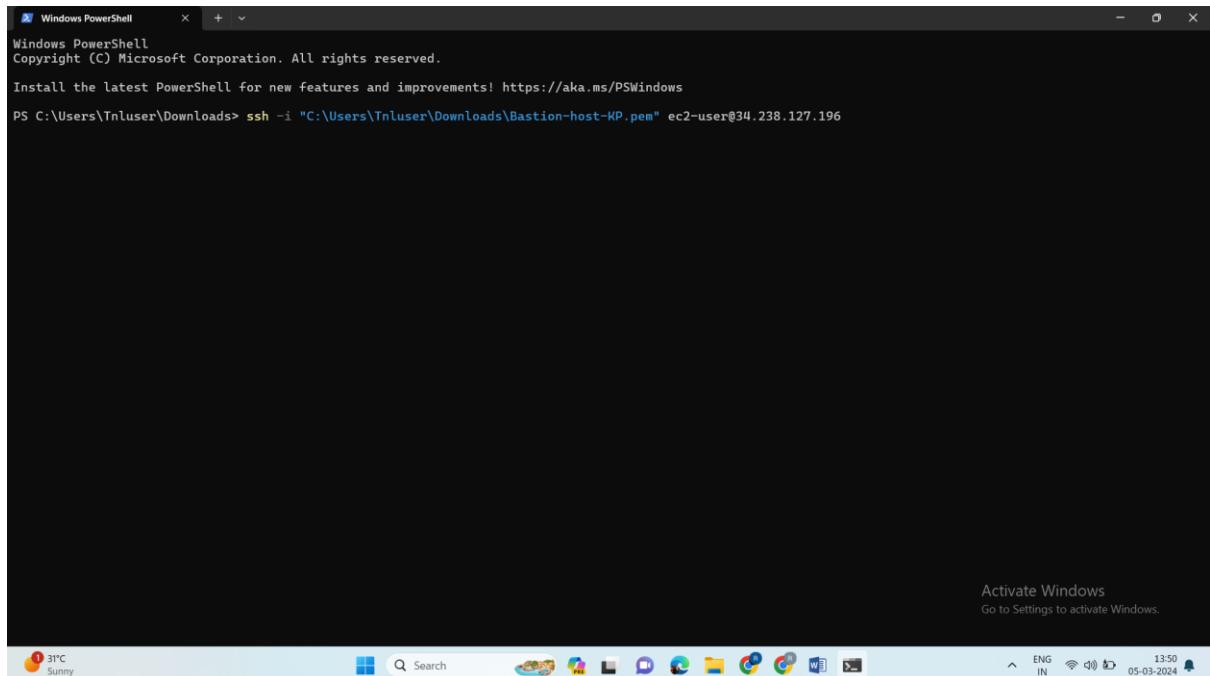


The screenshot shows the AWS EC2 Instances page. The left sidebar is collapsed. The main content area displays a table of instances. The table has columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. There are two instances listed:

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Bastion-Host	i-0b79f050d340370a6	Running	t2.micro	-	View alarms	us-east-1a
test	i-0eb2739fe6f0b1271	Terminated	t2.micro	-	View alarms	us-east-1a

Below the table, a modal window titled "Select an instance" is open, showing the same two instances. The "test" instance is selected. The bottom right of the modal shows "Activate Windows" and "Go to Settings to activate Windows". The bottom of the screen shows a Windows taskbar with various icons and system status.

Now let's connect to our instance, we should be able to do it as it is public



The screenshot shows a Windows PowerShell window. The command entered is:

```
PS C:\Users\Tnluser\Downloads> ssh -i "C:\Users\Tnluser\Downloads\Bastion-host-KP.pem" ec2-user@34.238.127.196
```

The bottom right of the screen shows a Windows taskbar with various icons and system status.

And we are connected successfully

Now we have to create two S3 bucket as per requirements. One is to store images for application and second for hosting website.

So the below bucket will be used to upload images and it has no public access

Start Course | Intellipaat x

Create S3 bucket | S3 | Global x

Gmail YouTube Maps Photos - Google Ph... Zoom My Courses | rv6025... Instances | EC2 Man... AWS-BATCH-PRAVI... Manage Your Orders W3Schools Tryit Ed... All Bookmarks

AWS Services Search [Alt+5] Global Rahul Verma

Amazon S3 > Buckets > Create bucket

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region US East (N. Virginia) us-east-1

Bucket type Info

General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

Directory - New
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name Info aws-project-bucket

Bucket name must be unique within the global namespace and follow the bucket naming rules. [See rules for bucket naming](#)

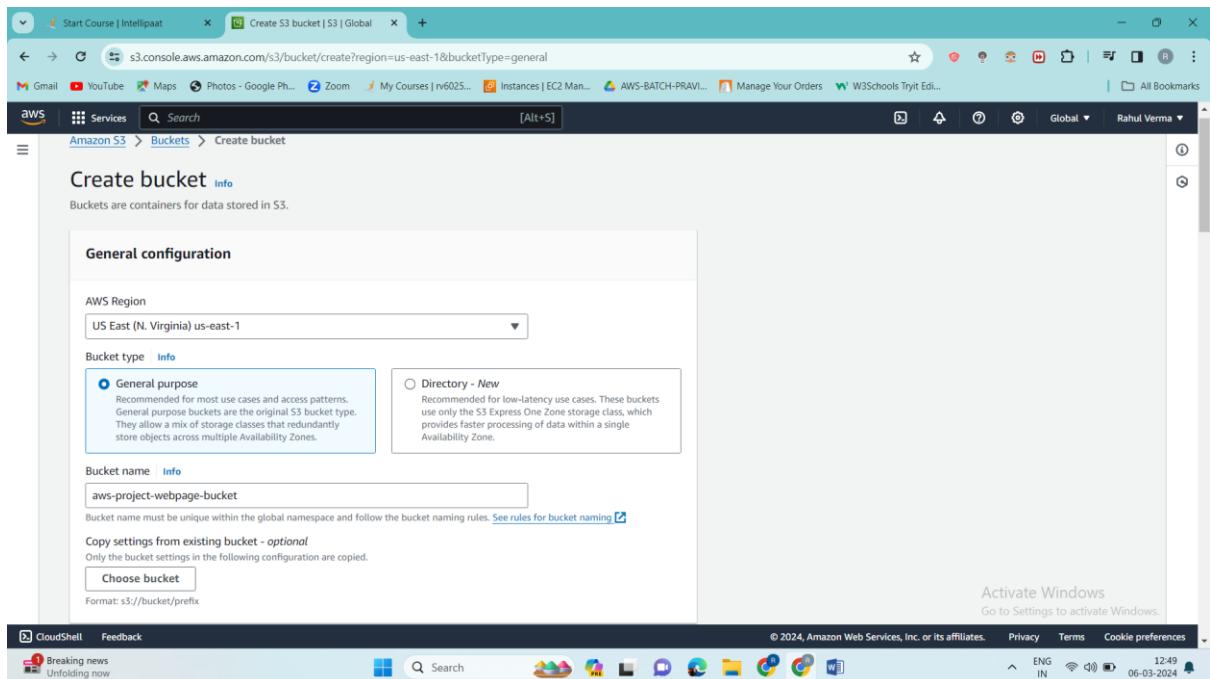
Copy settings from existing bucket - optional
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Format: `s3://bucket/prefix`

Activate Windows
Go to Settings to activate Windows.

Now will create bucket for our static website and it has public access



General configuration

AWS Region: US East (N. Virginia) us-east-1

Bucket type: General purpose Directory - New

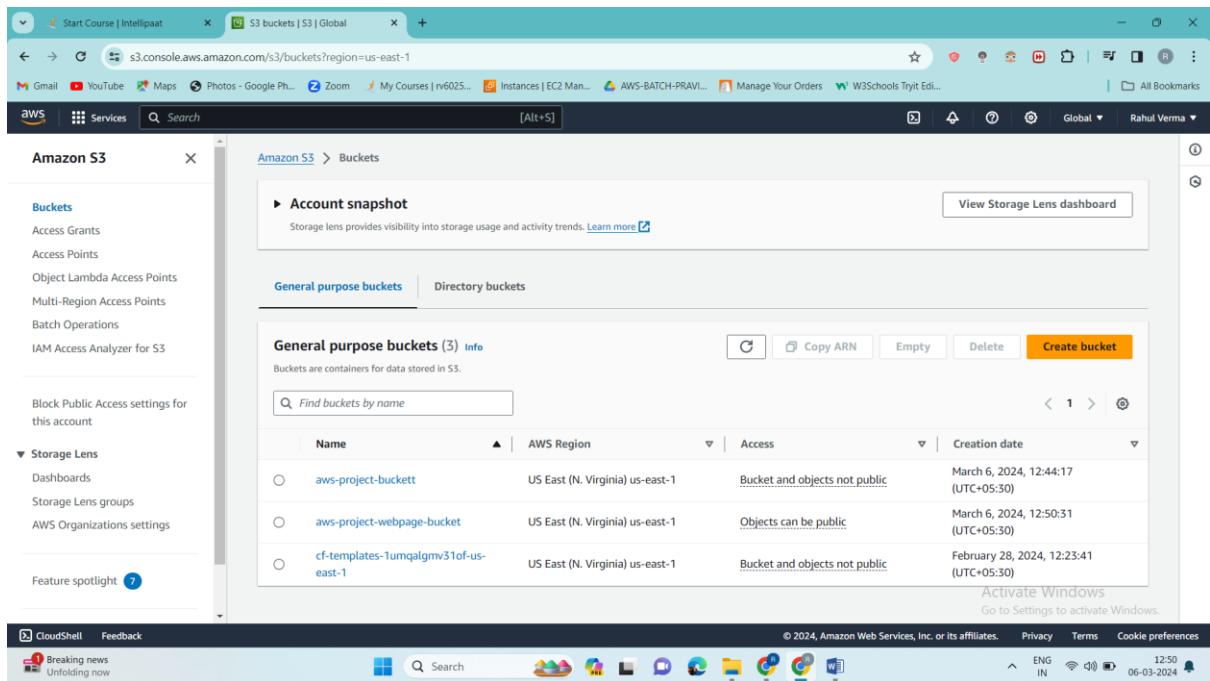
Bucket name: aws-project-webpage-bucket

Copy settings from existing bucket - optional

Choose bucket

Format: s3://bucket/prefix

Our buckets are created now



Amazon S3

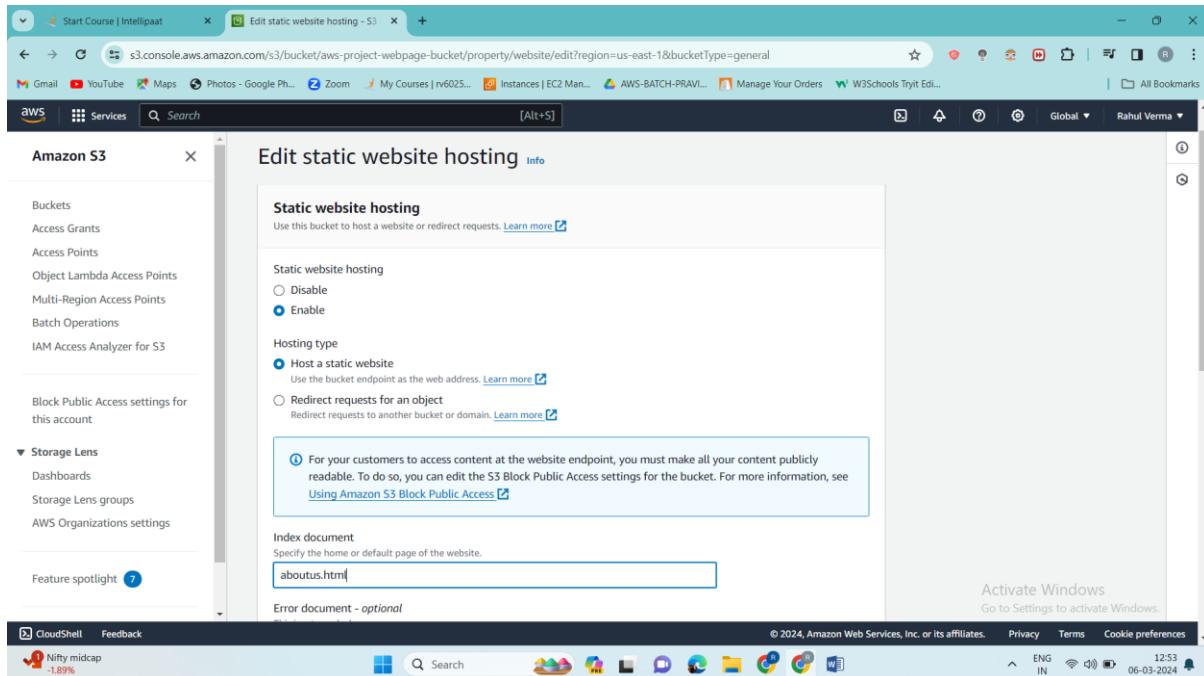
Buckets

General purpose buckets (3) Directory buckets

Name	AWS Region	Access	Creation date
aws-project-bucket	US East (N. Virginia) us-east-1	Bucket and objects not public	March 6, 2024, 12:44:17 (UTC+05:30)
aws-project-webpage-bucket	US East (N. Virginia) us-east-1	Objects can be public	March 6, 2024, 12:50:31 (UTC+05:30)
cf-templates-1umqalmv31of-us-east-1	US East (N. Virginia) us-east-1	Bucket and objects not public	February 28, 2024, 12:23:41 (UTC+05:30)

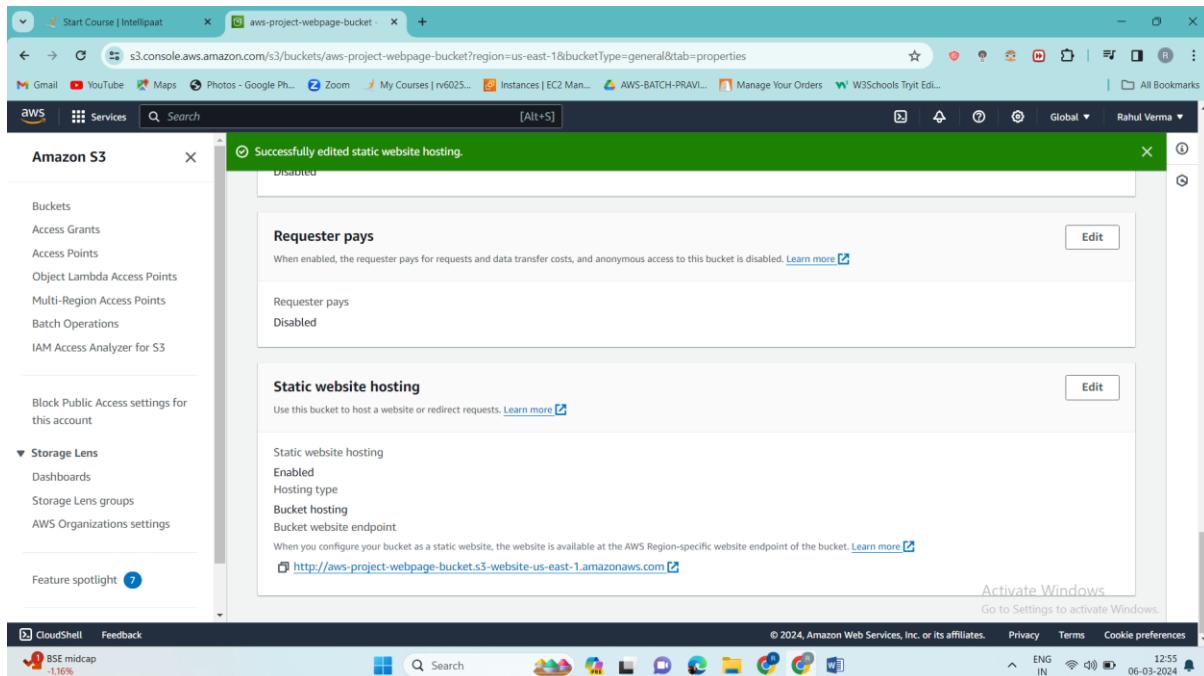
Now we will enable static website host in our bucket which is aws-project-webpage-bucket

Go to properties → edit static website hosting and enable it. Index doc- aboutus.html



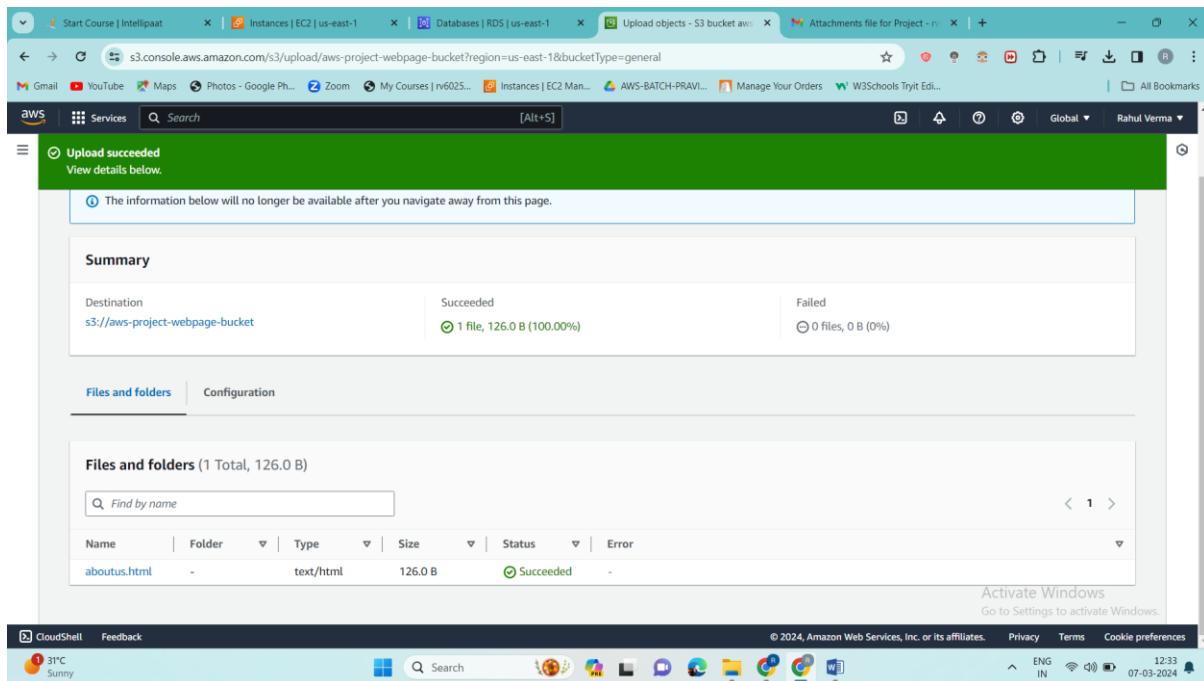
The screenshot shows the AWS S3 console with the 'Edit static website hosting' page for the 'aws-project-webpage-bucket'. The 'Static website hosting' section is enabled, and the 'Index document' is set to 'aboutus.html'. A note in the 'Index document' field states: 'For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see Using Amazon S3 Block Public Access'.

And this is the endpoint



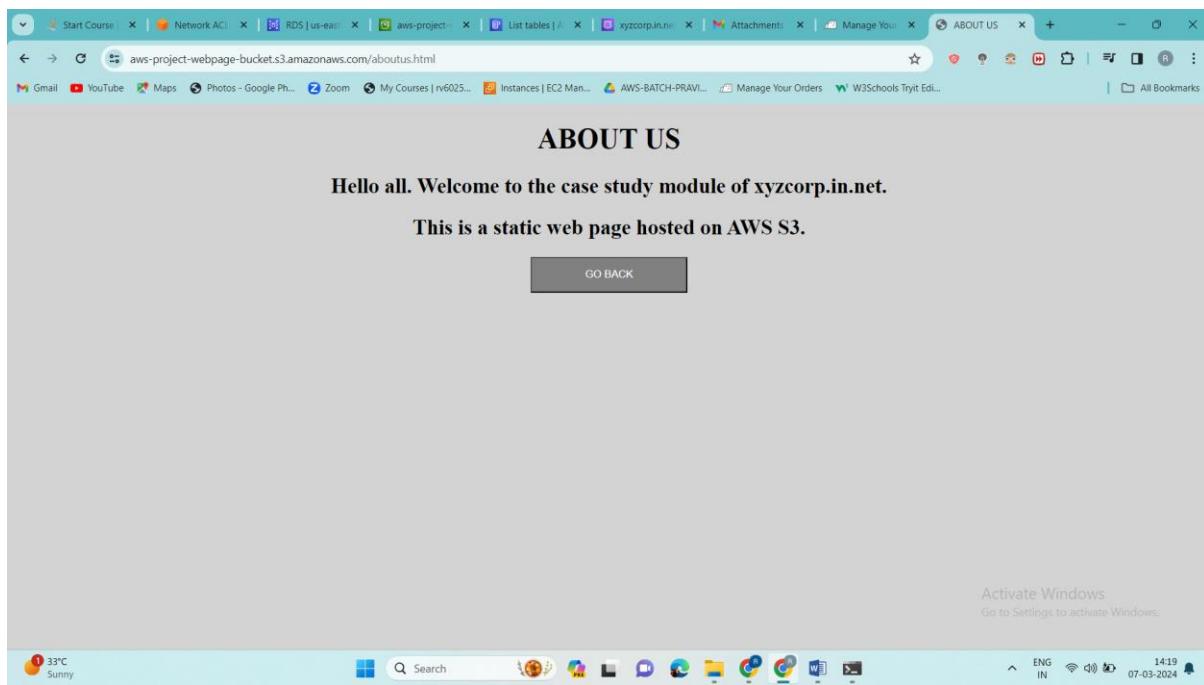
The screenshot shows the AWS S3 console with a green success message: 'Successfully edited static website hosting.' The 'Requester pays' section is disabled. The 'Static website hosting' section is enabled, and the 'Bucket website endpoint' is listed as <http://aws-project-webpage-bucket.s3-website-us-east-1.amazonaws.com>.

Now let's upload a file in our bucket `aboutus.html`



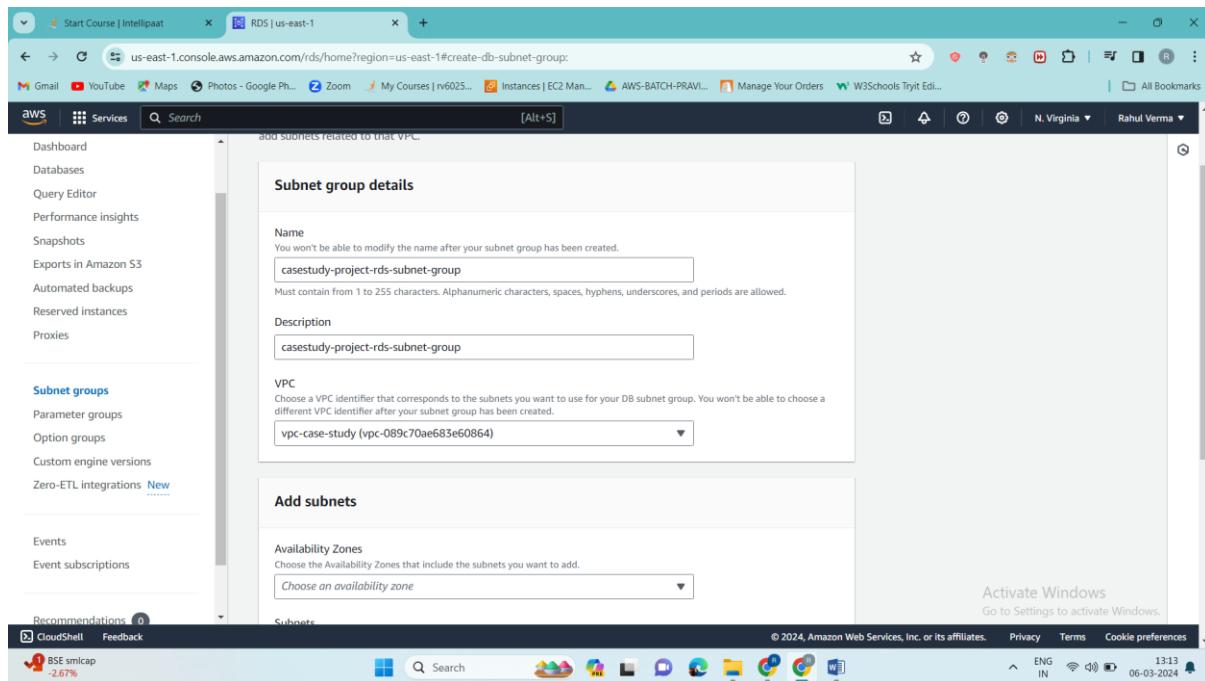
The screenshot shows the AWS S3 console with a green success message: "Upload succeeded". The summary table shows 1 file uploaded successfully (126.0 B) and 0 files failed. The "Files and folders" tab is selected, showing a table with one item: "aboutus.html" (text/html, 126.0 B, Succeeded). The status bar at the bottom indicates "CloudShell Feedback" and the date "07-03-2024".

And let's check if it is working properly or not



The screenshot shows a web browser displaying the static website "aboutus.html" from the "aws-project-webpage-bucket". The page content includes "ABOUT US", "Hello all. Welcome to the case study module of xyzcorp.in.net.", and "This is a static web page hosted on AWS S3.". A "GO BACK" button is visible. The status bar at the bottom indicates "CloudShell Feedback" and the date "07-03-2024".

Now will go to next step and we have to create RDS so before creating RDS will create Subnet group



Subnet group details

Name: casestudy-project-rds-subnet-group

Description: casestudy-project-rds-subnet-group

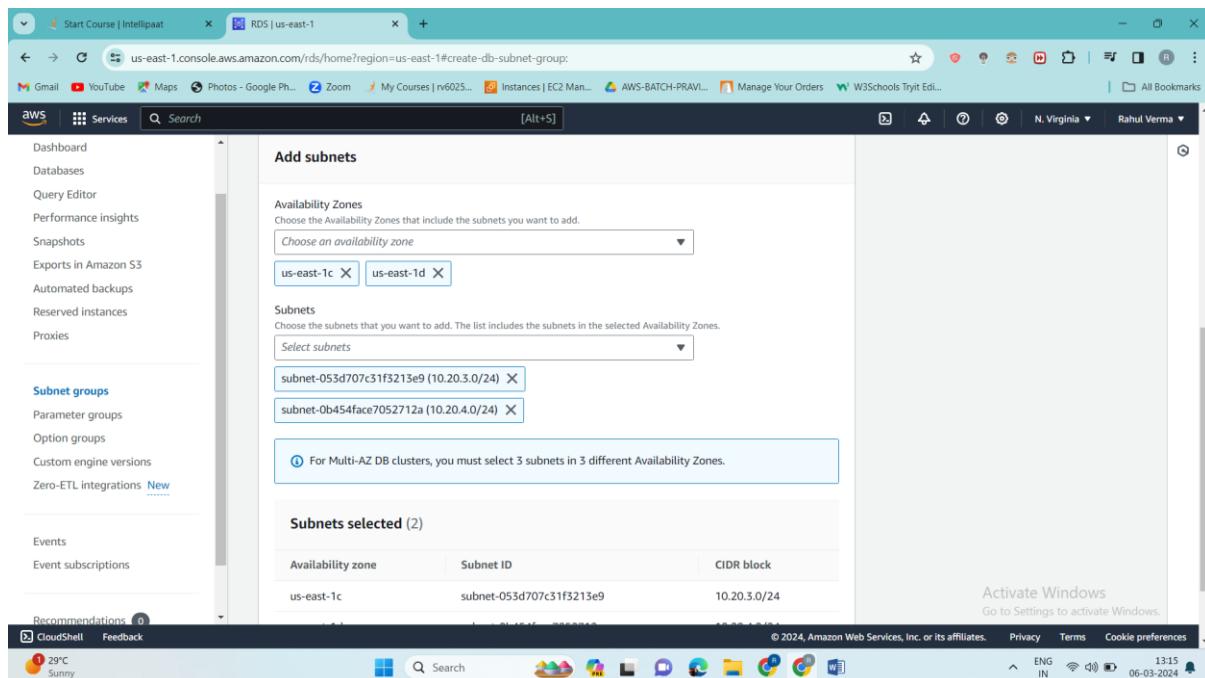
VPC: vpc-case-study (vpc-089c70ae683e60864)

Add subnets

Availability Zones: Choose an availability zone

Subnets: Choose subnets

Now will add subnets which us-east1c and us-east1d



Add subnets

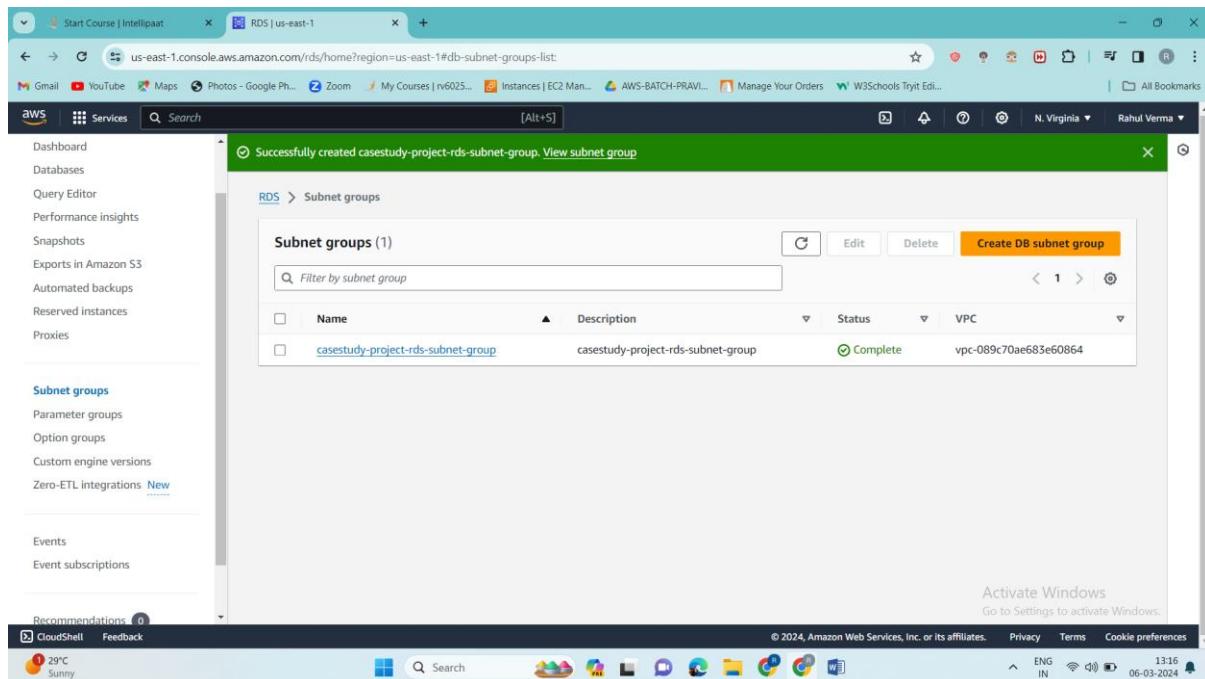
Availability Zones: Choose an availability zone

Subnets: Select subnets

For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones.

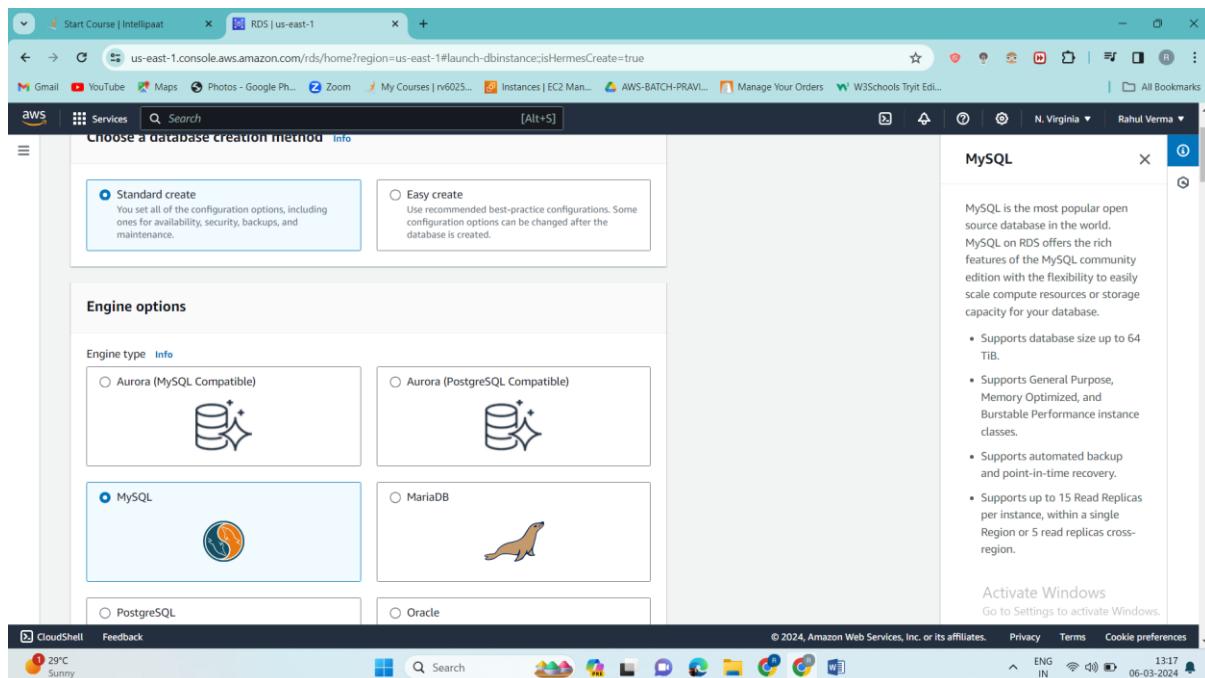
Availability zone	Subnet ID	CIDR block
us-east-1c	subnet-053d707c31f3213e9	10.20.3.0/24

It is created



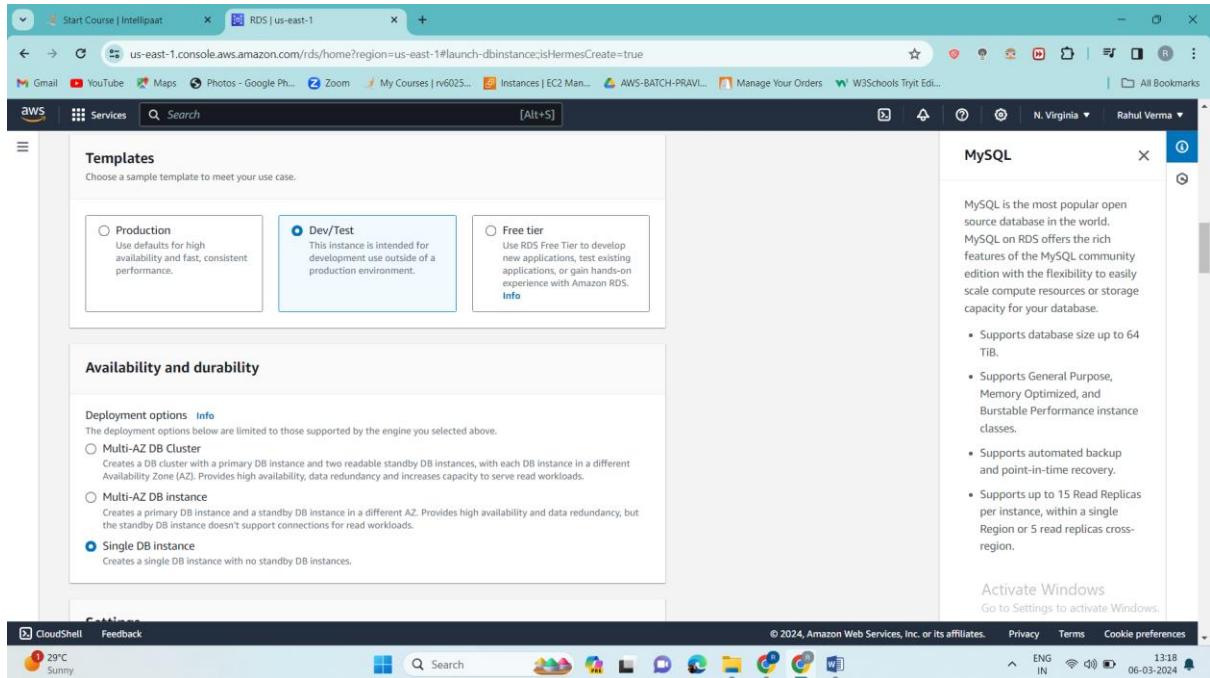
The screenshot shows the AWS RDS Subnet Groups page. A green success message at the top says "Successfully created casestudy-project-rds-subnet-group. View subnet group". The main table shows one subnet group named "casestudy-project-rds-subnet-group" with a status of "Complete" and a VPC of "vpc-089c70ae683e60864". The left sidebar includes links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, and Event subscriptions. The bottom right corner shows "Activate Windows" and the date "06-03-2024".

Now will have to create RDS according to steps using Mysql database



The screenshot shows the "Choose a database creation method" page. It offers two options: "Standard create" (selected) and "Easy create". The "Standard create" section includes a note: "You set all of the configuration options, including ones for availability, security, backups, and maintenance." The "MySQL" engine type is selected under "Engine options". The "MySQL" section on the right provides a brief description of MySQL and a bulleted list of features: "Supports database size up to 64 TiB.", "Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.", "Supports automated backup and point-in-time recovery.", and "Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region." The bottom right corner shows "Activate Windows" and the date "06-03-2024".

Templates Dev/test



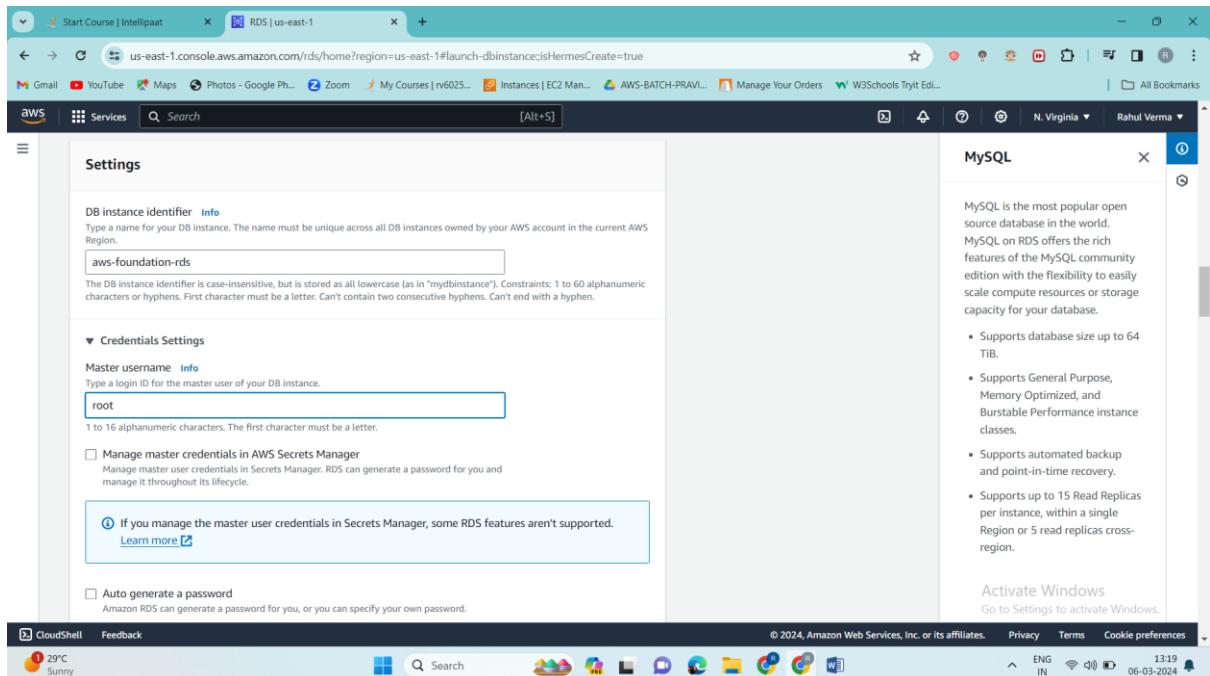
MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

Activate Windows
Go to Settings to activate Windows.

Db instance name- aws-foundation-rds and username- root password is- intel123



MySQL

MySQL is the most popular open source database in the world. MySQL on RDS offers the rich features of the MySQL community edition with the flexibility to easily scale compute resources or storage capacity for your database.

- Supports database size up to 64 TiB.
- Supports General Purpose, Memory Optimized, and Burstable Performance instance classes.
- Supports automated backup and point-in-time recovery.
- Supports up to 15 Read Replicas per instance, within a single Region or 5 read replicas cross-region.

Activate Windows
Go to Settings to activate Windows.

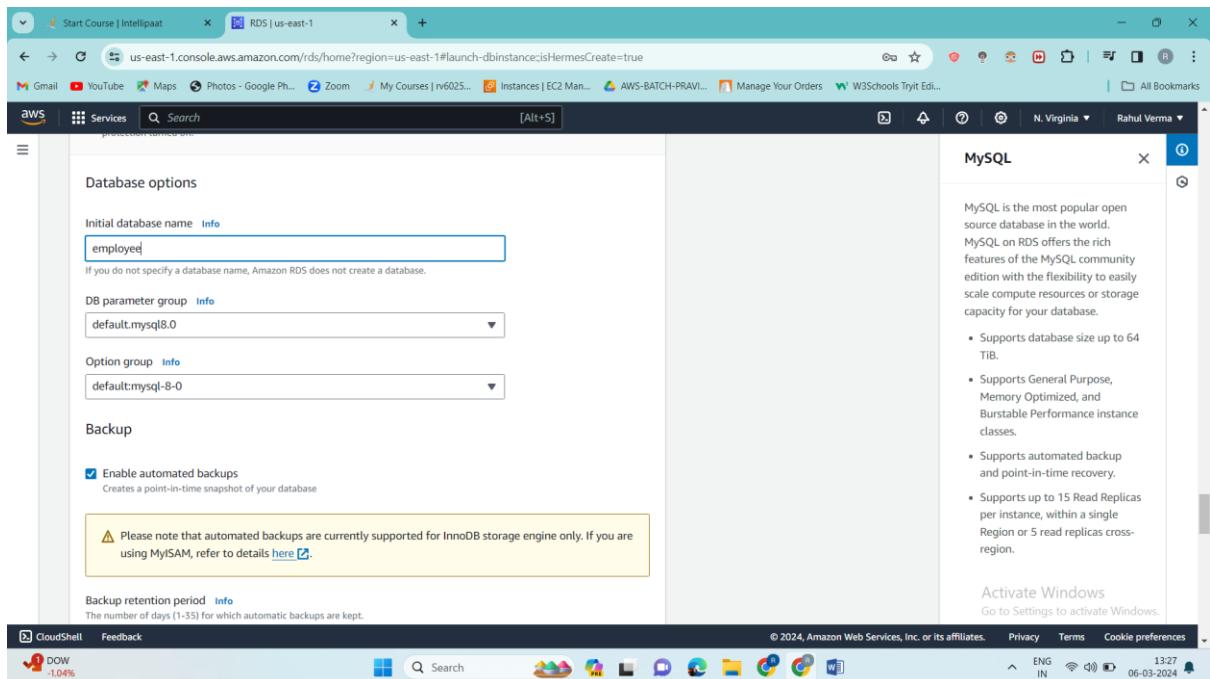
Now select our VPC and subnet group

The screenshot shows the AWS RDS console for creating a new database. The left panel is titled 'Connectivity' and includes sections for 'Compute resource', 'Network type', and 'Virtual private cloud (VPC)'. The 'Compute resource' section has two options: 'Don't connect to an EC2 compute resource' (selected) and 'Connect to an EC2 compute resource'. The 'Network type' section has two options: 'IPv4' (selected) and 'Dual-stack mode'. The 'Virtual private cloud (VPC)' section shows 'vpc-case-study (vpc-089c70ae683e60864)' selected, with a note that 4 Subnets and 4 Availability Zones are available. A note at the bottom states 'After a database is created, you can't change its VPC.' The right panel is titled 'MySQL' and provides a brief overview of the service, including its popularity and supported features like automated backups and up to 15 read replicas. The bottom of the screen shows the AWS navigation bar and a status bar indicating the user is in N. Virginia and the date is 06-03-2024.

Will select our security group

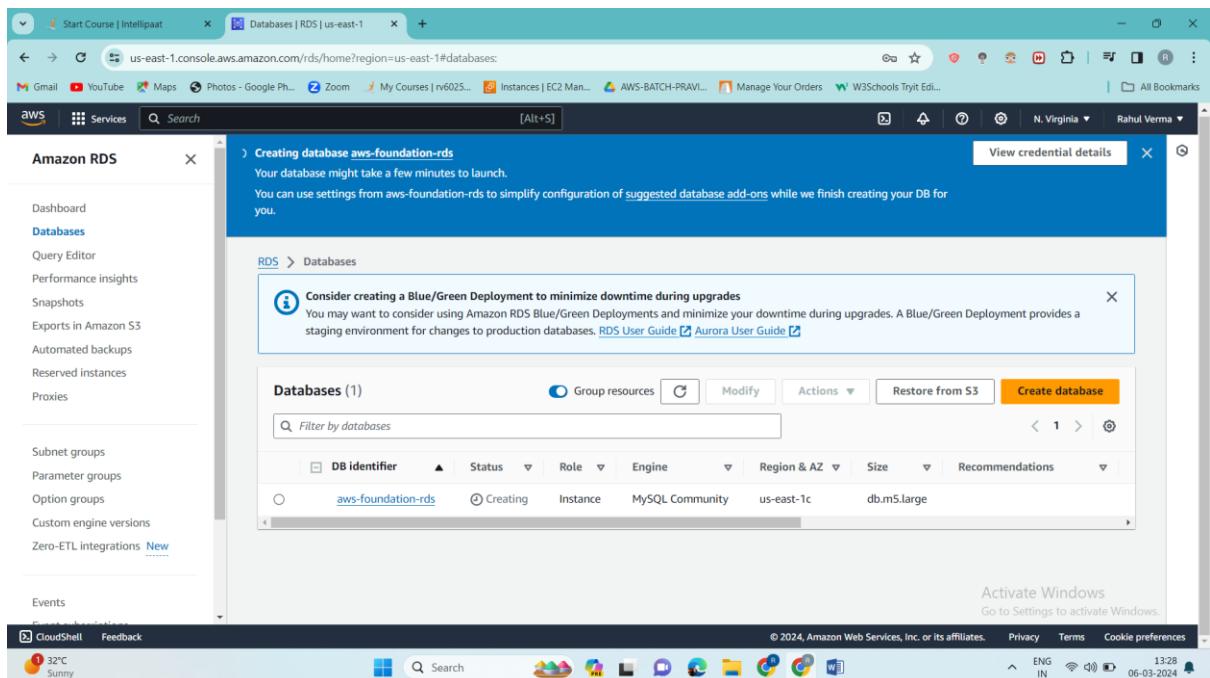
The screenshot shows the AWS RDS console for creating a new database. The left panel is titled 'VPC security group (firewall)' and includes sections for 'Existing VPC security groups', 'Availability Zone', 'RDS Proxy', and 'Certificate authority - optional'. The 'Existing VPC security groups' section shows 'RDS-DB-security-group' selected. The right panel is titled 'MySQL' and provides a brief overview of the service, including its popularity and supported features like automated backups and up to 15 read replicas. The bottom of the screen shows the AWS navigation bar and a status bar indicating the user is in N. Virginia and the date is 06-03-2024.

Database name as employee



The screenshot shows the AWS RDS console for the 'us-east-1' region. A new database named 'employee' is being created. The 'MySQL' section on the right provides a brief overview of the database engine, mentioning its popularity and various features. The 'Database options' section includes fields for the database name (set to 'employee'), DB parameter group (set to 'default.mysql8.0'), and Option group (set to 'default.mysql-8-0'). The 'Backup' section shows that automated backups are enabled. A note indicates that automated backups are currently supported for InnoDB storage engine only. The 'Backup retention period' is set to 35 days. The status bar at the bottom shows the date as 06-03-2024 and the time as 13:27.

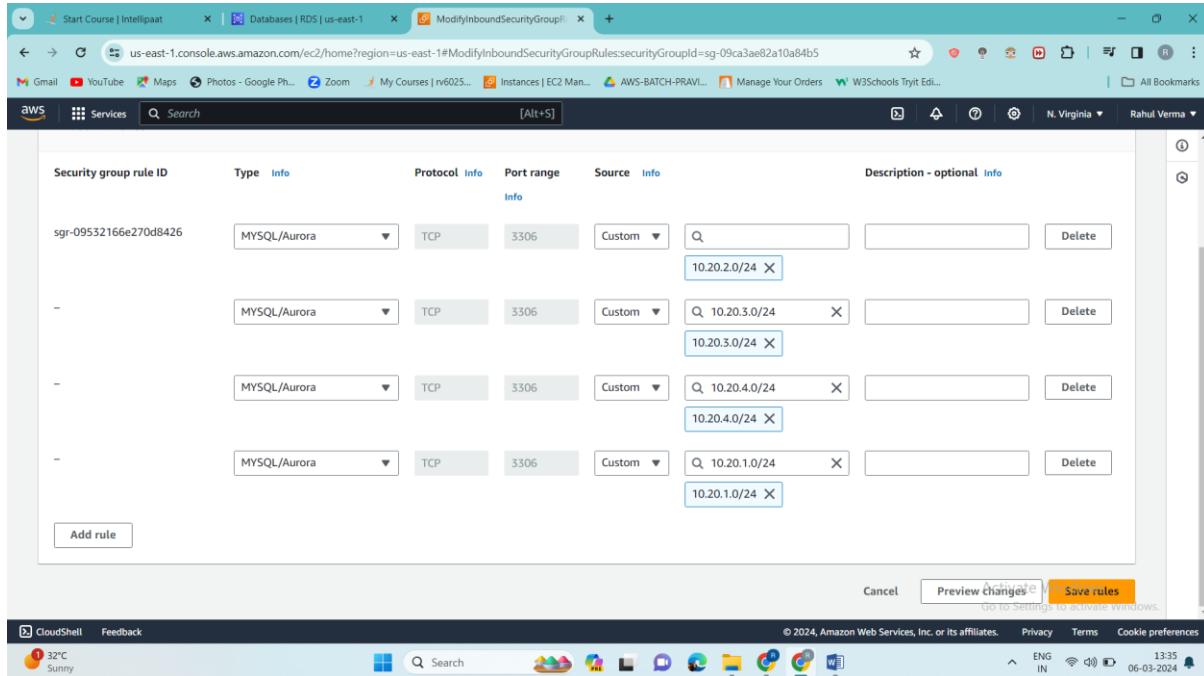
Our RDS is created now



The screenshot shows the AWS RDS console with the 'Databases' section selected. A message indicates that a database named 'aws-foundation-rds' is being created. The 'Databases' table lists one entry: 'aws-foundation-rds' (Status: Creating, Instance: MySQL Community, Engine: MySQL Community, Region & AZ: us-east-1c, Size: db.m5.large). A note about Blue/Green Deployments is visible. The status bar at the bottom shows the date as 06-03-2024 and the time as 13:28.

Now will connect to this RDS via Bastion host. But this RDS is does not have public access

So we have to edit security group of our RDS. Will allow public subnet

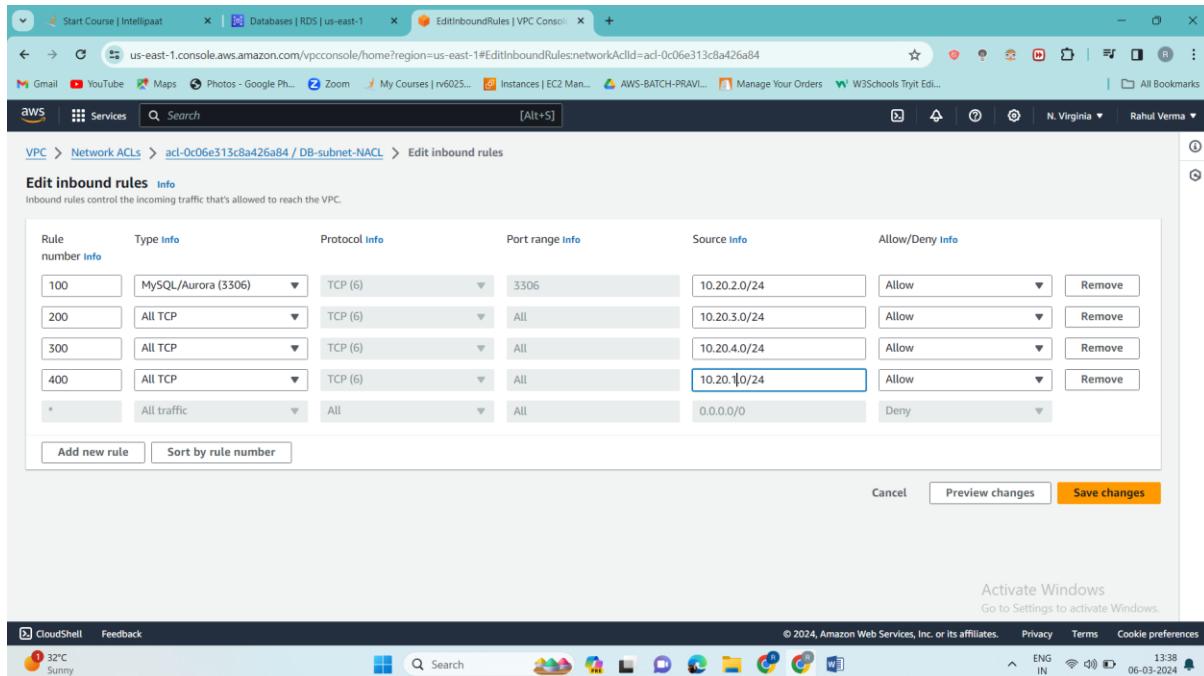


The screenshot shows the 'ModifyInboundSecurityGroupRules' interface for an RDS security group. The table lists five rules, all of which are currently disabled (indicated by a red 'X' in the 'Delete' button). The rules are as follows:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-09532166e270d8426	MySQL/Aurora	TCP	3306	Custom	10.20.2.0/24 X
-	MySQL/Aurora	TCP	3306	Custom	10.20.3.0/24 X
-	MySQL/Aurora	TCP	3306	Custom	10.20.4.0/24 X
-	MySQL/Aurora	TCP	3306	Custom	10.20.1.0/24 X

At the bottom left is an 'Add rule' button. At the bottom right are 'Cancel', 'Preview changes', and a highlighted 'Save rules' button.

And we have to make changes in Network ACL also we just have to allow public subnet also

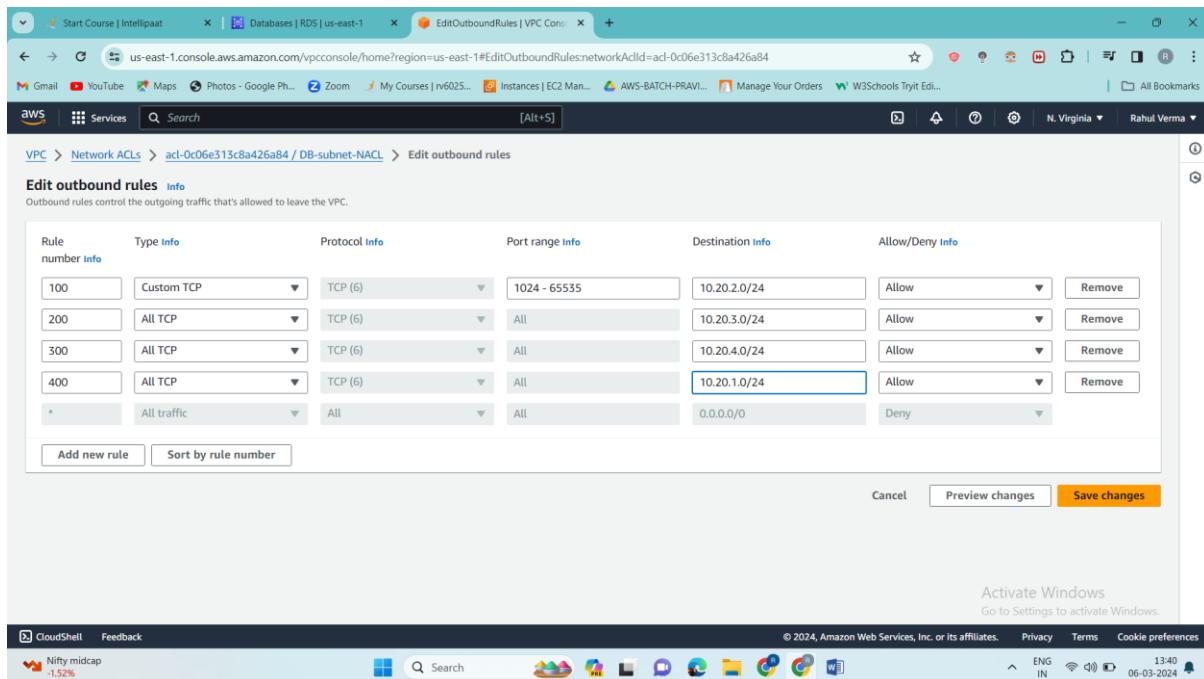


The screenshot shows the 'EditInboundRules' interface for a Network ACL. The table lists five inbound rules, all of which are currently disabled (indicated by a red 'X' in the 'Remove' button). The rules are as follows:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	MySQL/Aurora (3306)	TCP (6)	3306	10.20.2.0/24	Allow X
200	All TCP	TCP (6)	All	10.20.3.0/24	Allow X
300	All TCP	TCP (6)	All	10.20.4.0/24	Allow X
400	All TCP	TCP (6)	All	10.20.1.0/24	Allow X
*	All traffic	All	All	0.0.0.0/0	Deny X

At the bottom left is an 'Add new rule' button. At the bottom right are 'Cancel', 'Preview changes', and a highlighted 'Save changes' button.

And same in the outbound rule also



The screenshot shows the AWS VPC console with the URL us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#EditOutboundRules:networkAclId=acl-0c06e313c8a426a84. The page is titled "Edit outbound rules" under "Network ACLs". It displays a table of outbound rules with the following data:

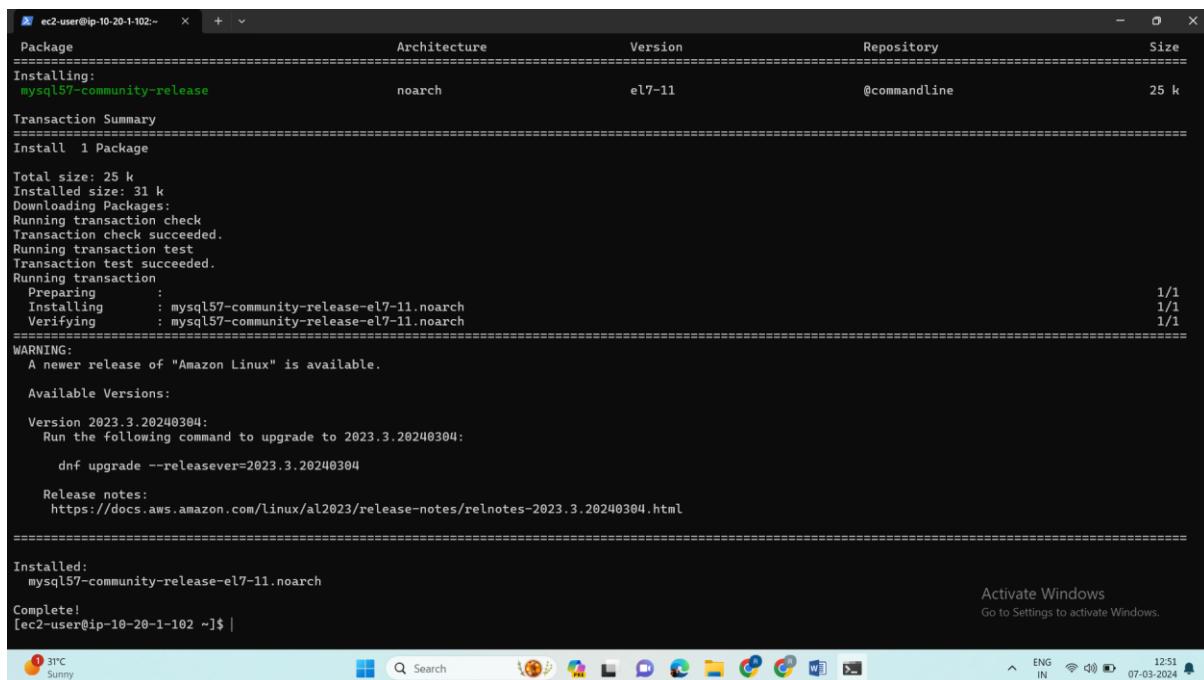
Rule number	Type	Protocol	Port range	Destination	Action
100	Custom TCP	TCP (6)	1024 - 65535	10.20.2.0/24	Allow
200	All TCP	TCP (6)	All	10.20.3.0/24	Allow
300	All TCP	TCP (6)	All	10.20.4.0/24	Allow
400	All TCP	TCP (6)	All	10.20.1.0/24	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Buttons at the bottom include "Add new rule", "Sort by rule number", "Cancel", "Preview changes", and "Save changes". The status bar at the bottom right shows "Activate Windows Go to Settings to activate Windows" and the date "06-03-2024".

Now let's connect to our instance and from this let's try to connect our RDS

Connected to our instance and let's install mysql in our ec2 instance

Command-- `sudo yum install -y https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm`



The screenshot shows an AWS CloudShell terminal window with the command history:

```
ec2-user@ip-10-20-1-102:~ % sudo yum install -y https://dev.mysql.com/get/mysql57-community-release-el7-11.noarch.rpm
Package                                         Architecture   Version      Repository  Size
=====
Installing:
  mysql57-community-release                   noarch        el7-11      @commandline 25 k
Transaction Summary
=====
Install 1 Package

Total size: 25 k
Installed size: 31 k
Downloading Packages:
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing                                           1/1
  Installing  mysql57-community-release-el7-11.noarch 1/1
  Verifying   mysql57-community-release-el7-11.noarch 1/1
=====
WARNING:
  A newer release of "Amazon Linux" is available.

Available Versions:
  Version 2023.3.20240304:
    Run the following command to upgrade to 2023.3.20240304:
      dnf upgrade --releasever=2023.3.20240304
    Release notes:
      https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====
Installed:
  mysql57-community-release-el7-11.noarch
Complete!
[ec2-user@ip-10-20-1-102 ~]$ |
```

The status bar at the bottom right shows "Activate Windows Go to Settings to activate Windows" and the date "07-03-2024".

Command--- sudo yum install -y mysql-community-client

```
[ec2-user@ip-10-20-1-102:~] Dependencies resolved.
=====
Package          Architecture Version      Repository  Size
=====
Installing:
  mysql-community-client      x86_64    5.7.44-1.el7      mysql57-community  31 M
Installing dependencies:
  mysql-community-common      x86_64    5.7.44-1.el7      mysql57-community  313 k
  mysql-community-libs        x86_64    5.7.44-1.el7      mysql57-community  3.0 M
  ncurses-compat-libs        x86_64    6.2-4.20200222.amzn2023.0.5      amazonlinux  323 k
=====
Transaction Summary
=====
Install 4 Packages

Total download size: 35 M
Installed size: 135 M
Downloading Packages:
(1/4): mysql-community-common-5.7.44-1.el7.x86_64.rpm      5.7 MB/s | 313 kB  00:00
(2/4): mysql-community-libs-5.7.44-1.el7.x86_64.rpm        47 MB/s | 3.0 MB  00:00
(3/4): ncurses-compat-libs-6.2-4.20200222.amzn2023.0.5.x86_64.rpm  2.4 MB/s | 323 kB  00:00
(4/4): mysql-community-client-5.7.44-1.el7.x86_64.rpm      53 MB/s | 31 MB   00:00
=====
Total
MySQL 5.7 Community Server
Importing GPG key 0x5072E1F5:
  Userid : MySQL Release Engineering <mysql-build@ss.oracle.com>
  Fingerprint: AUA9 4068 76FC BD3C 4567 70C8 8C71 8D3B 5072 EIF5
  From   : /etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
Key imported successfully
Import of key(s) didn't help, wrong key(s)?
Public key for mysql-community-client-5.7.44-1.el7.x86_64.rpm is not installed. Failing package is: mysql-community-client-5.7.44-1.el7.x86_64
GPG Keys are configured as: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
Public key for mysql-community-common-5.7.44-1.el7.x86_64.rpm is not installed. Failing package is: mysql-community-common-5.7.44-1.el7.x86_64
GPG Keys are configured as: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
Public key for mysql-community-libs-5.7.44-1.el7.x86_64.rpm is not installed. Failing package is: mysql-community-libs-5.7.44-1.el7.x86_64
GPG Keys are configured as: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-mysql
The downloaded packages were saved in cache until the next successful transaction.
You can remove cached packages by executing 'yum clean packages'.
Error: GPG: check FAILED
[ec2-user@ip-10-20-1-102 ~]$
```

Activate Windows
Go to Settings to activate Windows.

31°C Sunny 07-03-2024 12:53

Now will connect to our RDS-

Command--- Mysql -host (RDS end point) -port 3306 -u root(username) -p And it's done

```
[ec2-user@ip-10-20-1-102:~] Verifying : perl-B-1.80-477.amzn2023.0.6.x86_64          13/21
Verifying : perl-Data-Dumper-2.174-460.amzn2023.0.2.x86_64          14/21
Verifying : mysql-selinux-1.0-4-2.amzn2023.0.3.noarch          15/21
Verifying : perl-Math-BigInt-1;1.9998.18-458.amzn2023.0.2.noarch 16/21
Verifying : perl-File-Copy-2.34-477.amzn2023.0.6.noarch          17/21
Verifying : perl-base-2.27-477.amzn2023.0.6.noarch          18/21
Verifying : mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch 19/21
Verifying : perl-Math-Complex-1.59-477.amzn2023.0.6.noarch          20/21
Verifying : perl-FileHandle-2.03-477.amzn2023.0.6.noarch          21/21
=====
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:
  Version 2023.3.20240304:
    Run the following command to upgrade to 2023.3.20240304:
      dnf upgrade --releasever=2023.3.20240304

  Release notes:
    https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====

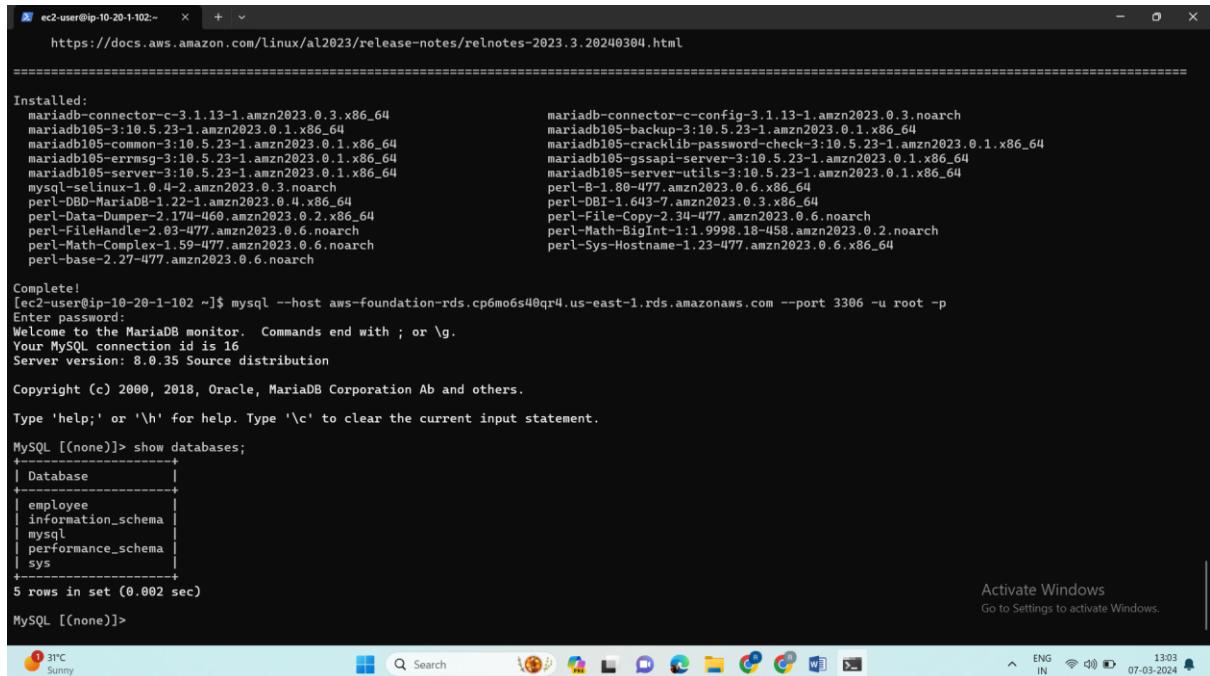
Installed:
  mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64          mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch
  mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64          mariadb105-backup-3:10.5.23-1.amzn2023.0.1.x86_64
  mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64          mariadb105-cracklib-password-check-3:10.5.23-1.amzn2023.0.1.x86_64
  mariadb105-errno-3:10.5.23-1.amzn2023.0.1.x86_64          mariadb105-gssapi-server-3:10.5.23-1.amzn2023.0.1.x86_64
  mariadb105-server-3:10.5.23-1.amzn2023.0.1.x86_64          mariadb105-server-utils-3:10.5.23-1.amzn2023.0.1.x86_64
  mysql-selinux-1.0-4-2.amzn2023.0.3.noarch          perl-B-1.80-477.amzn2023.0.6.x86_64
  perl-DBD-MariaDB-1.22-1.amzn2023.0.4.x86_64          perl-DBT-1.643-7.amzn2023.0.3.x86_64
  perl-Data-Dumper-2.174-460.amzn2023.0.2.x86_64          perl-File-Copy-2.34-477.amzn2023.0.6.noarch
  perl-FileHandle-2.03-477.amzn2023.0.6.noarch          perl-Math-BigInt-1;1.9998.18-458.amzn2023.0.2.noarch
  perl-Math-Complex-1.59-477.amzn2023.0.6.noarch          perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64
  perl-base-2.27-477.amzn2023.0.6.noarch

Complete!
[ec2-user@ip-10-20-1-102 ~]$ mysql --host aws-foundation-rds.cp6mo6s40qr4.us-east-1.rds.amazonaws.com --port 3306 -u root -p
Enter password: |
```

Activate Windows
Go to Settings to activate Windows.

31°C Sunny 07-03-2024 13:00

Command--- Show databases;



```
ec2-user@ip-10-20-1-102:~ + - x https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====
Installed:
mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64
mariadb105-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-common-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-errmsg-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-server-3.10.5.23-1.amzn2023.0.1.x86_64
mysql-selinux-1.0.4-2.amzn2023.0.3.noarch
perl-DBD-MariaDB-1.22-1.amzn2023.0.4.x86_64
perl-Data-Dumper-2.174-466.amzn2023.0.2.x86_64
perl-FileHandle-2.03-477.amzn2023.0.6.noarch
perl-Math-Complex-1.59-477.amzn2023.0.6.noarch
perl-base-2.27-477.amzn2023.0.6.noarch
=====
mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch
mariadb105-backup-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-cscklib-password-check-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-gssapi-server-3.10.5.23-1.amzn2023.0.1.x86_64
mariadb105-server-utils-3.10.5.23-1.amzn2023.0.1.x86_64
perl-B-1.80-477.amzn2023.0.6.x86_64
perl-DBI-1.643-7.amzn2023.0.3.x86_64
perl-File-Copy-2.34-477.amzn2023.0.6.noarch
perl-Math-BigInt-1.1.9998.18-458.amzn2023.0.2.noarch
perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64
=====
Complete!
[ec2-user@ip-10-20-1-102 ~]$ mysql --host aws-foundation-rds.cp6mo6s40qr4.us-east-1.rds.amazonaws.com --port 3306 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| employee |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.002 sec)

MySQL [(none)]>

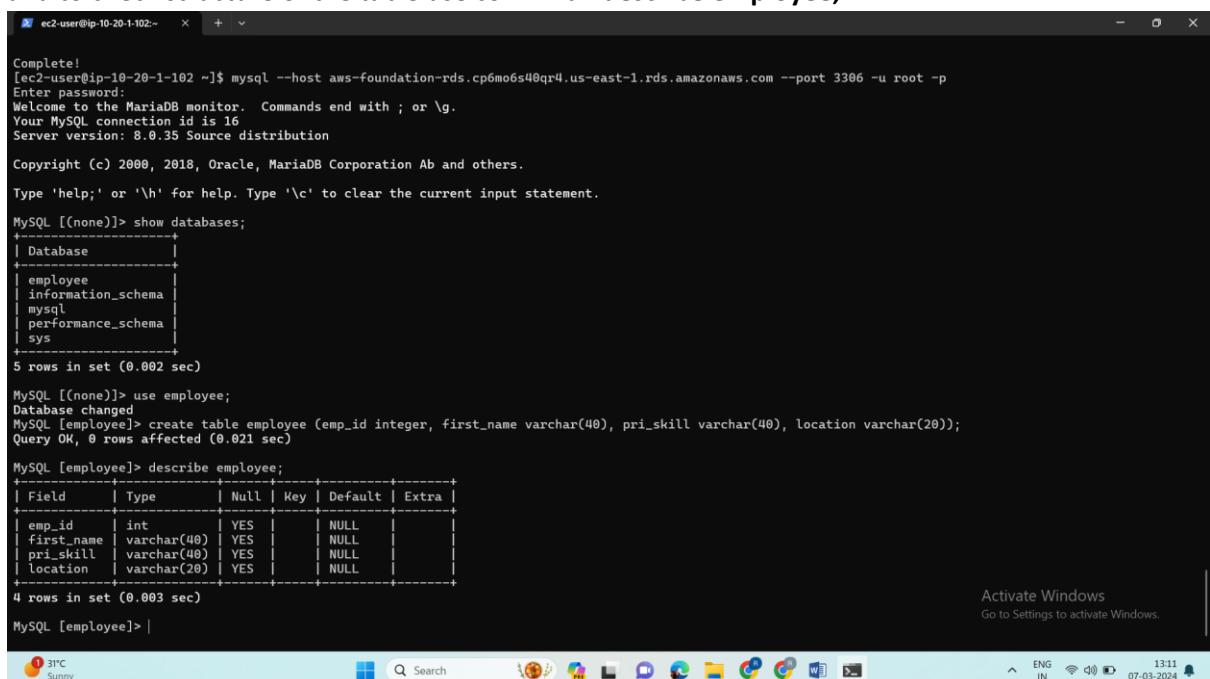
```

As we have already created employee database so will connect to this employee database and create a table

Command--- Use employee;

Then to create table command-- **create table employee (emp_id integer, first_name varchar(40), pri_skill varchar(40), location varchar(20));**

and to check structure of the table use commnd- **describe employee;**



```
ec2-user@ip-10-20-1-102:~ + - x https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====
Complete!
[ec2-user@ip-10-20-1-102 ~]$ mysql --host aws-foundation-rds.cp6mo6s40qr4.us-east-1.rds.amazonaws.com --port 3306 -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| employee |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.002 sec)

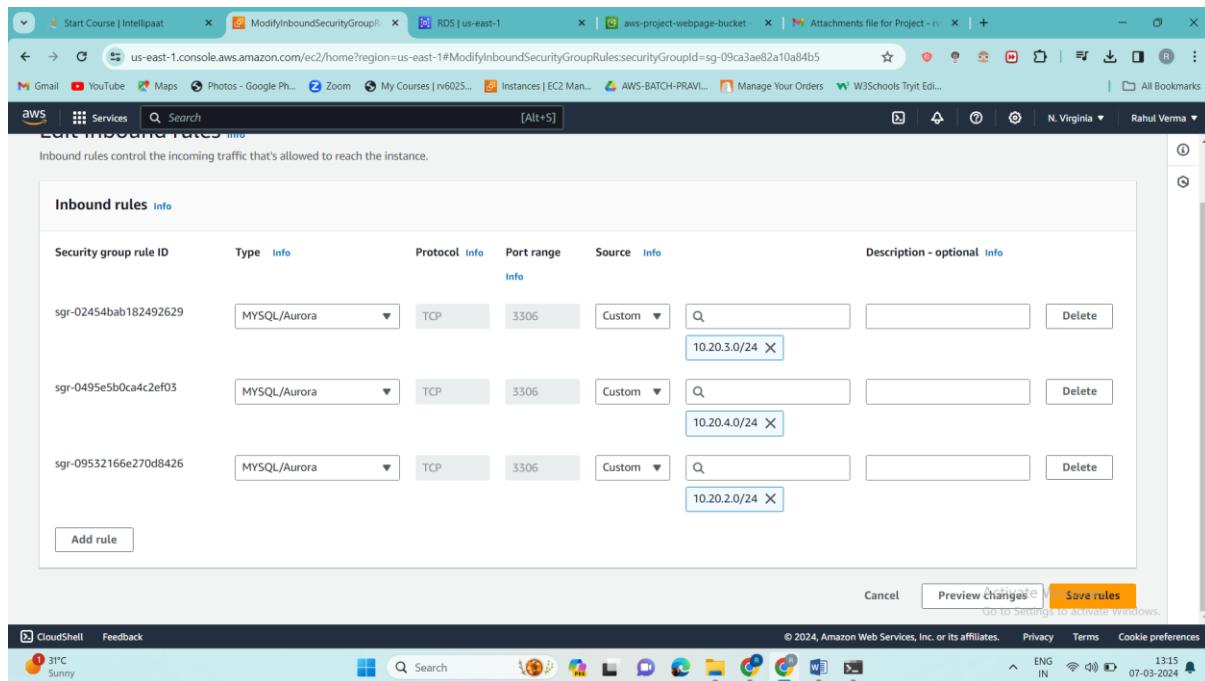
MySQL [(none)]> use employee;
Database changed
MySQL [employee]> create table employee (emp_id integer, first_name varchar(40), pri_skill varchar(40), location varchar(20));
Query OK, 0 rows affected (0.021 sec)

MySQL [employee]> describe employee;
+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+
| emp_id | int | YES | | NULL |
| first_name | varchar(40) | YES | | NULL |
| pri_skill | varchar(40) | YES | | NULL |
| location | varchar(20) | YES | | NULL |
+-----+-----+-----+-----+-----+
4 rows in set (0.003 sec)

MySQL [employee]>

```

now we have to edit security group of RDS we have to remove public subnet CIDR

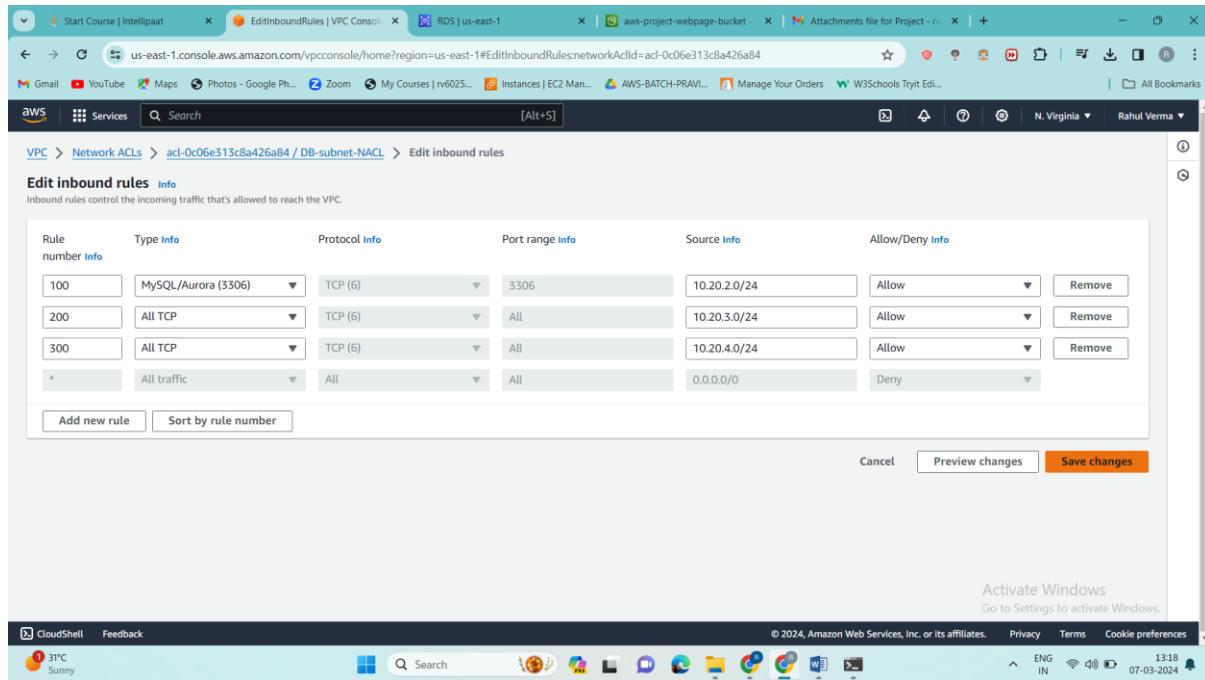


The screenshot shows the 'Edit inbound rules' page for an RDS security group. There are three rules listed:

Security group rule ID	Type	Protocol	Port range	Source	Description
sgr-02454bab182492629	MySQL/Aurora	TCP	3306	Custom (10.20.3.0/24)	
sgr-0495e5b0ca4c2ef03	MySQL/Aurora	TCP	3306	Custom (10.20.4.0/24)	
sgr-09532166e270d8426	MySQL/Aurora	TCP	3306	Custom (10.20.2.0/24)	

Buttons at the bottom include 'Add rule', 'Cancel', 'Preview changes', and a prominent orange 'Save rules' button.

And same with db subnet network ACL change both inbound & outbound (remove public CIDR 10.20.1.0/24)



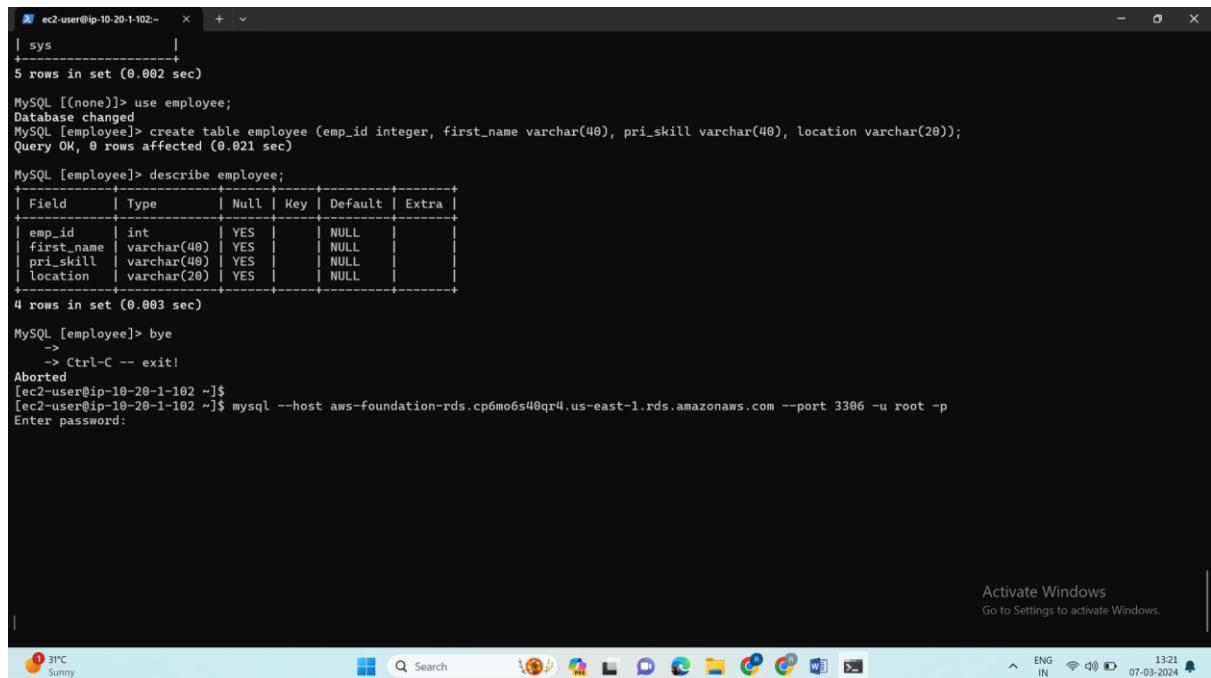
The screenshot shows the 'Edit inbound rules' page for a Network ACL (NACL). There is one rule listed:

Rule number	Type	Protocol	Port range	Source	Allow/Deny
100	MySQL/Aurora (3306)	TCP (6)	3306	10.20.2.0/24	Allow
200	All TCP	TCP (6)	All	10.20.3.0/24	Allow
300	All TCP	TCP (6)	All	10.20.4.0/24	Allow
*	All traffic	All	All	0.0.0.0/0	Deny

Buttons at the bottom include 'Add new rule', 'Sort by rule number', 'Cancel', 'Preview changes', and a prominent orange 'Save changes' button.

Now should not be able to connect to RDS via ec2 instance let's check now

And it's not connecting now



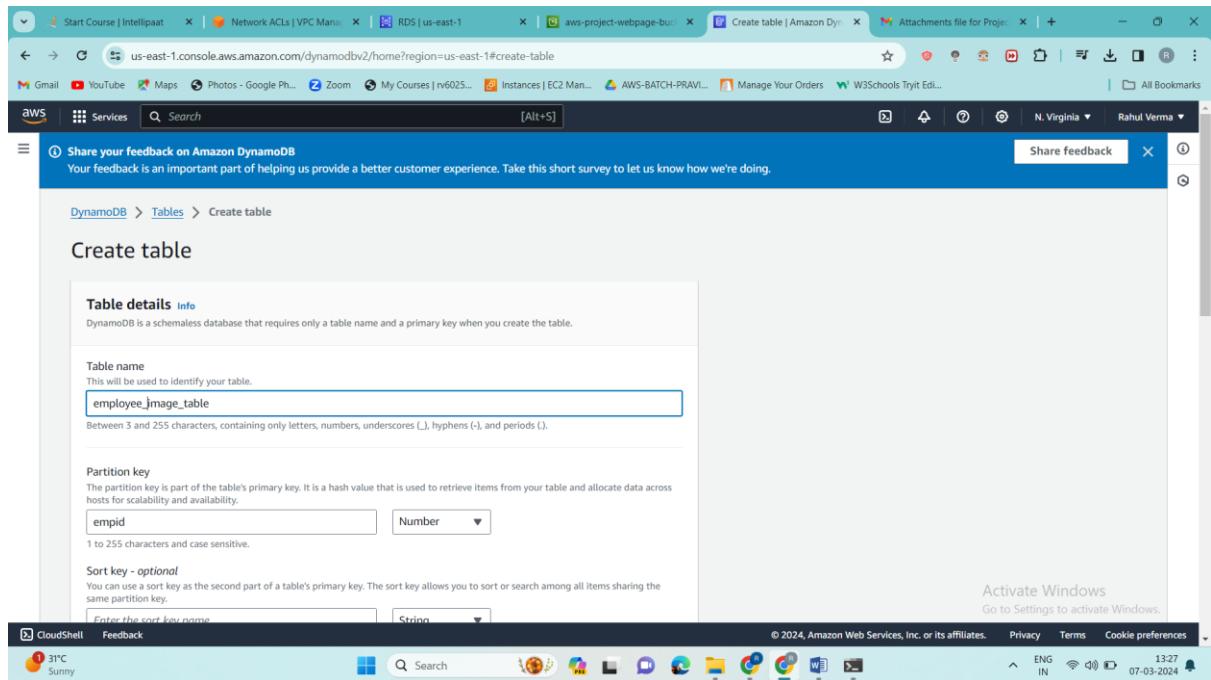
```
ec2-user@ip-10-20-1-102:~ | sys
5 rows in set (0.002 sec)

MySQL [(none)]> use employee;
Database changed
MySQL [employee]> create table employee (emp_id integer, first_name varchar(40), pri_skill varchar(40), location varchar(20));
Query OK, 0 rows affected (0.021 sec)

MySQL [employee]> describe employee;
+-----+-----+-----+-----+-----+
| Field | Type  | Null | Key | Default |
+-----+-----+-----+-----+-----+
| emp_id | int   | YES  |     | NULL    |
| first_name | varchar(40) | YES  |     | NULL    |
| pri_skill | varchar(40) | YES  |     | NULL    |
| location | varchar(20) | YES  |     | NULL    |
+-----+-----+-----+-----+-----+
4 rows in set (0.003 sec)

MySQL [employee]> bye
->
-> Ctrl-C -- exit!
Aborted
[ec2-user@ip-10-20-1-102 ~]$ [ec2-user@ip-10-20-1-102 ~]$ mysql --host aws-foundation-rds.cp6mo6s40qr4.us-east-1.rds.amazonaws.com --port 3306 -u root -p
Enter password:
```

Now according to our steps we have to create a DynamoDb table which will use to host the s3 meta data



Start Course | IntelliJ IDEA | Network ACLs | VPC Manager | RDS | us-east-1 | aws-project-webpage-build | Create table | Amazon DynamoDB | Attachments file for Project | +

us-east-1.console.aws.amazon.com/dynamodbv2/home?region=us-east-1#create-table

DynamoDB > Tables > Create table

Create table

Table details Info

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name

This will be used to identify your table.

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.)

Partition key

The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

1 to 255 characters and case sensitive.

Sort key - optional

You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

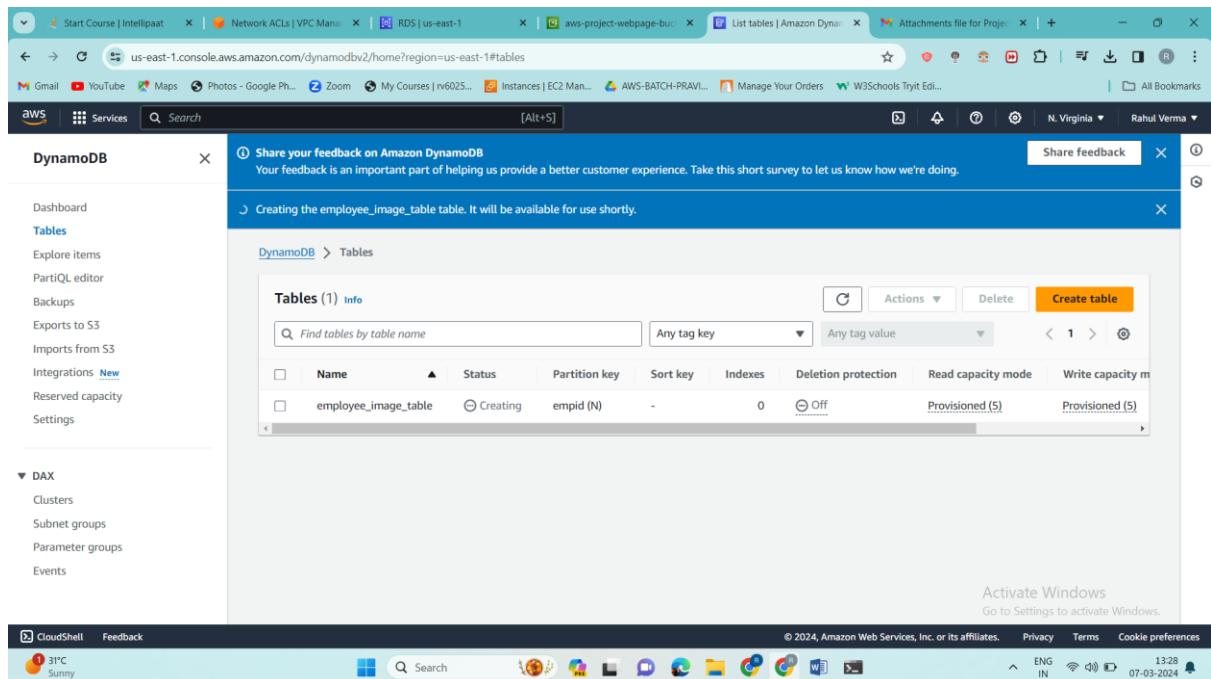
Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

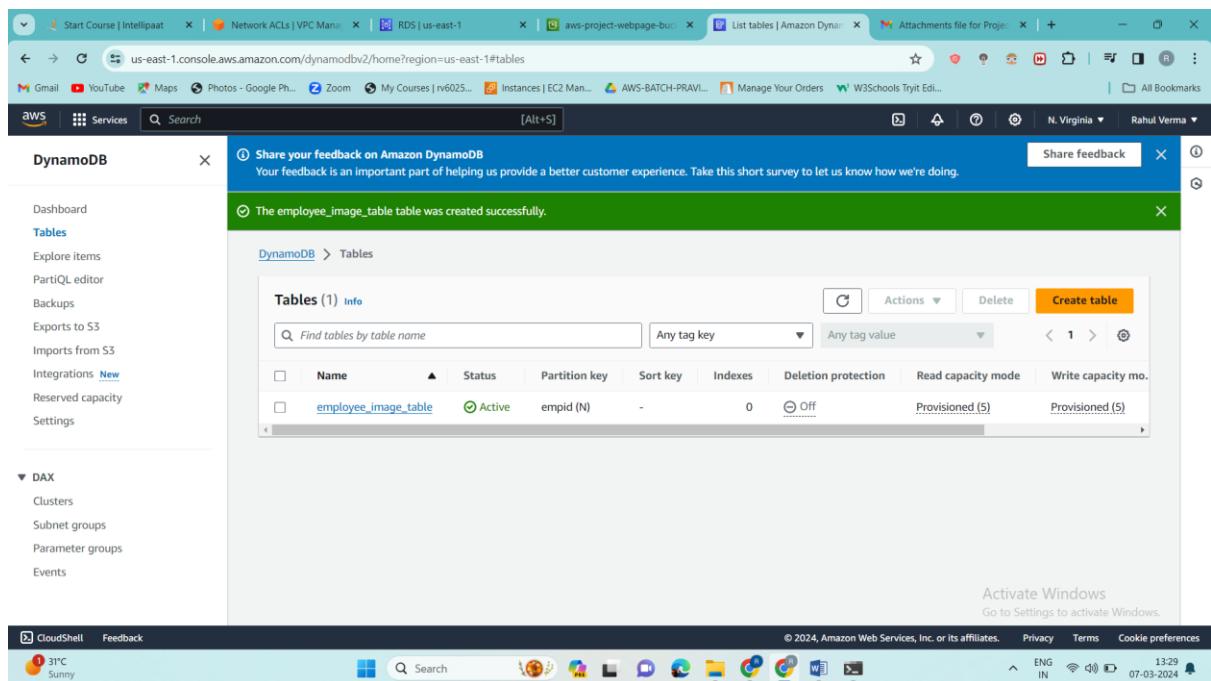
13:27 07-03-2024

Rest default settings and just click it on create button



The screenshot shows the AWS DynamoDB console. On the left, a sidebar menu is open with 'Tables' selected. The main area shows a table named 'employee_image_table' with the status 'Creating'. A modal window is open at the top, asking for feedback on the service. The status bar at the bottom right shows the date as 07-03-2024 and the time as 13:28.

And it's created successfully



The screenshot shows the AWS DynamoDB console. The table 'employee_image_table' now has a status of 'Active'. A green success message at the top of the main area states 'The employee_image_table table was created successfully.' The status bar at the bottom right shows the date as 07-03-2024 and the time as 13:29.

Now we have to setup our domain name for that I am going to use BIGROCK site and we have this xyzcorp.in.net

Now let's jump on to route 53 console in aws

Will create one hosted zone domain name is same xyzcorp.in.net

Hosted zone configuration

A hosted zone is a container that holds information about how you want to route traffic for a domain, such as example.com, and its subdomains.

Domain name [Info](#)
This is the name of the domain that you want to route traffic for.
xyzcorp.in.net

Valid characters: a-z, 0-9, ! * # \$ % & ' () ^ + , - / ; ; < = > ? @ [\] ^ _ ` { } , ~

Description - optional [Info](#)
This value lets you distinguish hosted zones that have the same name.
The hosted zone is used for...

The description can have up to 256 characters. 0/256

Type [Info](#)
The type indicates whether you want to route traffic on the internet or in an Amazon VPC.
 Public hosted zone
A public hosted zone determines how traffic is routed on the internet.
 Private hosted zone
A private hosted zone determines how traffic is routed within an Amazon VPC.

xyzcorp.in.net was successfully created.

Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.

Now we have to change NS records in our bigrock site

Route 53

xyzcorp.in.net was successfully created.

Now you can create records in the hosted zone to specify how you want Route 53 to route traffic for your domain.

Records (2) [Info](#)

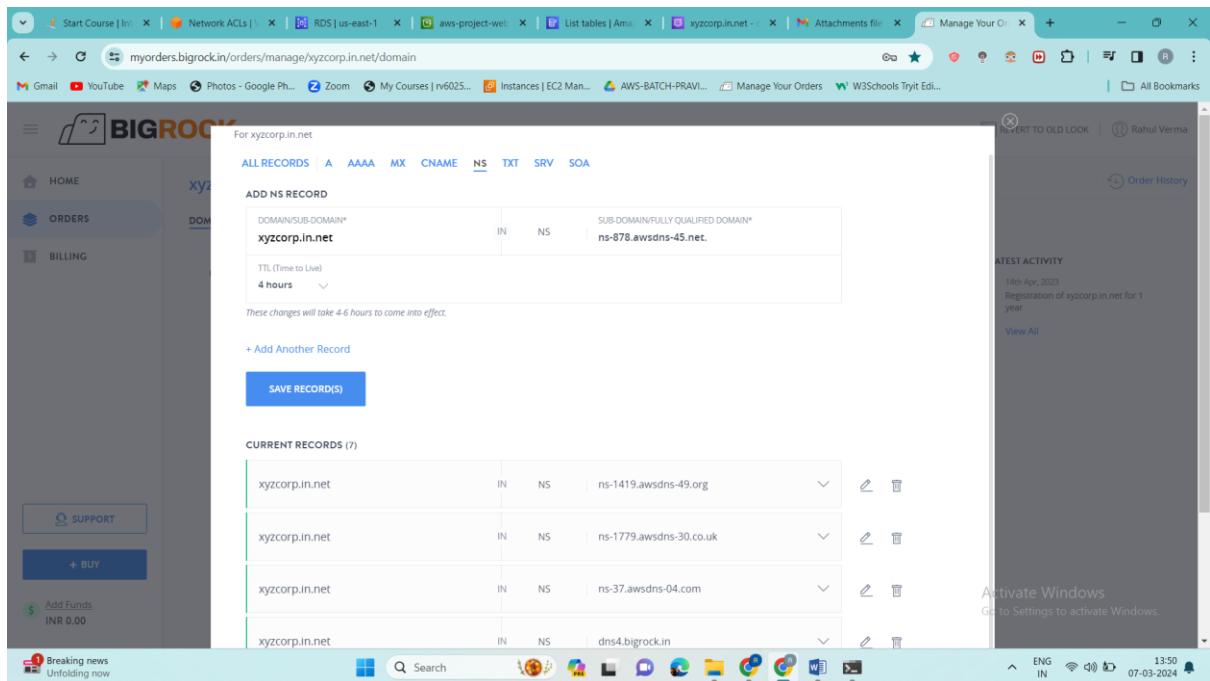
Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

Record ...	Type	Routing policy	Differ...	Alias	Value/Route traffic to	TTL (s...)	Health ...	Eva...
xyzcorp.in...	NS	Simple	-	No	ns-37.awsdns-04.com	172800	-	-
xyzcorp.in...	SOA	Simple	-	No	ns-1779.awsdns-30.co.uk.	ns-1419.awsdns-49.org.	ns-878.awsdns-45.net.	900

0 records selected

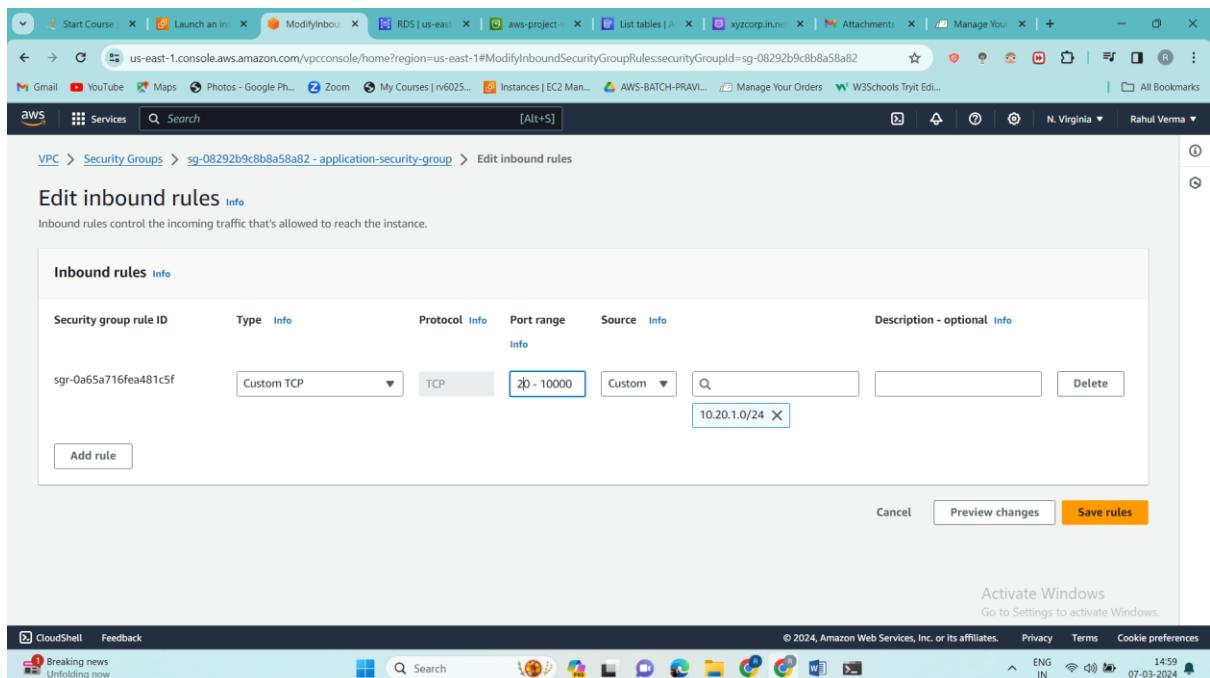
Activate Windows
Go to Settings to activate Windows.

This how one by one we have to add NS server in our bigrock site



The screenshot shows the BigRock website's DNS management interface. A new NS record is being added for the domain xyzcorp.in.net, pointing to ns-878.awsdns-45.net with a TTL of 4 hours. The current records table shows other NS records for the domain.

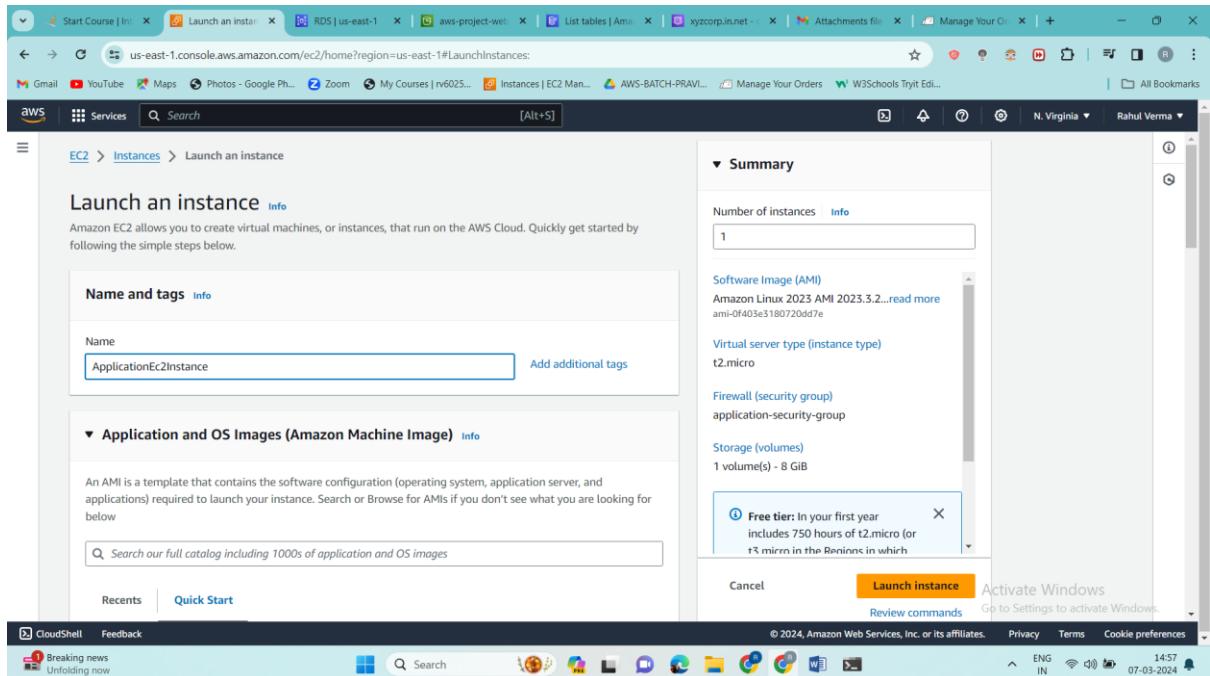
Now we have to edit our app security group change it to 20-10000 (otherwise will not be able to ssh into it)



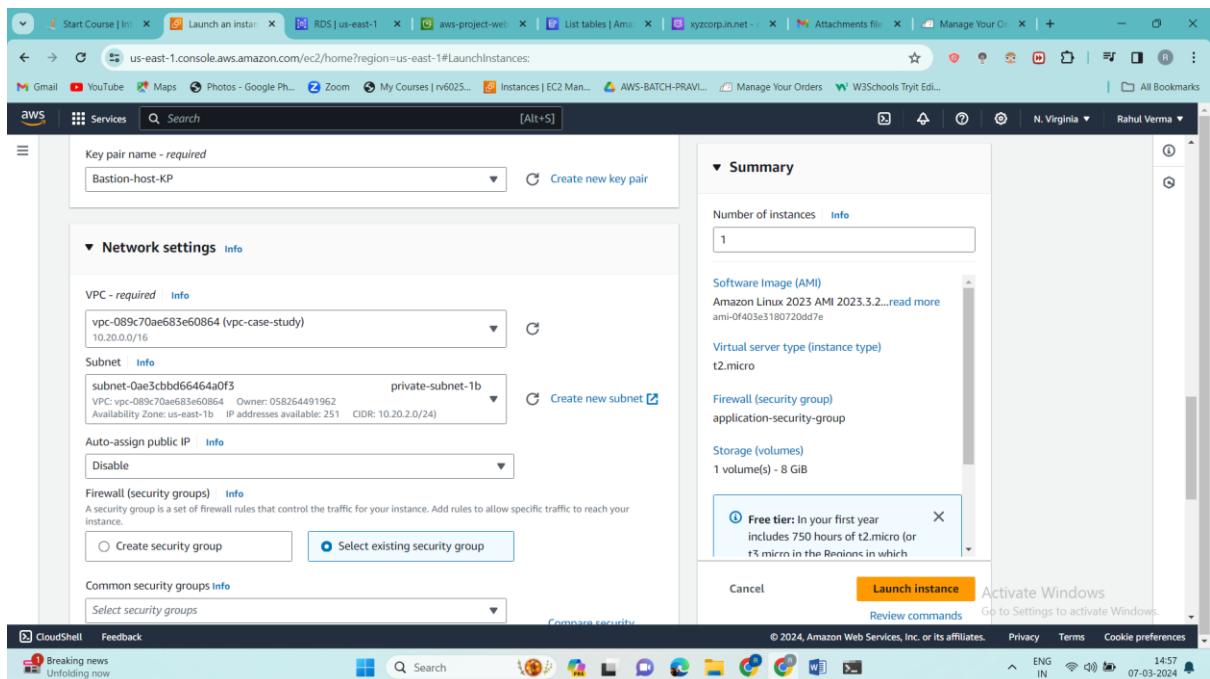
The screenshot shows the AWS VPC console's edit inbound rules interface. A new rule is being added for TCP port 20-10000 from 10.20.1.0/24. The rule is currently in preview mode.

Now according to our next step we have to create AMI using ec2 instance

And we have to setup our application also



Select our VPC, select private subnet 1b,existing SG- app security group & disable public ip



And our instance is created

The screenshot shows the AWS EC2 Instances page. The left sidebar is expanded, showing the following navigation structure:

- EC2 Dashboard
- EC2 Global View
- Events
- Console-to-Code
- Instances** (selected)
- Instances
- Instance Types
- Launch Templates
- Spot Requests
- Savings Plans
- Reserved Instances
- Dedicated Hosts
- Capacity Reservations
- New**

Images

- AMIs
- AMI Catalog

Elastic Block Store

- Volumes

At the bottom of the sidebar, there are links for CloudShell and Feedback, along with a system status bar showing 33°C and Sunny.

The main content area displays the following table for Instances (2):

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
Bastion-Host	i-0b79f050d340370a6	Running	t2.micro	2/2 checks passed	View alarms	us-east-1a
ApplicationEc2Instance	i-0dd85c29b4b612217	Pending	t2.micro	-	View alarms	us-east-1b

A modal window titled "Select an instance" is open, indicating that an instance needs to be chosen for further actions.

Command to copy keypair (note- we have to do this before login to our bastion host)

```
scp -i Bastion-host-KP.pem Bastion-host-KP.pem ec2-user@107.23.245.167:/home/ec2-user
```

and now login to our instance command—**ls** | and let's ssh our application ec2

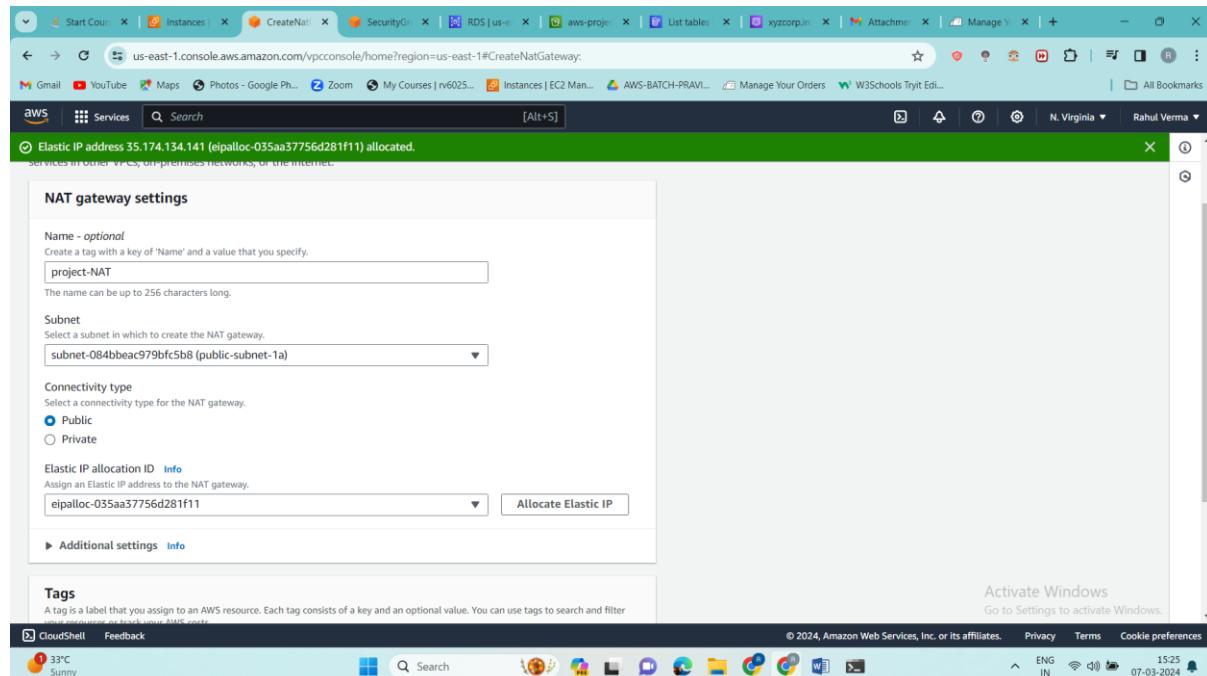
using this command- `ssh -i Bastion-host-KP.pem ec2-user@10.20.2.62`

Python was not installed

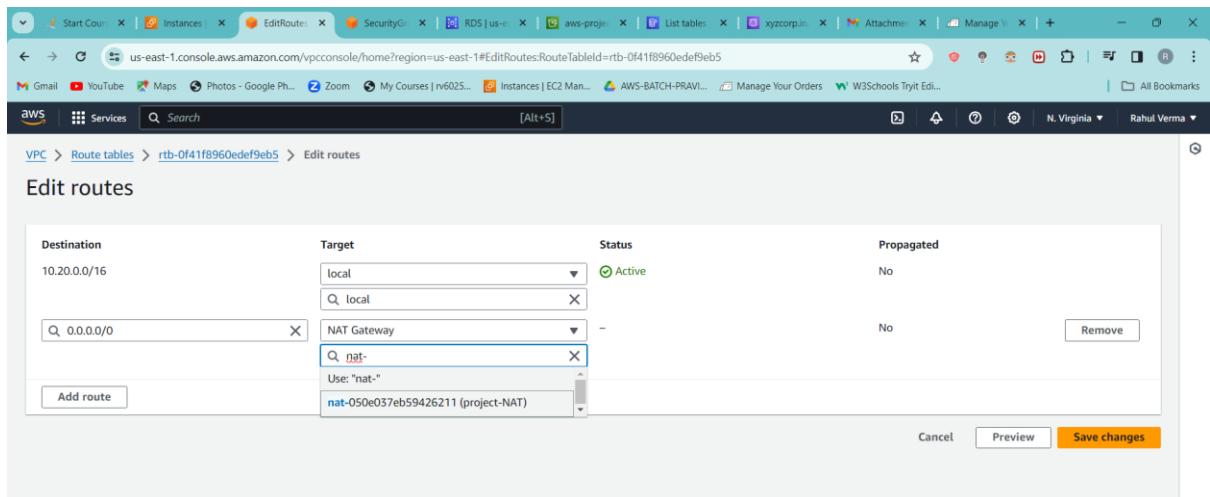
So let's install it we will not be able to do that because it does not have internet so for that we have to create NAT gateway

```
ec2-user@ip-10-20-1-102:~ + - PS C:\Users\Tnluser\Downloads> ssh -i "C:\Users\Tnluser\Downloads\Bastion-host-KP.pem" ec2-user@107.23.245.167
A newer release of "Amazon Linux" is available.
Version 2023.3.20240304:
Run "/usr/bin/dnf check-release-update" for full release and version update info
#
`_\_ ##### Amazon Linux 2023
~~ \#####\
~~ \###|
~~ \#/ ___ https://aws.amazon.com/linux/amazon-linux-2023
~~ \~-->
~~ \_/
~~ \_/
~~ \_/
`_m'
Last login: Thu Mar  7 09:35:58 2024 from 117.213.197.183
[ec2-user@ip-10-20-1-102 ~]$ ls -l
total 4
-rw-rw-r--. 1 ec2-user ec2-user 1674 Mar  7 09:40 Bastion-host-KP.pem
[ec2-user@ip-10-20-1-102 ~]$ ssh -i Bastion-host-KP.pem ec2-user@10.20.2.62
The authenticity of host '10.20.2.62' (10.20.2.62) can't be established.
ED25519 key fingerprint is SHA256:Ec0LmvzcbEonlTVvZ3WQh12qnallXkD2LAm6tickKSwg.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.20.2.62' (ED25519) to the list of known hosts.
@@@@@@@@@
@       WARNING: UNPROTECTED PRIVATE KEY FILE! @
@@@@@@@@@
Permissions 0664 for 'Bastion-host-KP.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key 'Bastion-host-KP.pem': bad permissions
ec2-user@10.20.2.62: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-20-1-102 ~]$ python --version
-bash: python: command not found
[ec2-user@ip-10-20-1-102 ~]$ |
```

We have to select our public subnet which is having internet access and will allocate elastic IP



And now we have to change route table of our private subnet

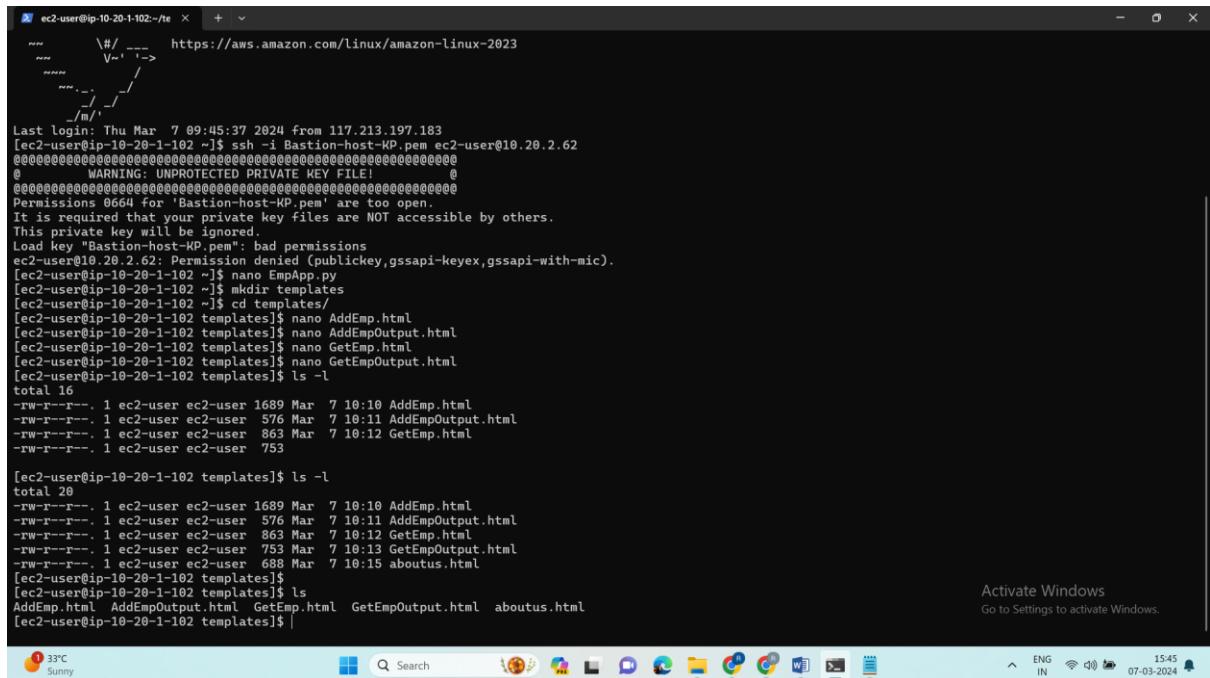


Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 33°C Sunny ENG IN 15:28 07-03-2024

And meanwhile NAT gateway is getting ready will add some dependencies

EmpApp.py, AddEmp.html, AddEmpOutput.html, GetEmp.html, GetEmpOutput.html & aboutus.html

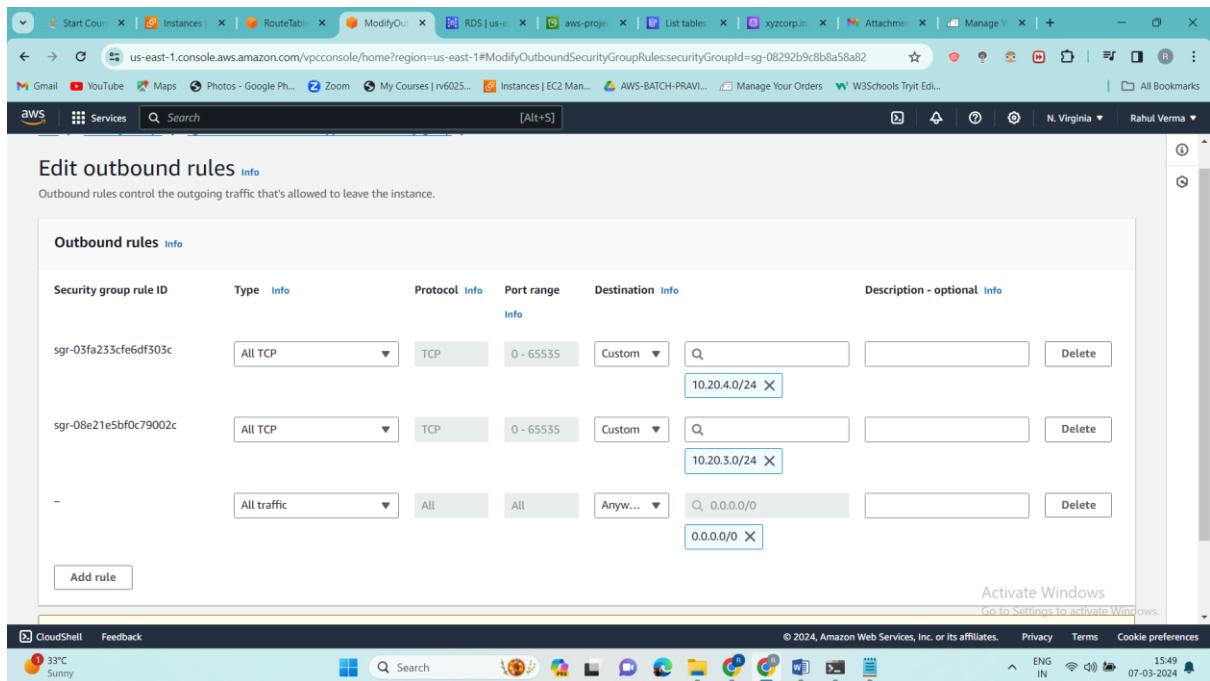


```
ec2-user@ip-10-20-1-102:~$ ssh -i Bastion-host-KP.pem ec2-user@10.20.2.62
Last login: Thu Mar  7 09:45:37 2024 from 117.213.197.183
[ec2-user@ip-10-20-1-102 ~]$ ssh -i Bastion-host-KP.pem ec2-user@10.20.2.62
[ec2-user@ip-10-20-1-102 ~]$ cd templates
[ec2-user@ip-10-20-1-102 templates]$ nano EmpApp.py
[ec2-user@ip-10-20-1-102 templates]$ nano AddEmp.html
[ec2-user@ip-10-20-1-102 templates]$ nano AddEmpOutput.html
[ec2-user@ip-10-20-1-102 templates]$ nano GetEmp.html
[ec2-user@ip-10-20-1-102 templates]$ nano GetEmpOutput.html
[ec2-user@ip-10-20-1-102 templates]$ ls -l
total 16
-rw-r--r--. 1 ec2-user ec2-user 1689 Mar  7 10:10 AddEmp.html
-rw-r--r--. 1 ec2-user ec2-user 576 Mar  7 10:11 AddEmpOutput.html
-rw-r--r--. 1 ec2-user ec2-user 863 Mar  7 10:12 GetEmp.html
-rw-r--r--. 1 ec2-user ec2-user 753 Mar  7 10:13 GetEmpOutput.html
-rw-r--r--. 1 ec2-user ec2-user 688 Mar  7 10:15 aboutus.html
[ec2-user@ip-10-20-1-102 templates]$ ls
AddEmp.html AddEmpOutput.html GetEmp.html GetEmpOutput.html aboutus.html
[ec2-user@ip-10-20-1-102 templates]$ |
```

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 33°C Sunny ENG IN 15:45 07-03-2024

We have to edit outbound rule of our app security group also



Outbound rules [Info](#)

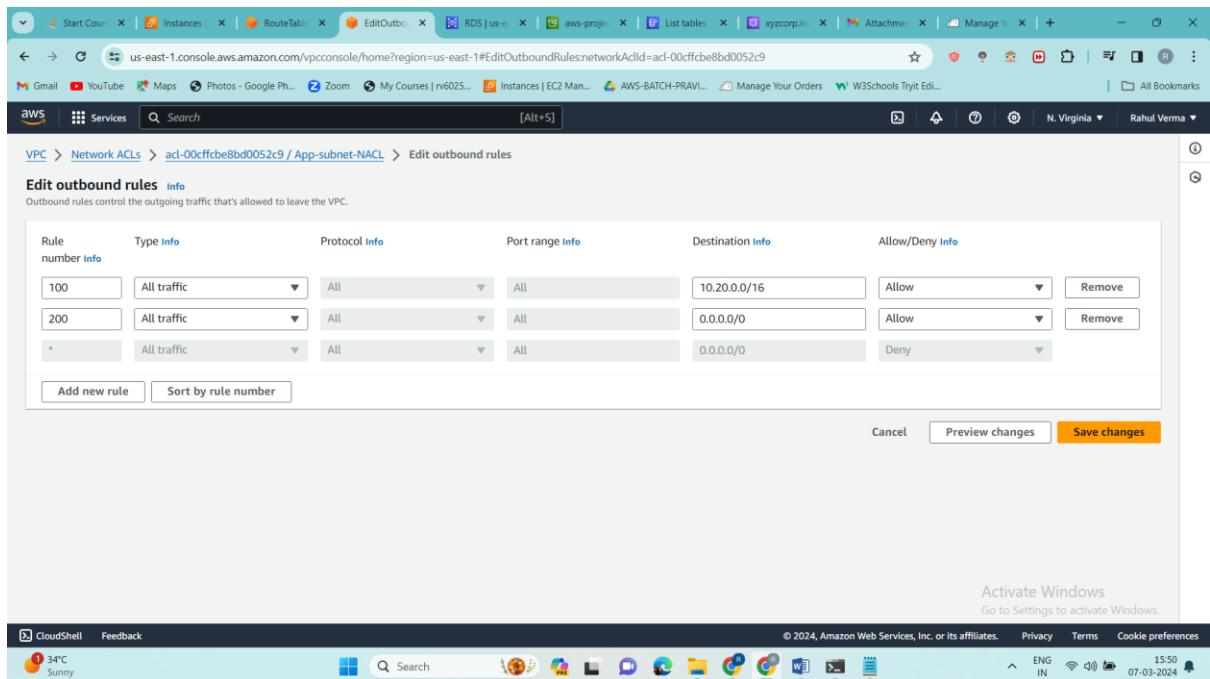
Outbound rules control the outgoing traffic that's allowed to leave the instance.

Security group rule ID	Type Info	Protocol Info	Port range Info	Destination Info	Description - optional Info
sgr-03fa233cfe6df303c	All TCP	TCP	0 - 65535	Custom Info	<input type="text" value="10.20.4.0/24"/> Delete
sgr-08e21e5bf0c79002c	All TCP	TCP	0 - 65535	Custom Info	<input type="text" value="10.20.3.0/24"/> Delete
-	All traffic	All	All	Any... Info	<input type="text" value="0.0.0.0/0"/> Delete

[Add rule](#)

Activate Windows
Go to Settings to activate Windows

And also with NACL outbound rule



VPC > Network ACLs > acl-00cffcbe8bd0052c9 / App-subnet-NACL > Edit outbound rules

Edit outbound rules [Info](#)

Outbound rules control the outgoing traffic that's allowed to leave the VPC.

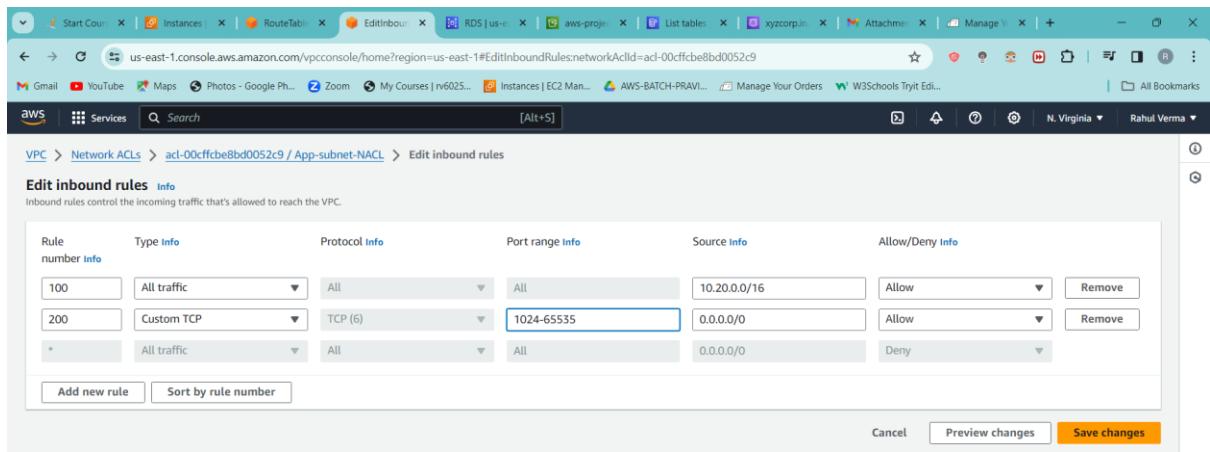
Rule number Info	Type Info	Protocol Info	Port range Info	Destination Info	Allow/Deny Info
100	All traffic	All	All	10.20.0.0/16	Allow Remove
200	All traffic	All	All	0.0.0.0/0	Allow Remove
*	All traffic	All	All	0.0.0.0/0	Deny Remove

[Add new rule](#) [Sort by rule number](#)

[Cancel](#) [Preview changes](#) [Save changes](#)

Activate Windows
Go to Settings to activate Windows

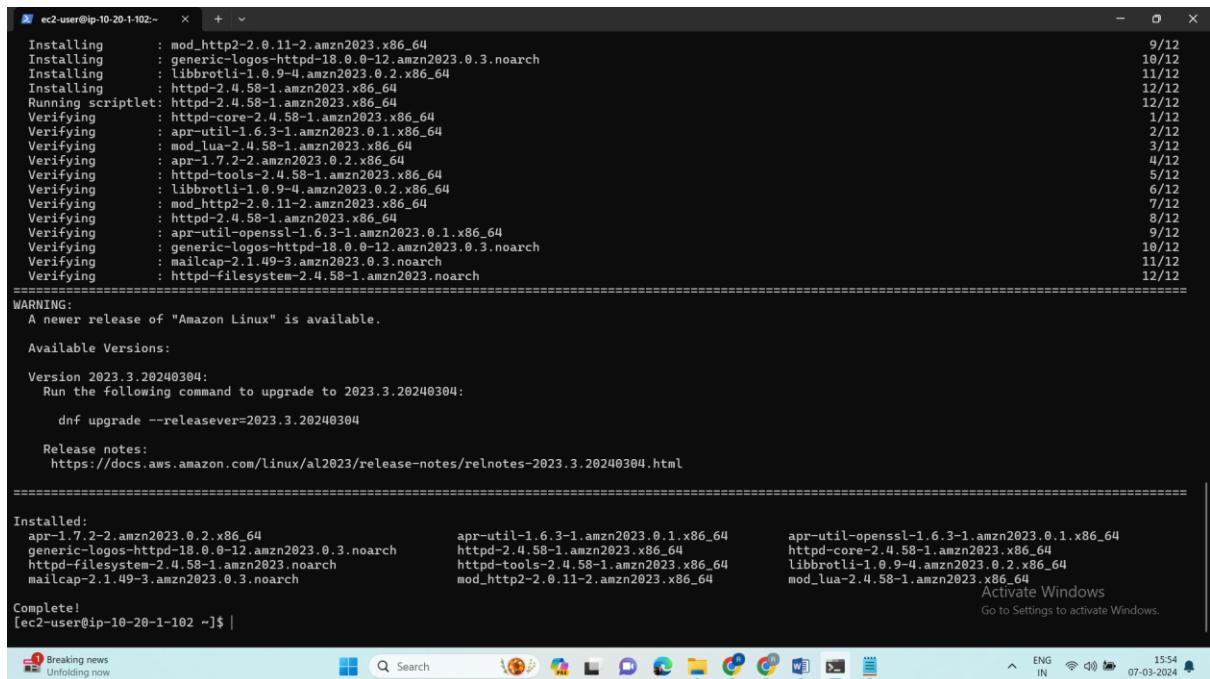
And in NACL inbound will change



Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 15:53 07-03-2024

And now internet is working in our private instance which is application instance



```
Installing : mod_http2-2.0.11-2.amzn2023.x86_64 9/12
Installing : generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch 10/12
Installing : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 11/12
Installing : httpd-2.4.58-1.amzn2023.x86_64 12/12
Running scriptlet: httpd-2.4.58-1.amzn2023.x86_64 12/12
Verifying : httpd-core-2.4.58-1.amzn2023.x86_64 1/12
Verifying : apr-util-1.6.3-1.amzn2023.0.1.x86_64 2/12
Verifying : mod_lua-2.4.58-1.amzn2023.x86_64 3/12
Verifying : apr-1.7.2-2.amzn2023.0.2.x86_64 4/12
Verifying : httpd-tools-2.4.58-1.amzn2023.x86_64 5/12
Verifying : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 6/12
Verifying : mod_http2-2.0.11-2.amzn2023.x86_64 7/12
Verifying : httpd-2.4.58-1.amzn2023.x86_64 8/12
Verifying : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64 9/12
Verifying : generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch 10/12
Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch 11/12
Verifying : httpd-filesystem-2.4.58-1.amzn2023.noarch 12/12
=====
WARNING:
A newer release of "Amazon Linux" is available.

Available Versions:
Version 2023.3.20240304:
Run the following command to upgrade to 2023.3.20240304:
dnf upgrade --releasever=2023.3.20240304

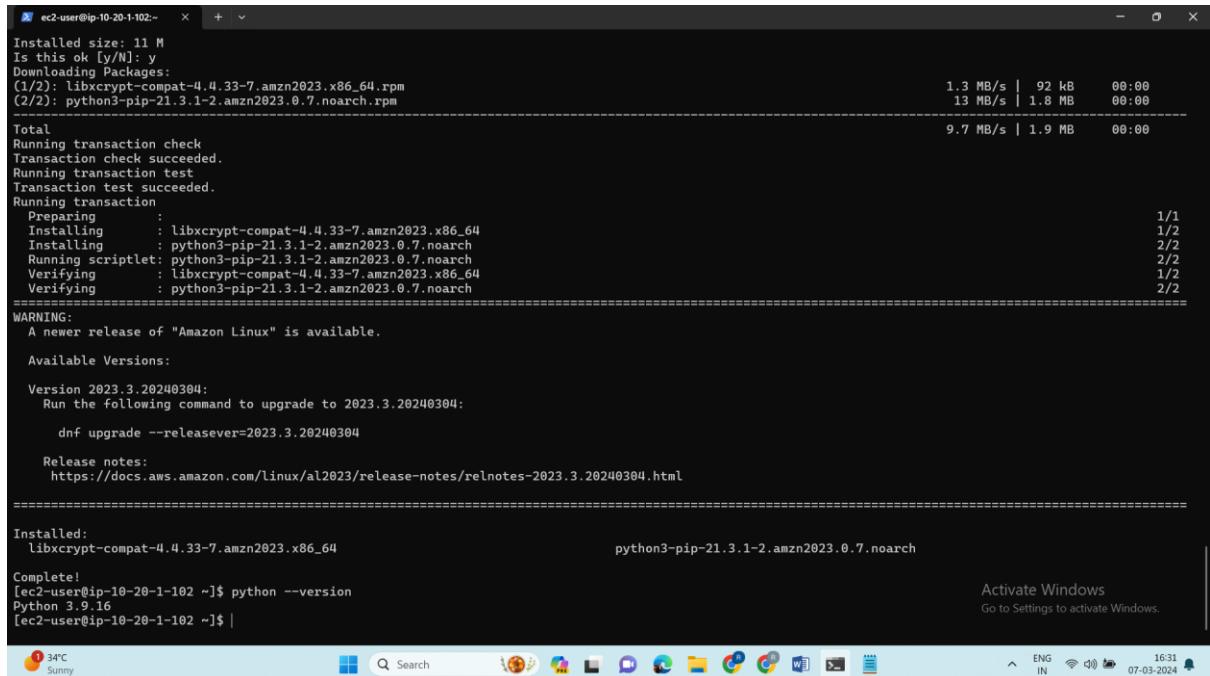
Release notes:
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====
Installed:
apr-1.7.2-2.amzn2023.0.2.x86_64      apr-util-1.6.3-1.amzn2023.0.1.x86_64      apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httd-18.0.0-12.amzn2023.0.3.noarch  httpd-2.4.58-1.amzn2023.x86_64      httpd-core-2.4.58-1.amzn2023.x86_64
httpd-filesystem-2.4.58-1.amzn2023.noarch  httpd-tools-2.4.58-1.amzn2023.x86_64      libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch      mod_http2-2.0.11-2.amzn2023.x86_64      mod_lua-2.4.58-1.amzn2023.x86_64
=====
Complete!
[ec2-user@ip-10-20-1-102 ~]$
```

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 15:54 07-03-2024

Now let's install python command- **sudo yum install python**

Sudo yum install python-pip



```
ec2-user@ip-10-20-1-102:~ x + | v
Installed size: 11 M
Is this ok [y/N]: y
Downloading Packages:
(1/2): libcrypt-compat-4.4.33-7.amzn2023.x86_64.rpm           1.3 MB/s | 92 kB   00:00
(2/2): python3-pip-21.3.1-2.amzn2023.0.7.noarch.rpm           13 MB/s | 1.8 MB   00:00
Total                                               9.7 MB/s | 1.9 MB   00:00
=====
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing : 1/1
  Installing : libcrypt-compat-4.4.33-7.amzn2023.x86_64           1/2
  Installing : python3-pip-21.3.1-2.amzn2023.0.7.noarch          2/2
  Running scriptlet: python3-pip-21.3.1-2.amzn2023.0.7.noarch
  Verifying   : libcrypt-compat-4.4.33-7.amzn2023.x86_64           2/2
  Verifying   : python3-pip-21.3.1-2.amzn2023.0.7.noarch          1/2
=====
=====
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:
  Version 2023.3.20240304:
    Run the following command to upgrade to 2023.3.20240304:
      dnf upgrade --releasever=2023.3.20240304

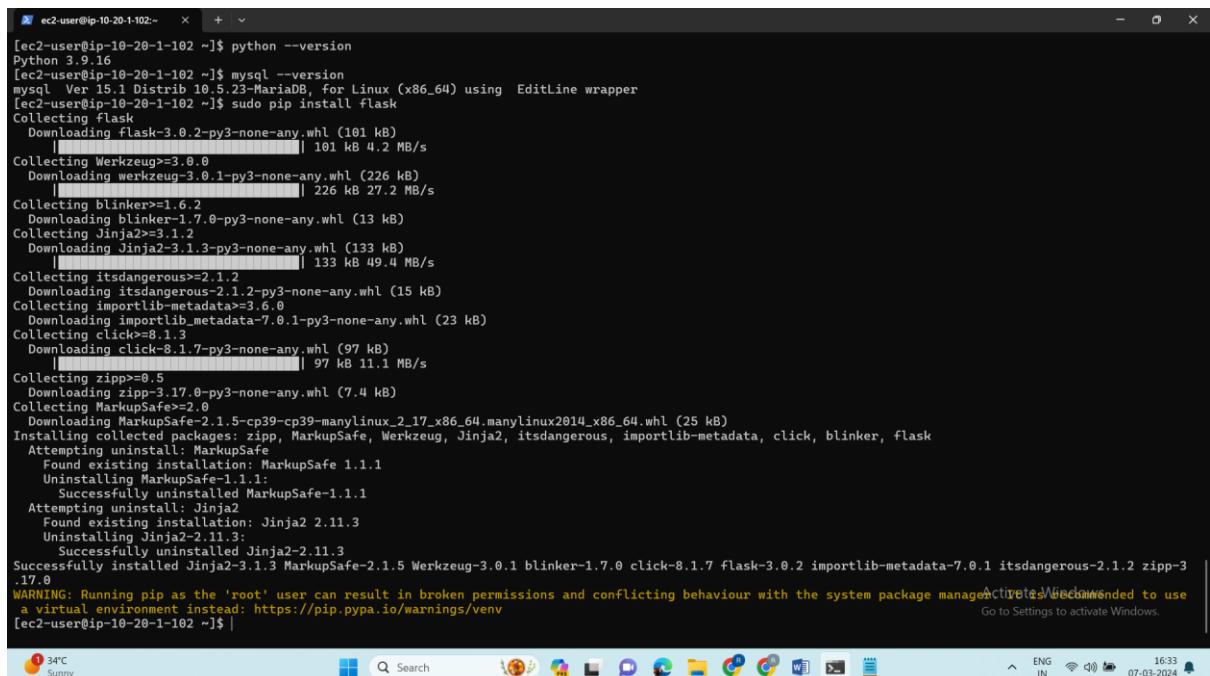
  Release notes:
    https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes-2023.3.20240304.html
=====

Installed:
  libcrypt-compat-4.4.33-7.amzn2023.x86_64                      python3-pip-21.3.1-2.amzn2023.0.7.noarch

Complete!
[ec2-user@ip-10-20-1-102 ~]$ python --version
Python 3.9.16
[ec2-user@ip-10-20-1-102 ~]$ |
```

Now we have to install flask

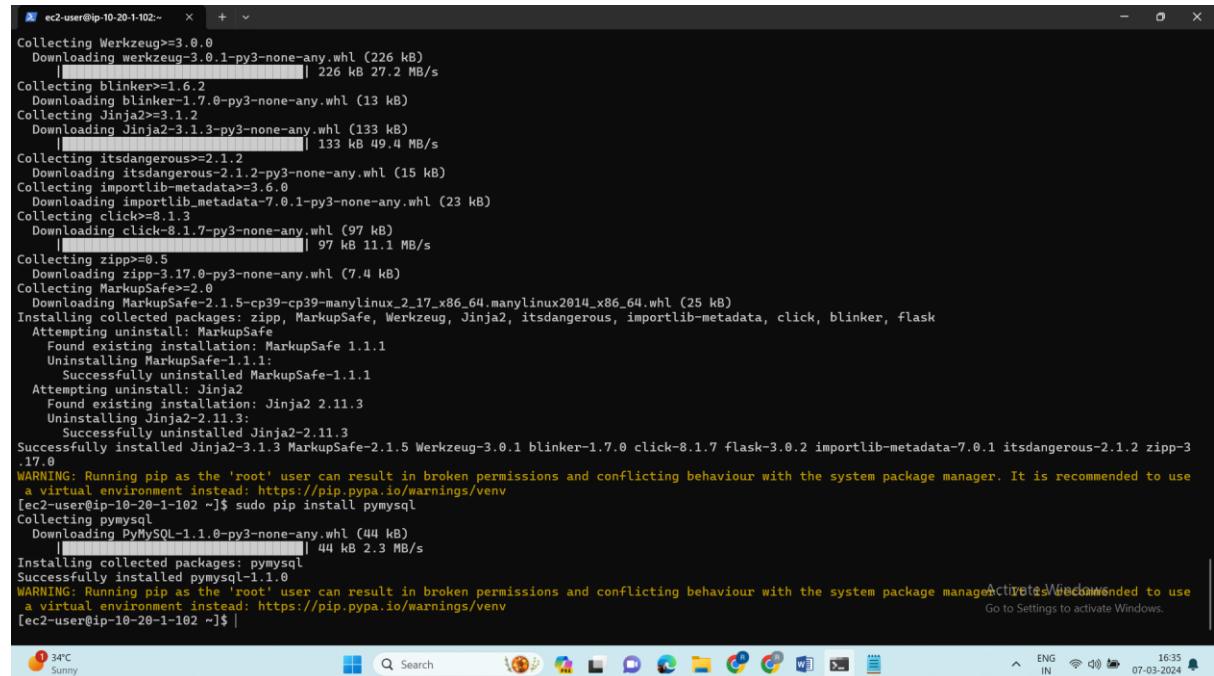
Command- sudo pip install flask



```
ec2-user@ip-10-20-1-102:~ x + | v
[ec2-user@ip-10-20-1-102 ~]$ python --version
Python 3.9.16
[ec2-user@ip-10-20-1-102 ~]$ mysql --version
mysql Ver 15.1 Distrib 10.5.23-MariaDB, for Linux (x86_64) using EditLine wrapper
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install flask
Collecting flask
  Downloading flask-3.0.2-py3-none-any.whl (101 kB)
Collecting Werkzeug==3.0.0
  Downloading werkzeug-3.0.1-py3-none-any.whl (226 kB)
Collecting blinker>=1.6.2
  Downloading blinker-1.7.0-py3-none-any.whl (13 kB)
Collecting Jinja2>=3.1.2
  Downloading Jinja2-3.1.3-py3-none-any.whl (133 kB)
Collecting itsdangerous>=2.1.2
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting importlib-metadata>=3.6.0
  Downloading importlib_metadata-3.6.1-py3-none-any.whl (23 kB)
Collecting click>=8.1.3
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
Collecting zipp>=0.5
  Downloading zipp-3.17.0-py3-none-any.whl (7.4 kB)
Collecting MarkupSafe==2.0
  Downloading MarkupSafe-2.1.5-cp39-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Installing collected packages: zipp, MarkupSafe, Werkzeug, Jinja2, itsdangerous, importlib-metadata, click, blinker, flask
  Attempting uninstall: MarkupSafe
    Found existing installation: MarkupSafe 1.1.1
    Uninstalling MarkupSafe-1.1.1:
      Successfully uninstalled MarkupSafe-1.1.1
  Attempting uninstall: Jinja2
    Found existing installation: Jinja2 2.11.3
    Uninstalling Jinja2-2.11.3:
      Successfully uninstalled Jinja2-2.11.3
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 importlib-metadata-3.6.1 itsdangerous-2.1.2 zipp-3.17.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is highly recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ |
```

And we have to install pymysql also

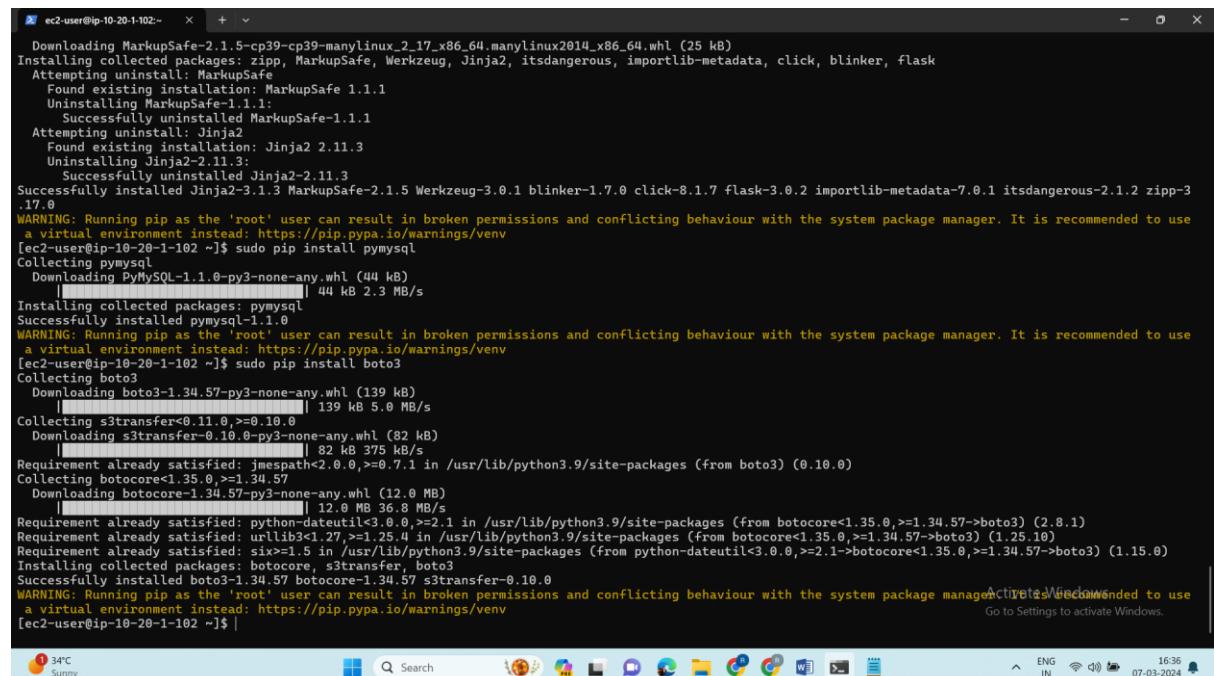
Command- **sudo pip install pymysql**



```
Collecting Werkzeug>=3.0.0
  Downloading werkzeug-3.0.1-py3-none-any.whl (226 kB)
Collecting blinker>=1.6.2
  Downloading blinker-1.7.0-py3-none-any.whl (13 kB)
Collecting Jinja2>=3.1.2
  Downloading Jinja2-3.1.3-py3-none-any.whl (133 kB)
Collecting itsdangerous>=2.1.2
  Downloading itsdangerous-2.1.2-py3-none-any.whl (15 kB)
Collecting importlib-metadata>=3.6.0
  Downloading importlib_metadata-7.0.1-py3-none-any.whl (23 kB)
Collecting click>=8.1.3
  Downloading click-8.1.7-py3-none-any.whl (97 kB)
Collecting zipp>=0.5
  Downloading zipp-3.17.0-py3-none-any.whl (7.4 kB)
Collecting MarkupSafe>=2.0
  Downloading MarkupSafe-2.1.5-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (25 kB)
Installing collected packages: zipp, MarkupSafe, Werkzeug, Jinja2, itsdangerous, importlib-metadata, click, blinker, flask
  Attempting uninstall: MarkupSafe
    Found existing installation: MarkupSafe 1.1.1
    Uninstalling MarkupSafe-1.1.1:
      Successfully uninstalled MarkupSafe-1.1.1
  Attempting uninstall: Jinja2
    Found existing installation: Jinja2 2.11.3
    Uninstalling Jinja2-2.11.3:
      Successfully uninstalled Jinja2-2.11.3
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 importlib-metadata-7.0.1 itsdangerous-2.1.2 zipp-3.17.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install pymysql
Collecting pymysql
  Downloading PyMySQL-1.1.0-py3-none-any.whl (44 kB)
Installing collected packages: pymysql
Successfully installed pymysql-1.1.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$
```

And boto3 as well

Command- **sudo pip install boto3**



```
Collecting MarkupSafe-2.1.5-cp39-cp39-manylinux_2_17_x86_64_manylinux2014_x86_64.whl (25 kB)
Installing collected packages: MarkupSafe
  Attempting uninstall: MarkupSafe
    Found existing installation: MarkupSafe 1.1.1
    Uninstalling MarkupSafe-1.1.1:
      Successfully uninstalled MarkupSafe-1.1.1
  Attempting uninstall: Jinja2
    Found existing installation: Jinja2 2.11.3
    Uninstalling Jinja2-2.11.3:
      Successfully uninstalled Jinja2-2.11.3
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 importlib-metadata-7.0.1 itsdangerous-2.1.2 zipp-3.17.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install pymysql
Collecting pymysql
  Downloading PyMySQL-1.1.0-py3-none-any.whl (44 kB)
Installing collected packages: pymysql
Successfully installed pymysql-1.1.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install boto3
Collecting boto3
  Downloading boto3-1.34.57-py3-none-any.whl (139 kB)
Collecting s3transfer<0.11.0,>=0.10.8
  Downloading s3transfer-0.10.0-py3-none-any.whl (82 kB)
Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /usr/lib/python3.9/site-packages (from boto3) (0.10.0)
Collecting botocore<1.35.0,>=1.34.57
  Downloading botocore-1.34.57-py3-none-any.whl (12.0 MB)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /usr/lib/python3.9/site-packages (from botocore<1.35.0,>=1.34.57->boto3) (2.8.1)
Requirement already satisfied: urllib3<1.27,>=1.25.4 in /usr/lib/python3.9/site-packages (from botocore<1.35.0,>=1.34.57->boto3) (1.25.10)
Requirement already satisfied: six<1.5 in /usr/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.35.0,>=1.34.57->boto3) (1.15.8)
Installing collected packages: botocore, s3transfer, boto3
Successfully installed boto3-1.34.57 botocore-1.34.57 s3transfer-0.10.8
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$
```

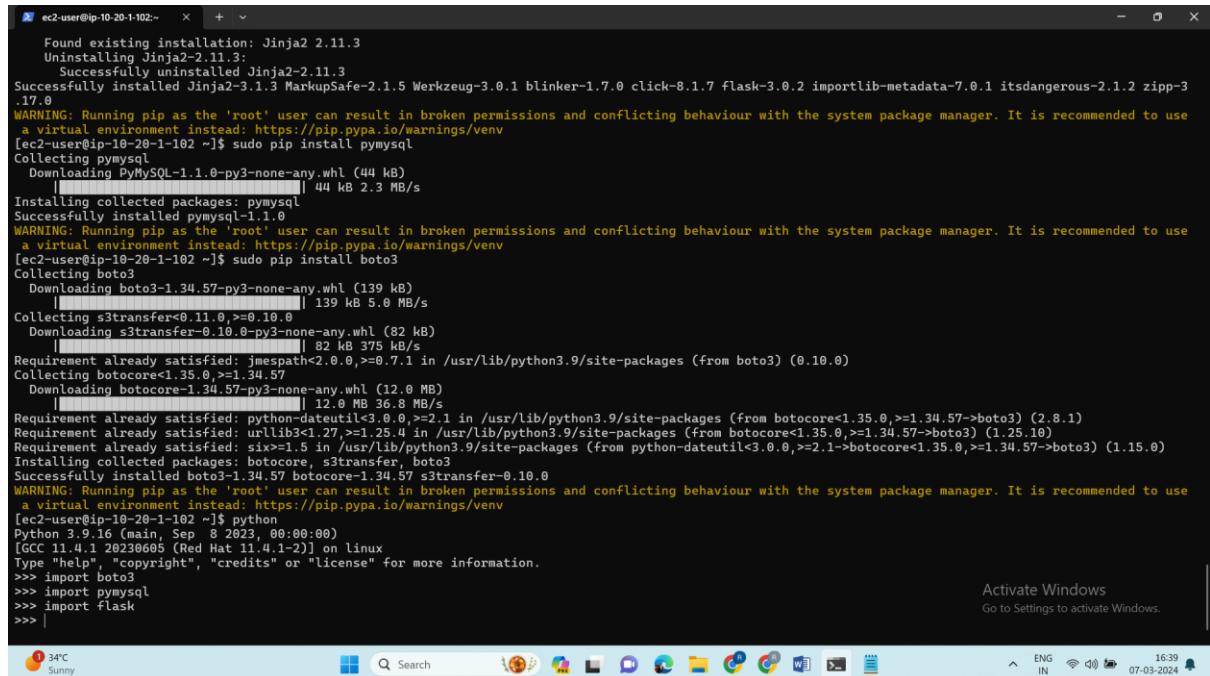
Command-

Python

```
>>>import boto3
```

```
>>>import pymysql
```

```
>>>import flask
```

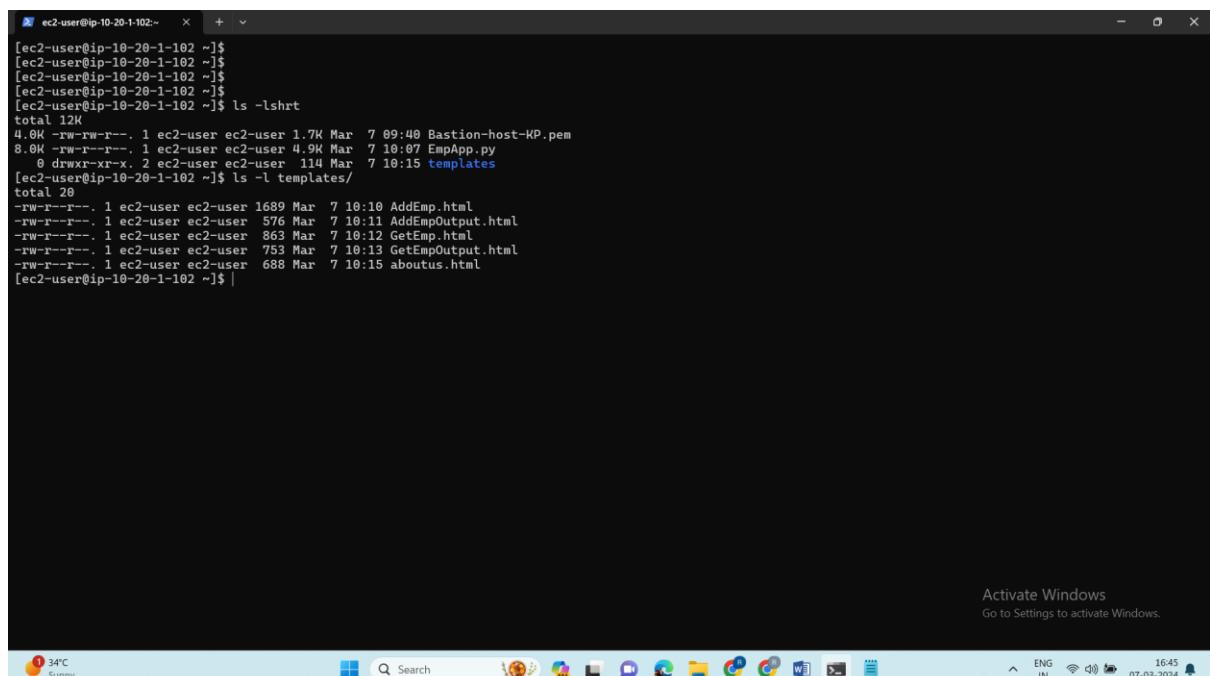


```
ec2-user@ip-10-20-1-102 ~ + 
Found existing installation: Jinja2 2.11.3
Uninstalling Jinja2-2.11.3:
  Successfully uninstalled Jinja2-2.11.3
Successfully installed Jinja2-3.1.3 MarkupSafe-2.1.5 Werkzeug-3.0.1 blinker-1.7.0 click-8.1.7 flask-3.0.2 importlib-metadata-7.0.1 itsdangerous-2.1.2 zipp-3.17.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install pymysql
Collecting pymysql
  Downloading PyMySQL-1.1.0-py3-none-any.whl (44 kB)
Installing collected packages: pymysql
Successfully installed pymysql-1.1.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ sudo pip install boto3
Collecting boto3
  Downloading boto3-1.34.57-py3-none-any.whl (139 kB)
Collecting s3transfer<0.11.0,>=0.10.0
  Downloading s3transfer-0.10.0-py3-none-any.whl (82 kB)
Requirement already satisfied: jmespath<2.0.0,>=0.7.1 in /usr/lib/python3.9/site-packages (from boto3) (0.10.0)
Collecting botocore<1.35.0,>=1.34.57
  Downloading botocore-1.34.57-py3-none-any.whl (12.0 MB)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /usr/lib/python3.9/site-packages (from botocore<1.35.0,>=1.34.57->boto3) (2.8.1)
Requirement already satisfied: urllib3<1.27,>=1.25.4 in /usr/lib/python3.9/site-packages (from botocore<1.35.0,>=1.34.57->boto3) (1.25.10)
Requirement already satisfied: six<=1.5 in /usr/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.35.0,>=1.34.57->boto3) (1.15.0)
Installing collected packages: botocore, s3transfer, boto3
Successfully installed boto3-1.34.57 botocore-1.34.57 s3transfer-0.10.0
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
[ec2-user@ip-10-20-1-102 ~]$ python
Python 3.9.16 (Main, Sep  8 2023, 00:00:00)
[GCC 11.4.1 20230605 (Red Hat 11.4.1-2)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import boto3
>>> import pymysql
>>> import flask
>>> |
```

Ctrl+D to exit

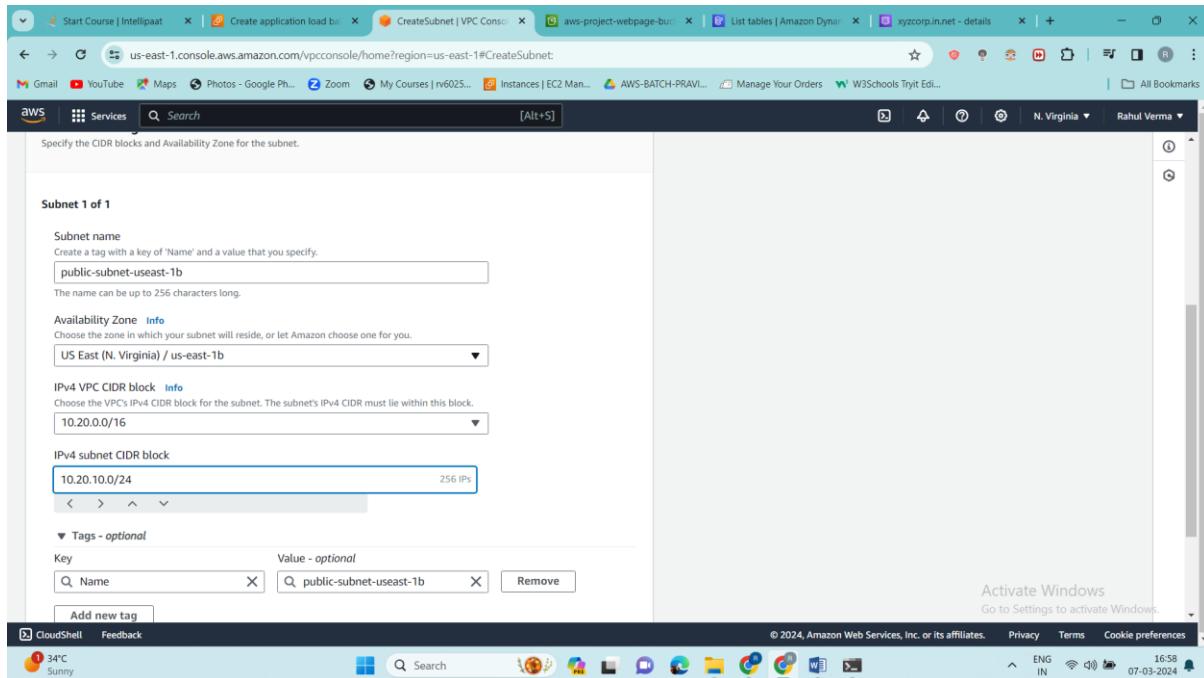
ls -lshrt

ls -l templates/



```
[ec2-user@ip-10-20-1-102 ~]$ 
[ec2-user@ip-10-20-1-102 ~]$ 
[ec2-user@ip-10-20-1-102 ~]$ 
[ec2-user@ip-10-20-1-102 ~]$ ls -lshrt
total 12K
4.0K -rw-rw-r-- 1 ec2-user ec2-user 1.7K Mar  7 09:40 Bastion-host-KP.pem
8.0K -rw-r--r-- 1 ec2-user ec2-user 4.9K Mar  7 10:07 EmpApp.py
  0 drwxr-xr-x. 2 ec2-user ec2-user 114 Mar  7 10:15 templates/
[ec2-user@ip-10-20-1-102 ~]$ ls -l templates/
total 20
-rw-r--r--. 1 ec2-user ec2-user 1689 Mar  7 10:10 AddEmp.html
-rw-r--r--. 1 ec2-user ec2-user 576 Mar  7 10:11 AddEmpOutput.html
-rw-r--r--. 1 ec2-user ec2-user 863 Mar  7 10:12 GetEmp.html
-rw-r--r--. 1 ec2-user ec2-user 753 Mar  7 10:13 GetEmpOutput.html
-rw-r--r--. 1 ec2-user ec2-user 688 Mar  7 10:15 aboutus.html
[ec2-user@ip-10-20-1-102 ~]$ |
```

We have to create one more public subnet in us-east-1b (because loadbalancer requires minimum two subnets)



Specify the CIDR blocks and Availability Zone for the subnet.

Subnet 1 of 1

Subnet name
Create a tag with a key of 'Name' and a value that you specify.

The name can be up to 256 characters long.

Availability Zone **Info**
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block **Info**
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs

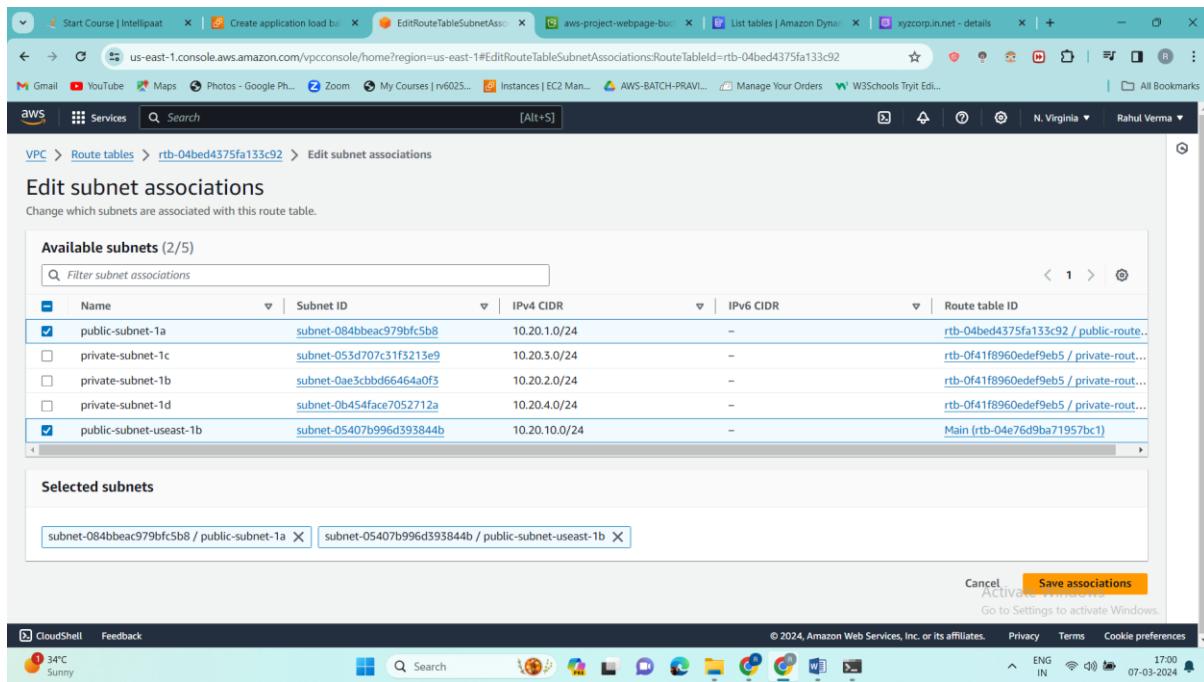
Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="public-subnet-useast-1b"/>

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 16:58 ENG IN 07-03-2024

Now we have to add this subnet to our public route table also



Edit subnet associations
Change which subnets are associated with this route table.

Available subnets (2/5)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
<input checked="" type="checkbox"/> public-subnet-1a	subnet-084bbeac979bfc5b8	10.20.1.0/24	-	rtb-04bed4375fa133c92 / public-route..
<input type="checkbox"/> private-subnet-1c	subnet-053d707c31f3213e9	10.20.3.0/24	-	rtb-0f41f8960e0def9eb5 / private-route..
<input type="checkbox"/> private-subnet-1b	subnet-0ae3cbb66464a0f3	10.20.2.0/24	-	rtb-0f41f8960e0def9eb5 / private-route..
<input type="checkbox"/> private-subnet-1d	subnet-0b454fce7052712a	10.20.4.0/24	-	rtb-0f41f8960e0def9eb5 / private-route..
<input checked="" type="checkbox"/> public-subnet-useast-1b	subnet-05407b996d393844b	10.20.10.0/24	-	Main (rtb-04e76d9ba71957bc1)

Selected subnets

Save associations

Cancel **Save associations**
Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 17:00 ENG IN 07-03-2024

And we have to add this in public NACL also

Available subnets (2/5)

Name	Subnet ID	Associated with	Availability Zone	IPv4 CIDR	IPv6 CIDR
public-subnet-useast-1b	subnet-05407b996d393844b	acl-0b55932535c781ae2	us-east-1b	10.20.10.0/24	-
public-subnet-1a	subnet-084bbeac979bfc5b8	acl-0bfe419a99926c7f3 / Public-subnet-NACL	us-east-1a	10.20.1.0/24	-
private-subnet-1c	subnet-053d707c31f3215e9	acl-0c06e313c8a426a84 / DB-5...	us-east-1c	10.20.3.0/24	-
private-subnet-1b	subnet-0ae3cbbd66464a0f3	acl-00cffcbe8bd0052c9 / App-...	us-east-1b	10.20.2.0/24	-
private-subnet-1d	subnet-0b454face7052712a	acl-0c06e313c8a426a84 / DB-5...	us-east-1d	10.20.4.0/24	-

Selected subnets

subnet-084bbeac979bfc5b8 / public-subnet-1a X subnet-05407b996d393844b / public-subnet-useast-1b X

Cancel Save changes

Activate Windows
Go to Settings to activate Windows

Now we have to create a load balancer – application loadbalancer

Compare and select load balancer type

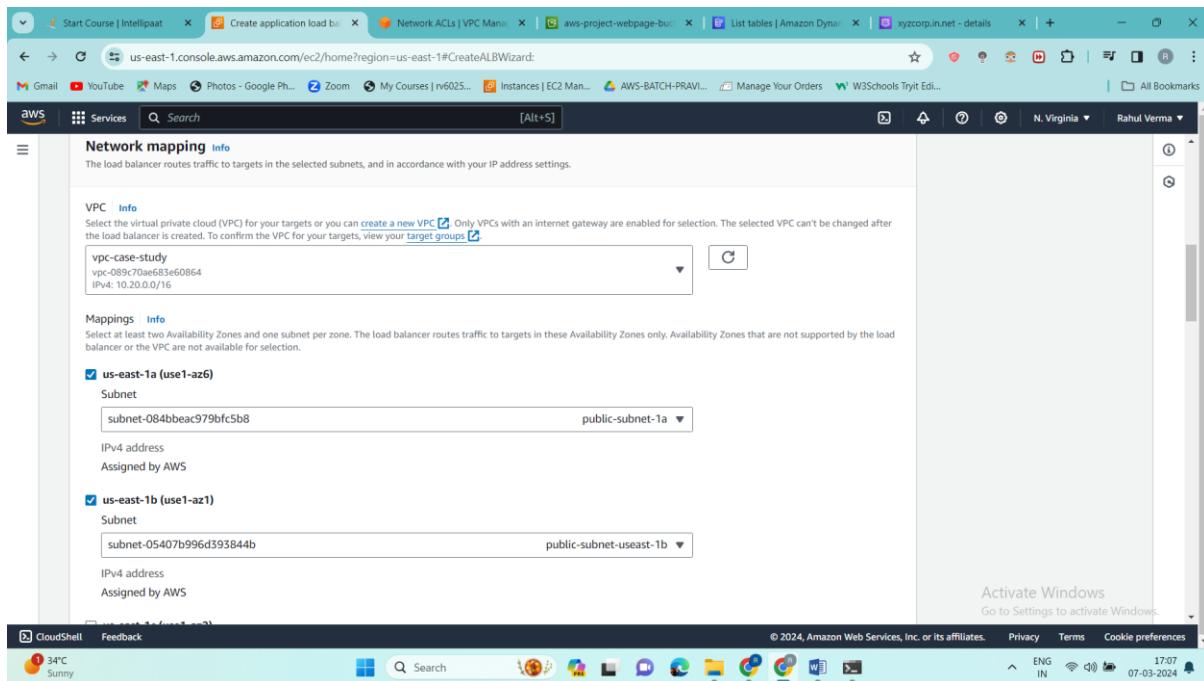
A complete feature-by-feature comparison along with detailed highlights is also available. [Learn more](#)

Load balancer types
Application Load Balancer Info
Choose an Application Load
Network Load Balancer Info
Choose a Network Load Balancer
Gateway Load Balancer Info
Choose a Gateway Load Balancer

Activate Windows
Go to Settings to activate Windows

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN 17:02 07-03-2024

Select our VPC and public subnet us-east-1a public subnet and useast-1b public subnet



Network mapping Info
The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC Info
Select the virtual private cloud (VPC) for your targets or you can [create a new VPC](#). Only VPCs with an internet gateway are enabled for selection. The selected VPC can't be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

vpc-case-study
vpc-089r70aa683e60864
IPv4: 10.20.0.0/16

Mappings Info
Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection.

us-east-1a (use1-az6)
Subnet: **subnet-084bbeac979bfc5b8** **public-subnet-1a**

IPv4 address
Assigned by AWS

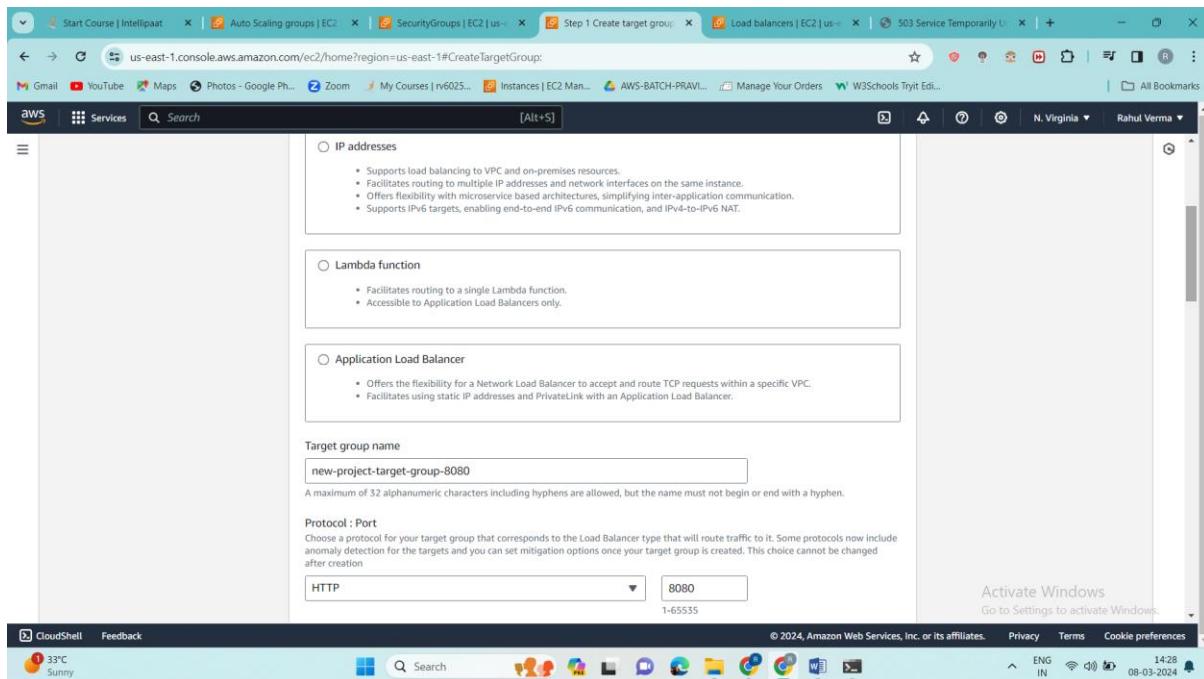
us-east-1b (use1-az1)
Subnet: **subnet-05407b996d393844b** **public-subnet-useast-1b**

IPv4 address
Assigned by AWS

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 34°C Sunny ENG IN 17:07 07-03-2024

We haven't created any target group so will create one



Start Course | Intellipaat | Create application load balancer | Network ACLs | VPC Manager | aws-project-webpage-bu... | List tables | Amazon DynamoDB | xyzcorp.in.net - details | + | us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateALBWizard: | Gmail | YouTube | Maps | Photos - Google Photos | Zoom | My Courses | rv6025... | Instances | EC2 Manager | AWS-BATCH-PRAVI... | Manage Your Orders | W3Schools TryIt Editor | All Bookmarks | N. Virginia | Rahul Verma

Services Search [Alt+S]

IP addresses

- Supports load balancing to VPC and on-premises resources.
- Facilitates routing to multiple IP addresses and network interfaces on the same instance.
- Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Supports IPv6 targets, enabling end-to-end IPv6 communication, and IPv4-to-IPv6 NAT.

Lambda function

- Facilitates routing to a single Lambda function.
- Accessible to Application Load Balancers only.

Application Load Balancer

- Offers the flexibility for a Network Load Balancer to accept and route TCP requests within a specific VPC.
- Facilitates using static IP addresses and PrivateLink with an Application Load Balancer.

Target group name: **new-project-target-group-8080**

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol : Port

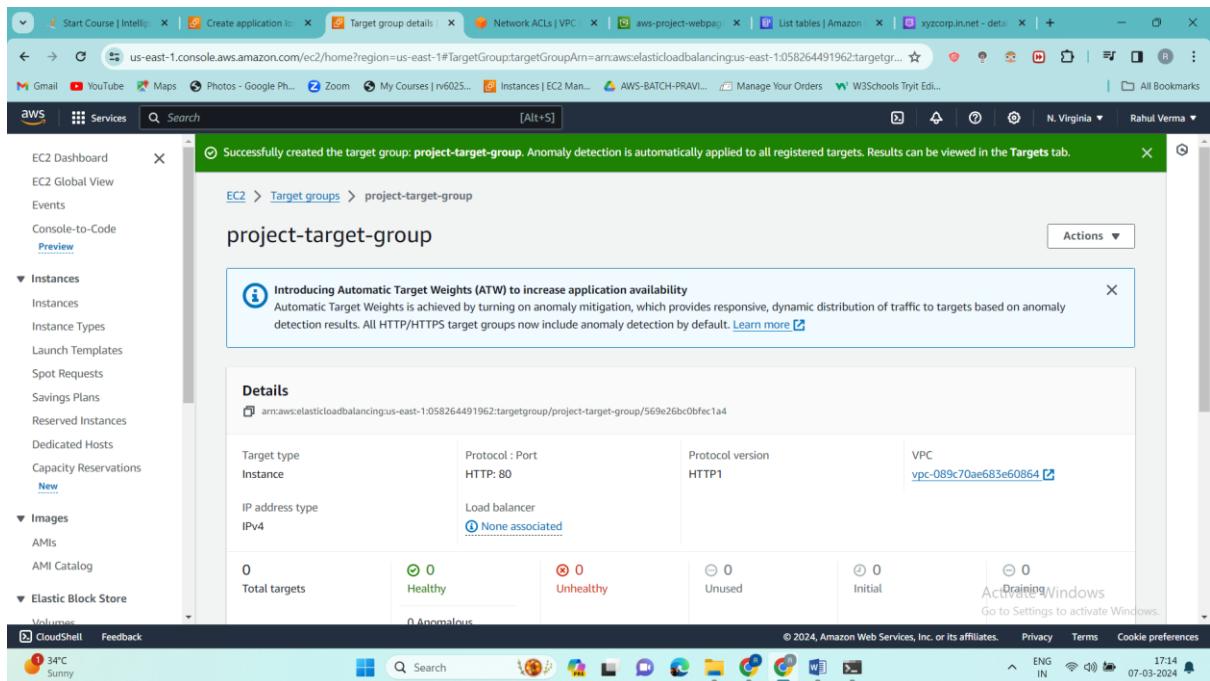
Choose a protocol for your target group that corresponds to the Load Balancer type that will route traffic to it. Some protocols now include anomaly detection for the targets and you can set mitigation options once your target group is created. This choice cannot be changed after creation.

HTTP 8080 1-65535

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 33°C Sunny ENG IN 14:28 08-03-2024

We are not selecting those 2 instances and just click it on create button



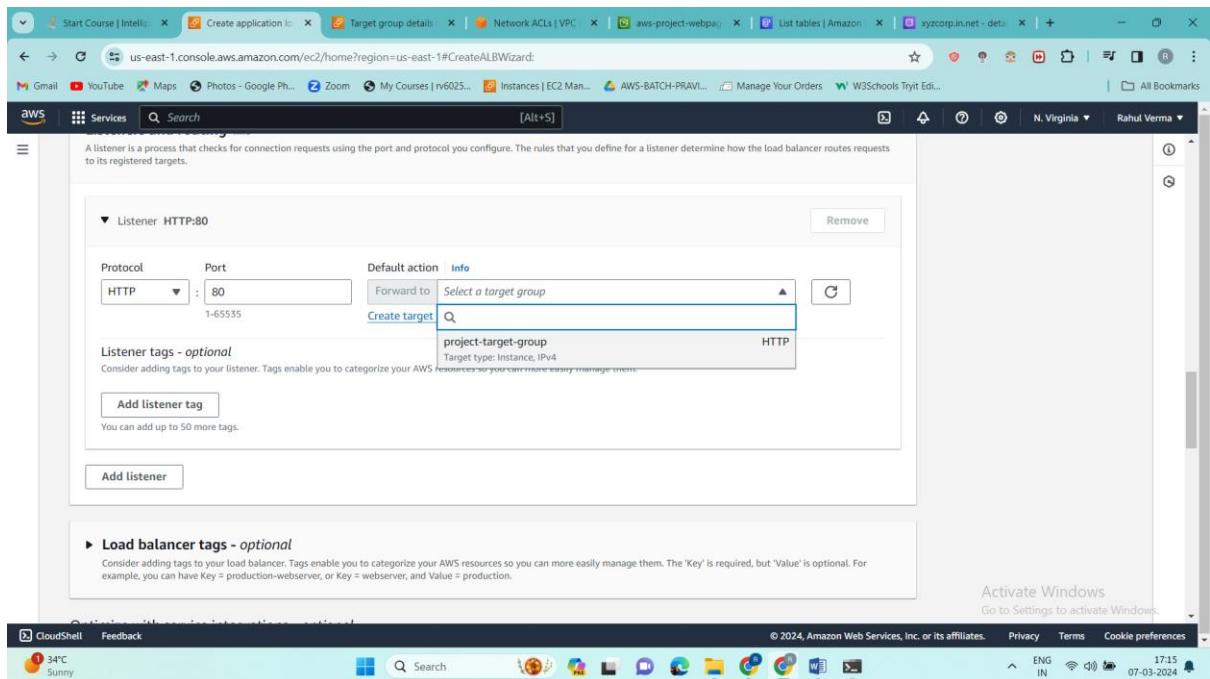
Successfully created the target group: **project-target-group**. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the **Targets** tab.

project-target-group

Introducing Automatic Target Weights (ATW) to increase application availability
Automatic Target Weights is achieved by turning on anomaly mitigation, which provides responsive, dynamic distribution of traffic to targets based on anomaly detection results. All HTTP/HTTPS target groups now include anomaly detection by default. [Learn more](#)

Details			
arn:aws:elasticloadbalancing:us-east-1:058264491962:targetgroup/project-target-group/569e26bc0bfec1a4			
Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC vpc-089c70ae683e60864
IP address type IPv4	Load balancer None associated		
0 Total targets	0 Healthy	0 Unhealthy	0 Unused
	0 Anomalous		0 Initial
			0 Draining

Now will select our target group



A listener is a process that checks for connection requests using the port and protocol you configure. The rules that you define for a listener determine how the load balancer routes requests to its registered targets.

Listener HTTP:80

Protocol	Port	Default action	Info
HTTP	: 80	Forward to	Select a target group

Listener tags - optional
Consider adding tags to your listener. Tags enable you to categorize your AWS resources so you can more easily manage them.

[Add listener tag](#)

Load balancer tags - optional
Consider adding tags to your load balancer. Tags enable you to categorize your AWS resources so you can more easily manage them. The 'Key' is required, but 'Value' is optional. For example, you can have Key = production-webserver, or Key = webserver, and Value = production.

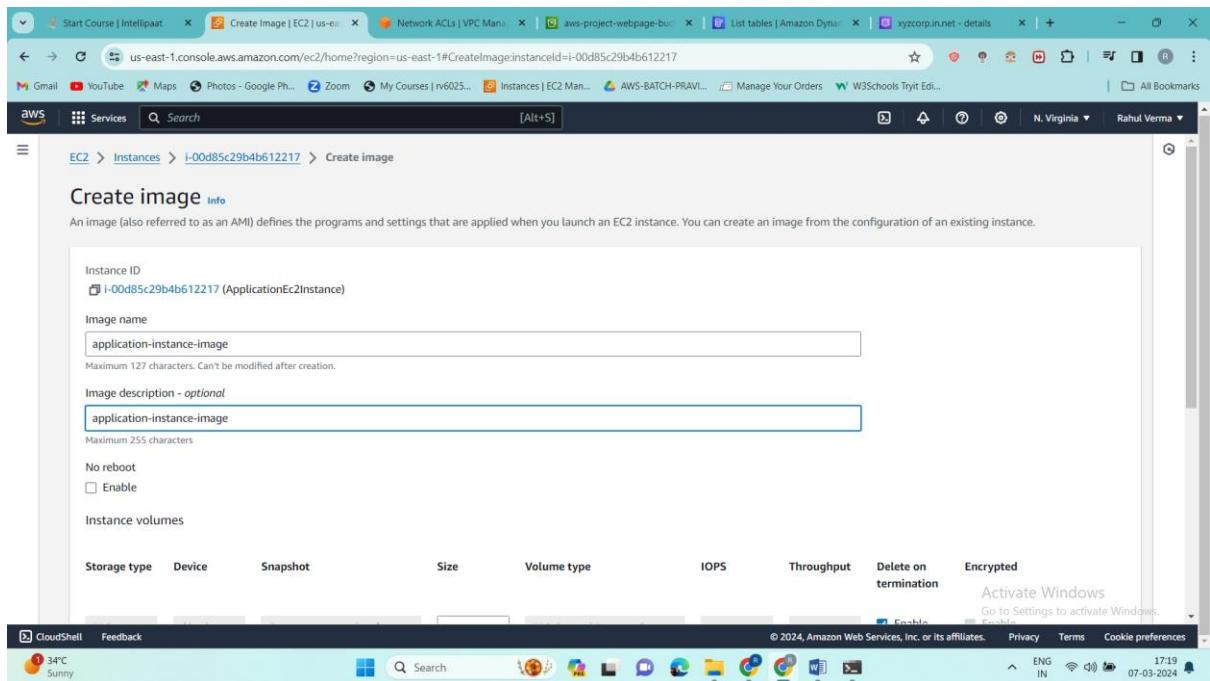
And our load balancer is created now

The screenshot shows the AWS EC2 Load Balancers page. A green success message box at the top states: "Successfully created load balancer: aws-foundation-alb. It might take a few minutes for your load balancer to fully set up and route traffic. Targets will also take a few minutes to complete the registration process and pass initial health checks." Below this, the "aws-foundation-alb" load balancer is listed with its details: Load balancer type: Application, Status: Provisioning, VPC: [vpc-089c70ae683e60864](#), IP address type: IPv4, Scheme: Internet-facing, Hosted zone: Z355XDDOTRQ7X7K, Availability Zones: subnet-05407b996d393844b (us-east-1b), subnet-084bbeac979bfc5b8 (us-east-1a), and Date created: March 7, 2024, 17:15 (UTC+05:30). A "Give feedback" button is also present. The left sidebar shows the navigation menu for EC2 services.

Now we have to create AMI from application instance

The screenshot shows the AWS EC2 Instances page. It displays two instances: "Bastion-Host" and "ApplicationEc2Instance". The "ApplicationEc2Instance" is selected. A context menu is open over this instance, with the "Launch instances" option highlighted. Other options in the menu include Connect, View details, Manage instance state, Instance settings, Networking, Security, Image and templates, and Monitor and troubleshoot. Below the instances, the "Instance summary" section provides details for the selected instance: Instance ID: i-00d85c29b4b612217, Public IPv4 address: -, Private IPv4 address: 10.20.2.62, Instance state: Running, Private IP DNS name: ip-10-20-2-62.ec2.internal, Instance type: t2.micro, and Elastic IP addresses: -. A "Create image" button is also visible in the context menu. The left sidebar shows the navigation menu for EC2 services.

And name it



EC2 > Instances > i-00d85c29b4b612217 > Create image

Create image Info

An image (also referred to as an AMI) defines the programs and settings that are applied when you launch an EC2 instance. You can create an image from the configuration of an existing instance.

Instance ID: i-00d85c29b4b612217 (ApplicationEc2Instance)

Image name: application-instance-image

Maximum 127 characters. Can't be modified after creation.

Image description - *optional*: application-instance-image

Maximum 255 characters.

No reboot: Enable

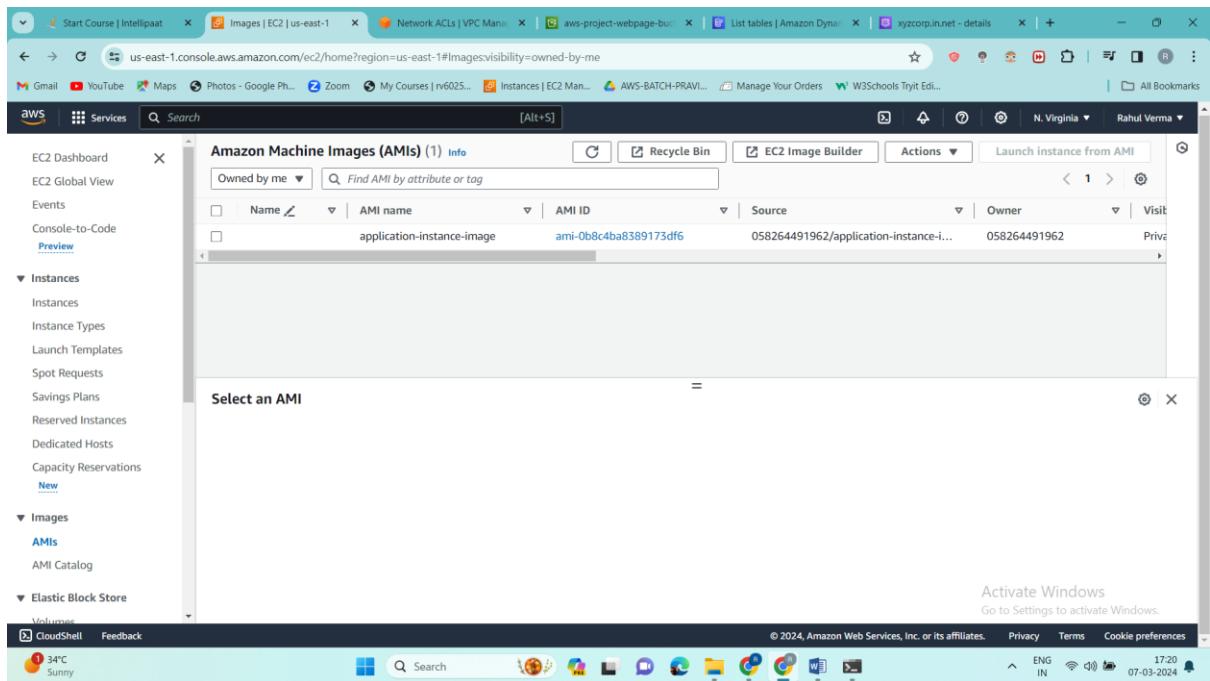
Instance volumes

Storage type	Device	Snapshot	Size	Volume type	IOPS	Throughput	Delete on termination	Encrypted

Activate Windows
Go to Settings to activate Windows

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 34°C Sunny 17:19 07-03-2024 ENG IN

Ami is created now



EC2 Dashboard

EC2 Global View

Events

Console-to-Code

Preview

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

New

Images

AMIs

AMI Catalog

Elastic Block Store

CloudShell Feedback

34°C Sunny

Amazon Machine Images (AMIs) (1) Info

Owned by me

Actions

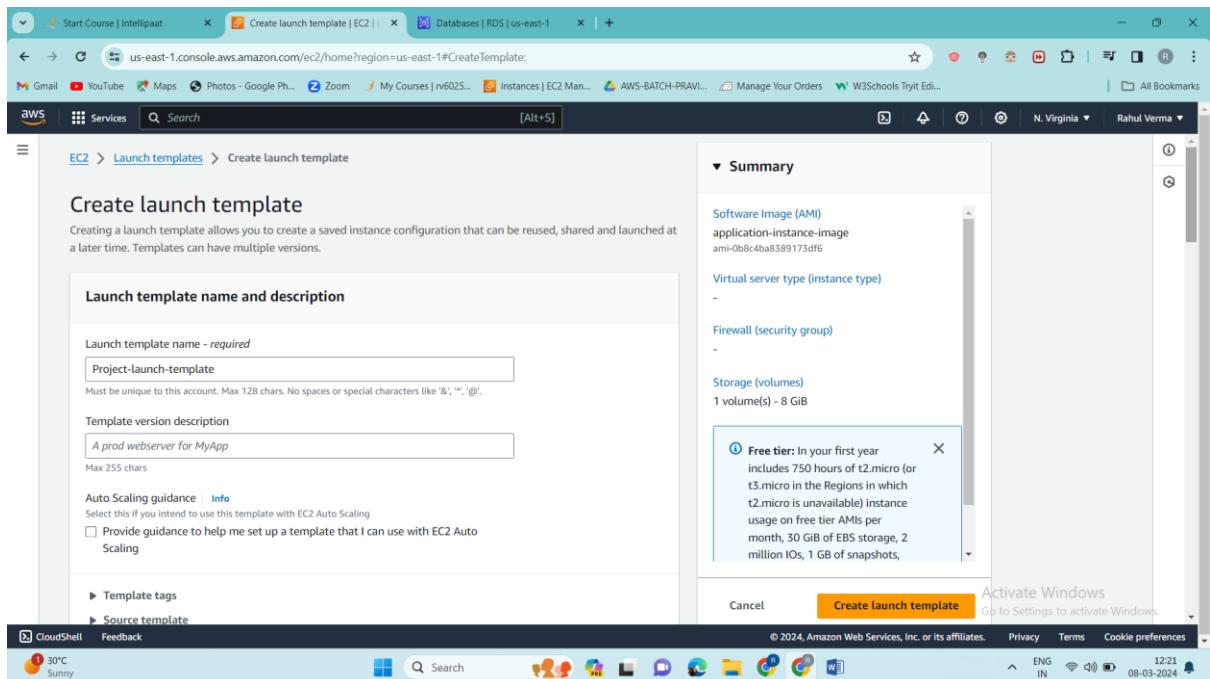
Name	AMI name	AMI ID	Source	Owner	Visit
	application-instance-image	ami-0b8c4ba8389173df6	058264491962/application-instance-i...	058264491962	Private

Select an AMI

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 34°C Sunny 17:20 07-03-2024 ENG IN

Now we have to create launch template



Start Course | Intellipaat >Create launch template | EC2 | Databases | RDS | us-east-1

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTemplate

Gmail YouTube Maps Photos - Google Ph... Zoom My Courses | rv6025... Instances | EC2 Man... AWS-BATCH-PRAVI... Manage Your Orders W3Schools Tryit Ed... All Bookmarks

aws Services Search [Alt+S]

EC2 > Launch templates > Create launch template

Create launch template

Creating a launch template allows you to create a saved instance configuration that can be reused, shared and launched at a later time. Templates can have multiple versions.

Launch template name and description

Launch template name - required
Project-launch-template
Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '<', '@'.

Template version description
A prod webserver for MyApp
Max 255 chars

Auto Scaling guidance [Info](#)
Select this if you intend to use this template with EC2 Auto Scaling
 Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

Template tags
Source template

CloudShell Feedback 30°C Sunny 12:21 08-03-2024 © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN Activate Windows Go to Settings to activate Windows.

Summary

Software Image (AMI)
application-instance-image
ami-0b8c4ba8389173df6

Virtual server type (instance type)

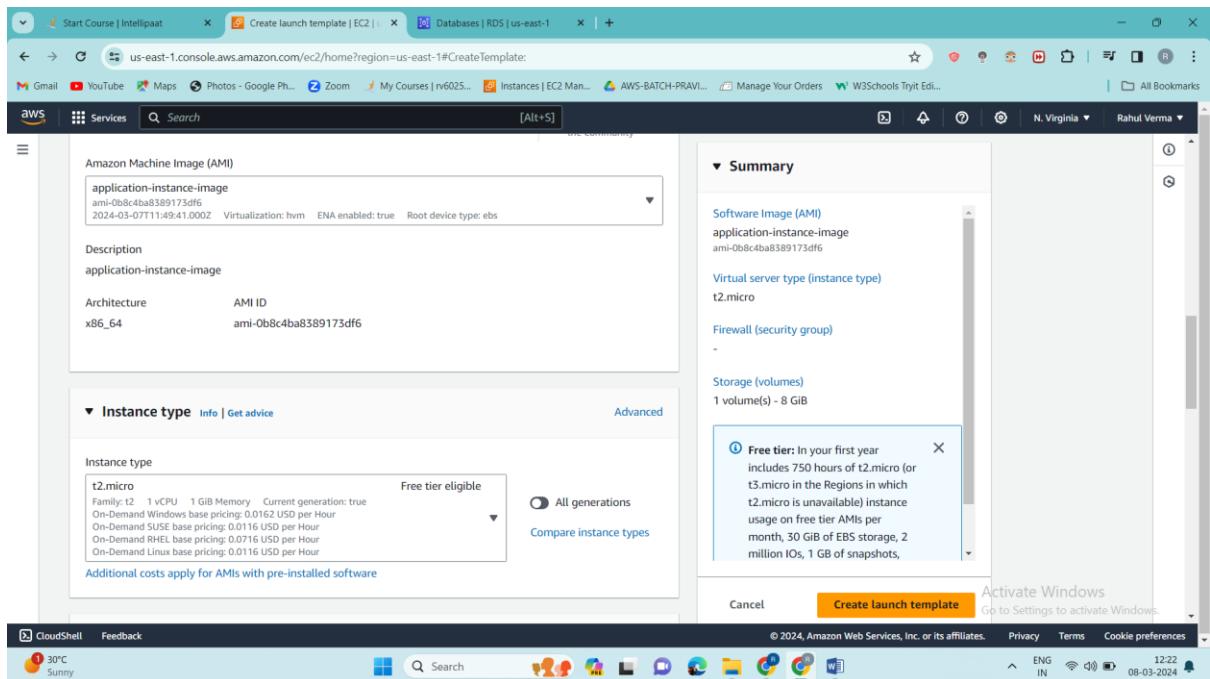
Firewall (security group)

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, 1000 API calls per month, and 1000 Lambda invocations per month.

Cancel Create launch template

Select your AMI



Start Course | Intellipaat Create launch template | EC2 | Databases | RDS | us-east-1

us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#CreateTemplate

Gmail YouTube Maps Photos - Google Ph... Zoom My Courses | rv6025... Instances | EC2 Man... AWS-BATCH-PRAVI... Manage Your Orders W3Schools Tryit Ed... All Bookmarks

aws Services Search [Alt+S]

EC2 > Launch templates > Create launch template

Amazon Machine Image (AMI)

application-instance-image
ami-0b8c4ba8389173df6 2024-03-07T11:49:41.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description
application-instance-image

Architecture AMI ID
x86_64 ami-0b8c4ba8389173df6

Instance type [Info](#) | [Get advice](#) Advanced

Instance type
t2.micro Family: t2 1 vCPU 1 GiB Memory Current generation: true Free tier eligible
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.0116 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

All generations Compare instance types

Additional costs apply for AMIs with pre-installed software

CloudShell Feedback 30°C Sunny 12:22 08-03-2024 © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences ENG IN Activate Windows Go to Settings to activate Windows.

Summary

Software Image (AMI)
application-instance-image
ami-0b8c4ba8389173df6

Virtual server type (instance type)
t2.micro

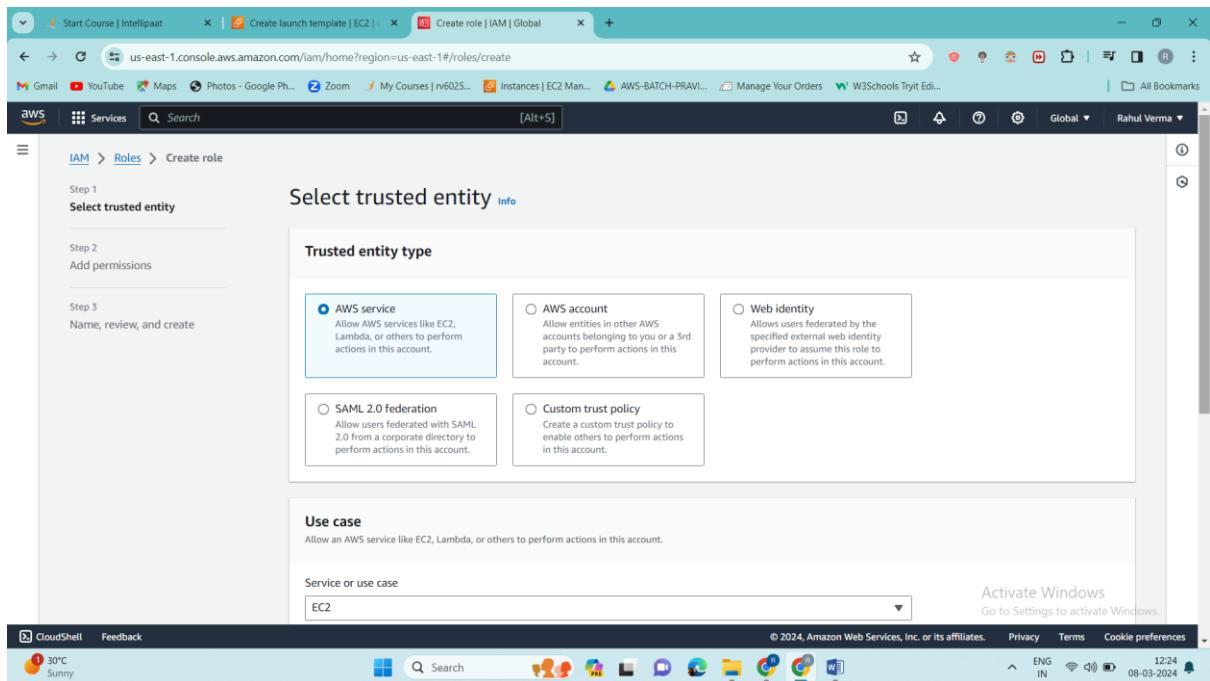
Firewall (security group)

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, 1000 API calls per month, and 1000 Lambda invocations per month.

Cancel Create launch template

We have to create IAM role also



Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity

Trusted entity type

- AWS service
- AWS account
- Web identity
- SAML 2.0 federation
- Custom trust policy

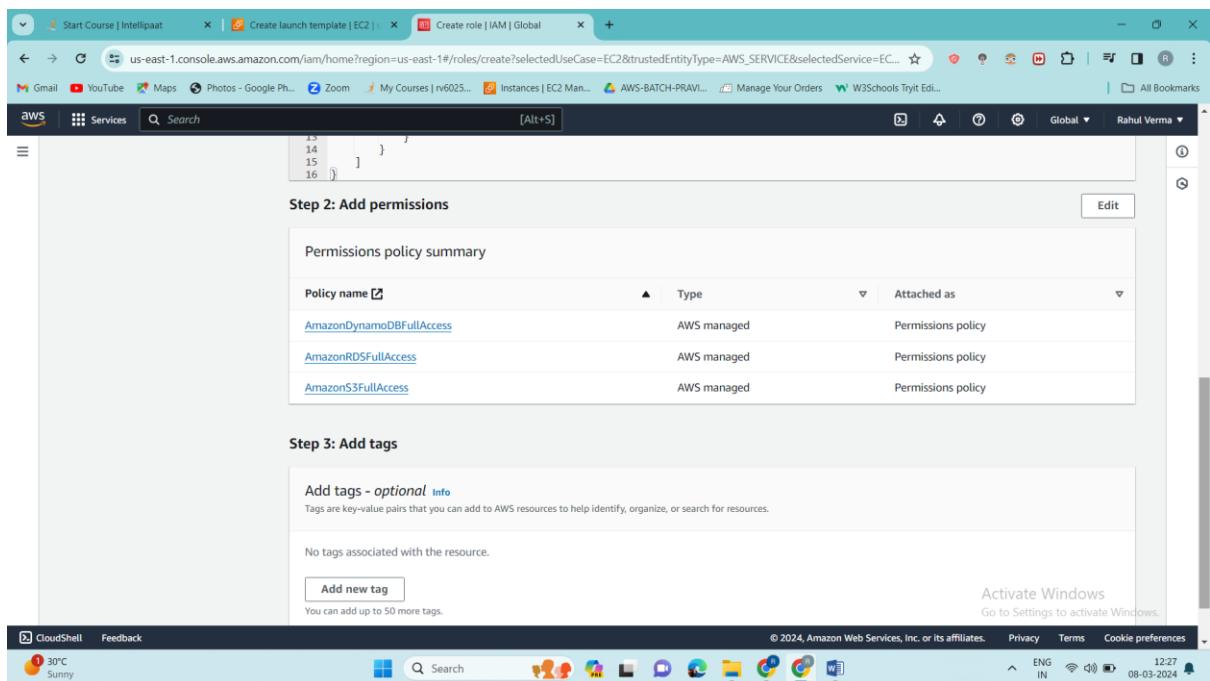
Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Service or use case

EC2

We have to add 3 permissions s3 full access, RDS full access & DynamoDb full access



Step 2: Add permissions

Policy name	Type	Attached as
AmazonDynamoDBFullAccess	AWS managed	Permissions policy
AmazonRDSFullAccess	AWS managed	Permissions policy
AmazonS3FullAccess	AWS managed	Permissions policy

Step 3: Add tags

Add tags - optional

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

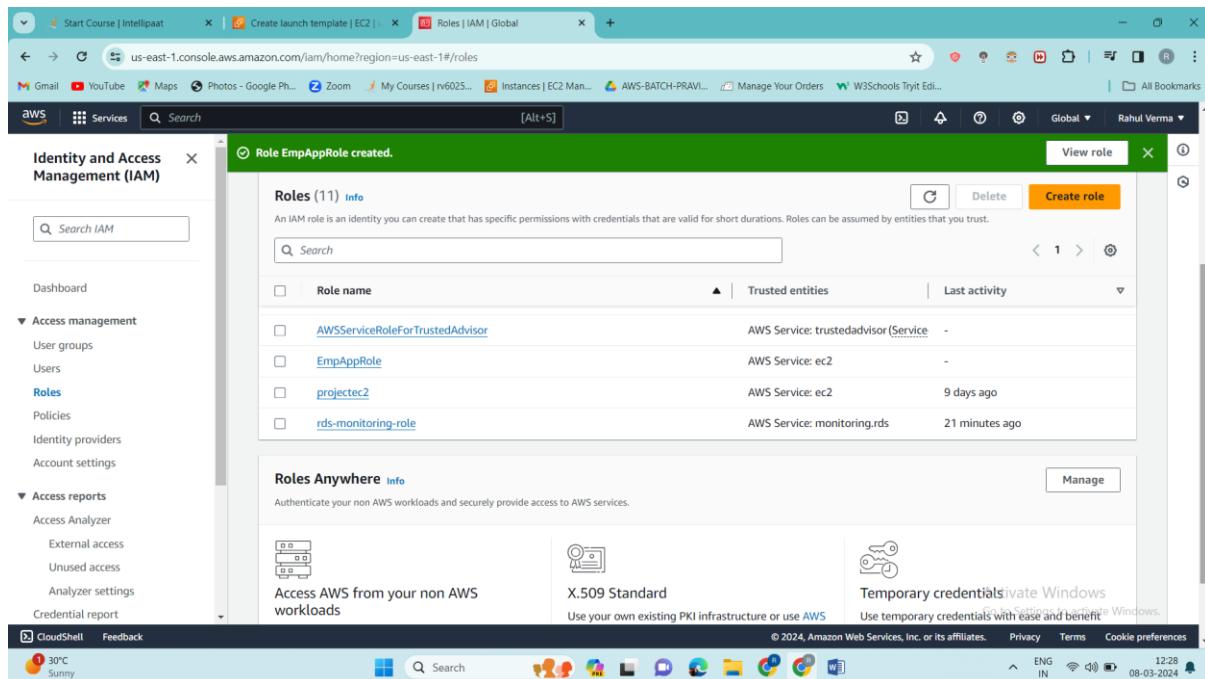
Add new tag

You can add up to 50 more tags.

Activate Windows

Go to Settings to activate Windows.

It's created

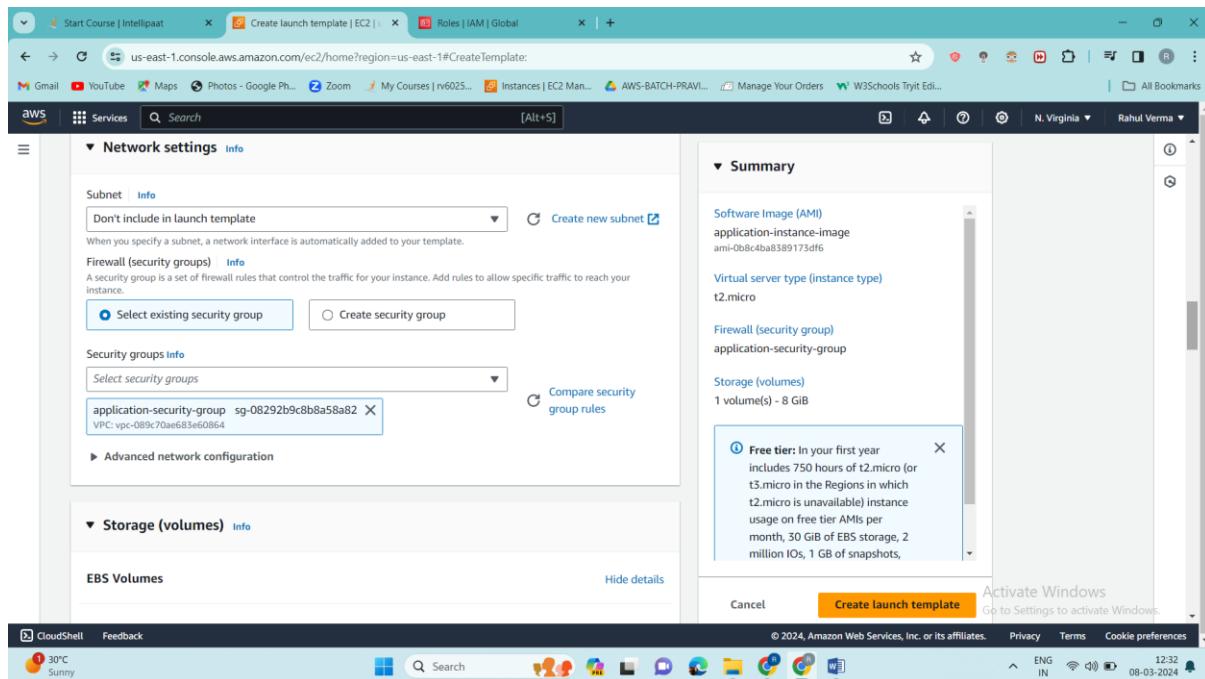


The screenshot shows the AWS IAM Roles page. The left sidebar is the 'Identity and Access Management (IAM)' menu. The main content area shows a table titled 'Roles (11) info' with the following data:

Role name	Trusted entities	Last activity
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
EmpAppRole	AWS Service: ec2	-
projectec2	AWS Service: ec2	9 days ago
rds-monitoring-role	AWS Service: monitoring.rds	21 minutes ago

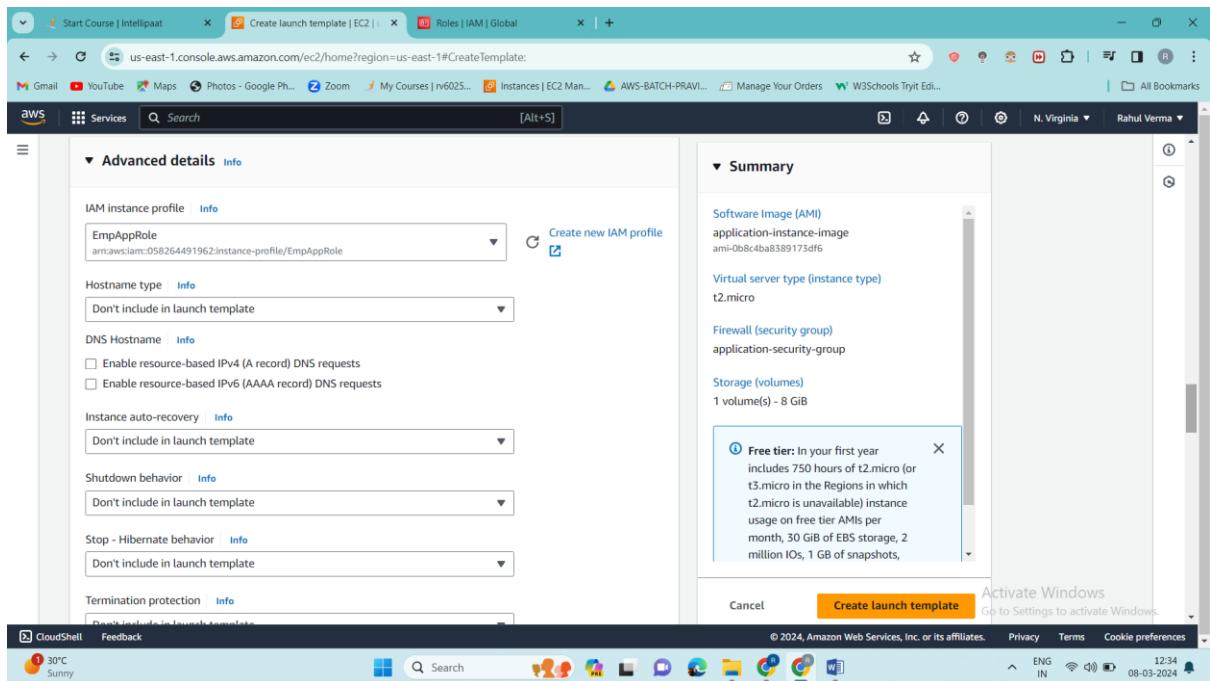
Below the table, there are three options: 'Access AWS from your non AWS workloads', 'X.509 Standard', and 'Temporary credentials (private Windows)'. The 'Temporary credentials' option is highlighted with a tooltip: 'Use temporary credentials with ease and benefit. Go to Settings to activate Windows.'

Now come back to launch template page and select application security group



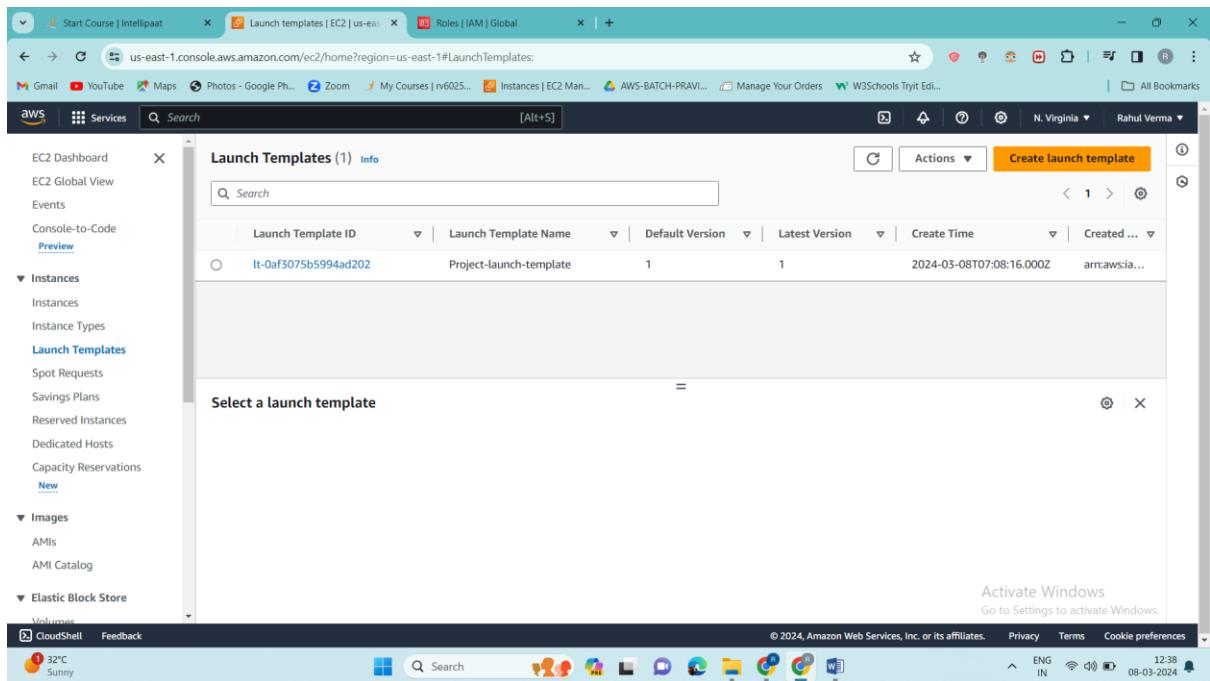
The screenshot shows the 'Create launch template' page. The left sidebar has 'Network settings' and 'Storage (volumes)' sections. The 'Network settings' section includes fields for 'Subnet' (set to 'Don't include in launch template'), 'Firewall (security groups)' (set to 'Select existing security group'), and 'Security groups' (listing 'application-security-group sg-08292b9c8b8a58a82' with a VPC ID). The 'Storage (volumes)' section includes an 'EBS Volumes' table with one row. A tooltip for the 'Free tier' is shown in the bottom right: 'Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOPS, 1 GB of snapshots.'

And in advance details select your IAM role



The screenshot shows the 'Create launch template' wizard in the AWS CloudFormation console. The current step is 'Advanced details'. A modal window titled 'Free tier' is open, providing information about the free tier benefits for t2.micro instances. The 'Create launch template' button is highlighted in orange at the bottom right of the modal.

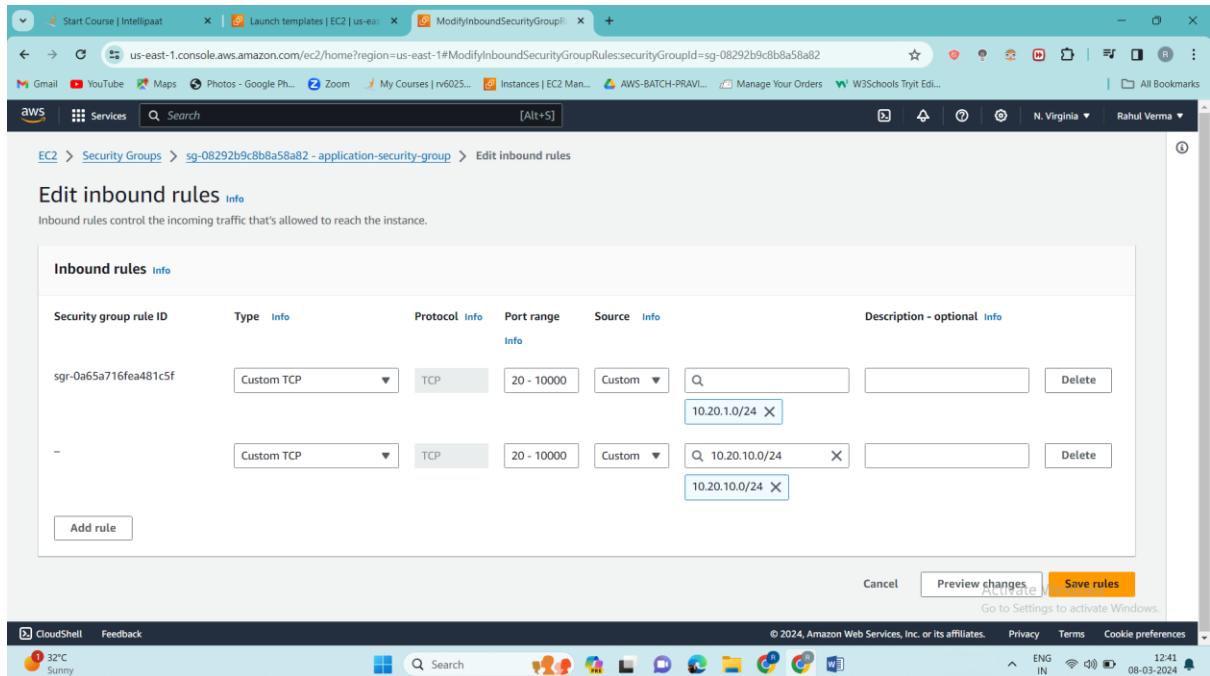
Rest default settings and create your launch template



The screenshot shows the 'Launch Templates' list page in the AWS CloudFormation console. It displays a single launch template named 'Project-launch-template' with a status of 'Active'. The 'Actions' dropdown menu for this template is open, showing options like 'Edit', 'Delete', and 'Clone'. The left sidebar shows the navigation menu for CloudFormation, including 'Launch Templates' which is currently selected.

We have to make some changes in application security group

We have to allow our second public subnet also

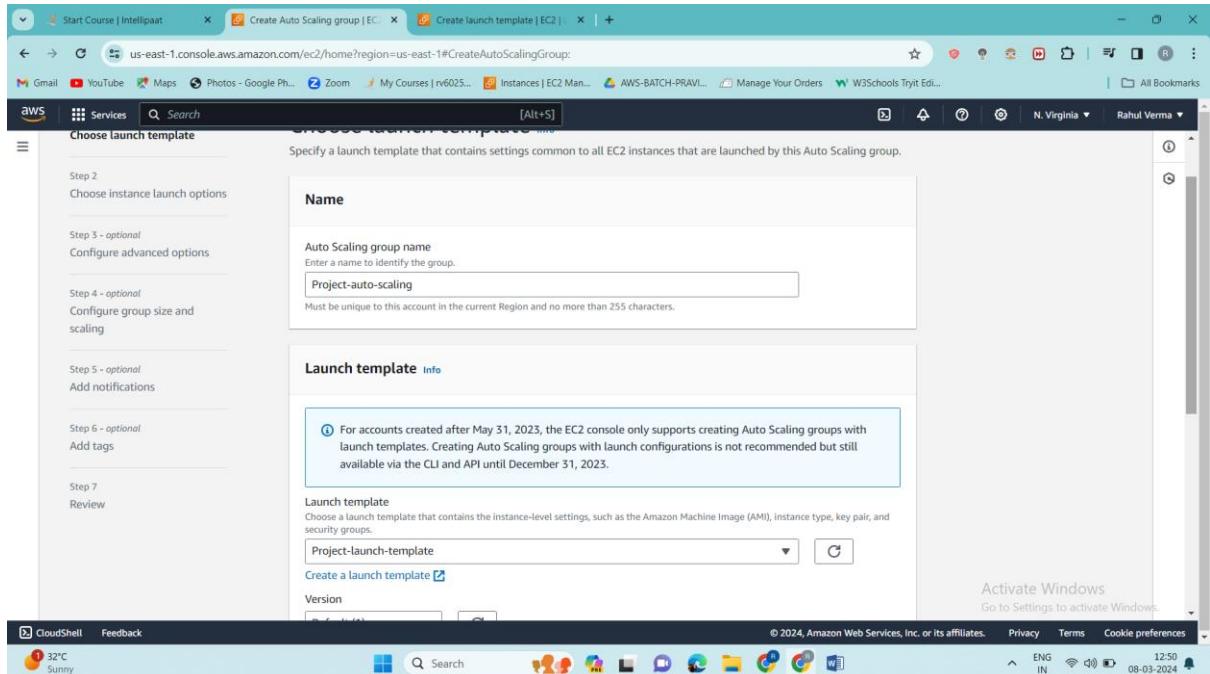


The screenshot shows the 'Edit inbound rules' page for a security group. There are two rules listed:

- Rule 1: Type: Custom TCP, Protocol: TCP, Port range: 20 - 10000, Source: Custom, Description: 10.20.1.0/24
- Rule 2: Type: Custom TCP, Protocol: TCP, Port range: 20 - 10000, Source: Custom, Description: 10.20.10.0/24

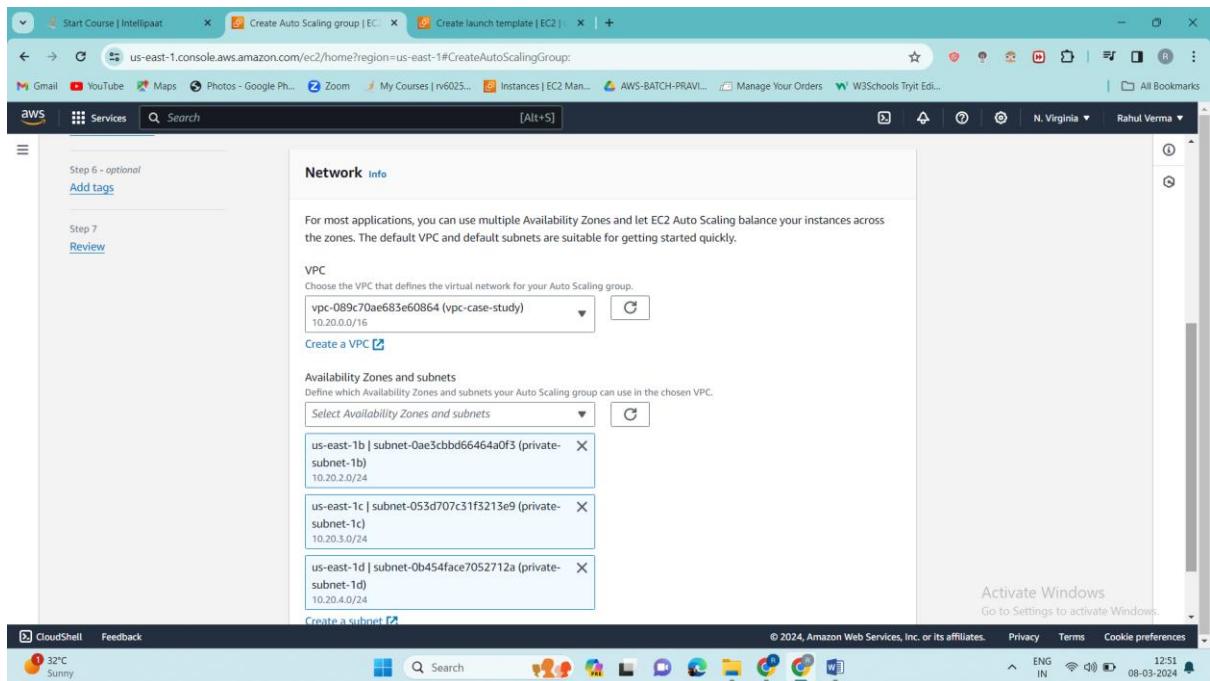
At the bottom right, there are 'Cancel', 'Preview changes', and 'Save rules' buttons. The 'Save rules' button is highlighted in orange.

Now we can create our auto scaling group



The screenshot shows the 'Create Auto Scaling group' wizard at Step 2: 'Choose instance launch options'. The left sidebar shows steps 2 through 7. The main area is titled 'Launch template' and shows a note: 'For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.' The 'Launch template' dropdown is set to 'Project-launch-template'. A 'Save' button is at the bottom right.

Select your VPC and select all private subnets



Step 6 - optional
Add tags

Step 7
Review

Network info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-089c70ae683e60864 (vpc-case-study)

Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets

us-east-1b | subnet-0ae3cbdb664a0f3 (private-subnet-1b)
10.20.2.0/24

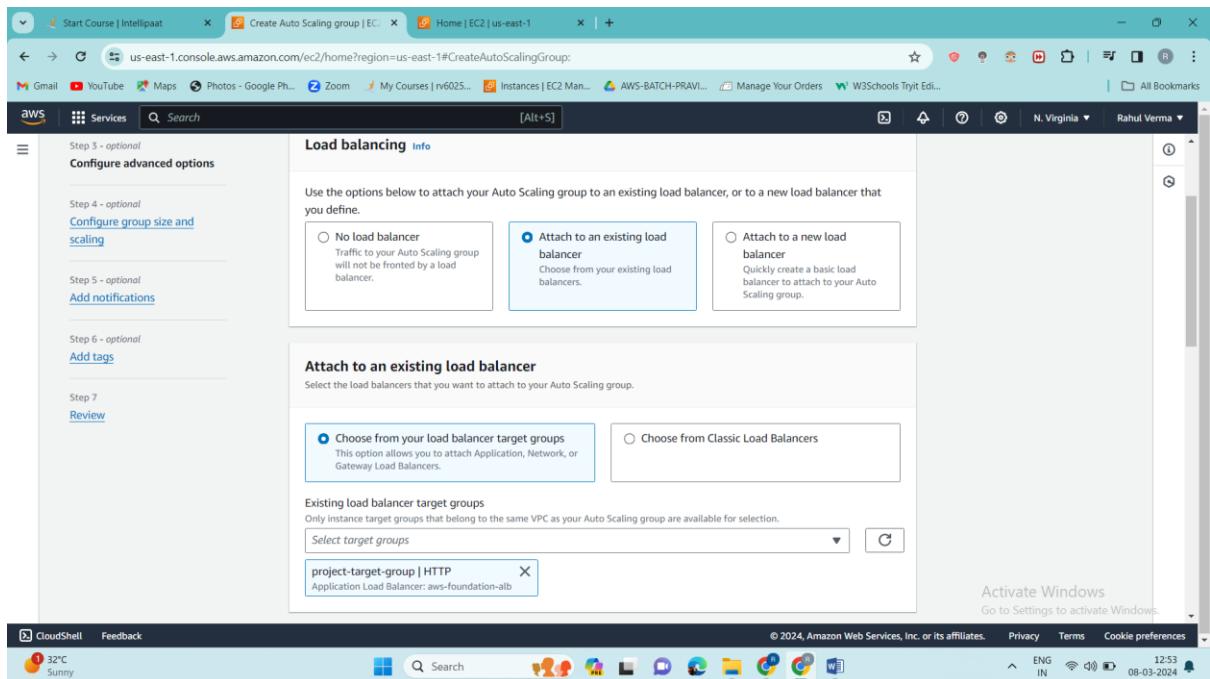
us-east-1c | subnet-053d707c31f3213e9 (private-subnet-1c)
10.20.3.0/24

us-east-1d | subnet-0b454face7052712a (private-subnet-1d)
10.20.4.0/24

Activate Windows
Go to Settings to activate Windows.

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 32°C Sunny 12:51 08-03-2024

Attach your Loadbalancer



Step 3 - optional
Configure advanced options

Step 4 - optional
Configure group size and scaling

Step 5 - optional
Add notifications

Step 6 - optional
Add tags

Step 7
Review

Load balancing info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.

No load balancer
Traffic to your Auto Scaling group will not be fronted by a load balancer.

Attach to an existing load balancer
Choose from your existing load balancers.

Attach to a new load balancer
Quickly create a basic load balancer to attach to your Auto Scaling group.

Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.

Choose from your load balancer target groups
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

Existing load balancer target groups
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

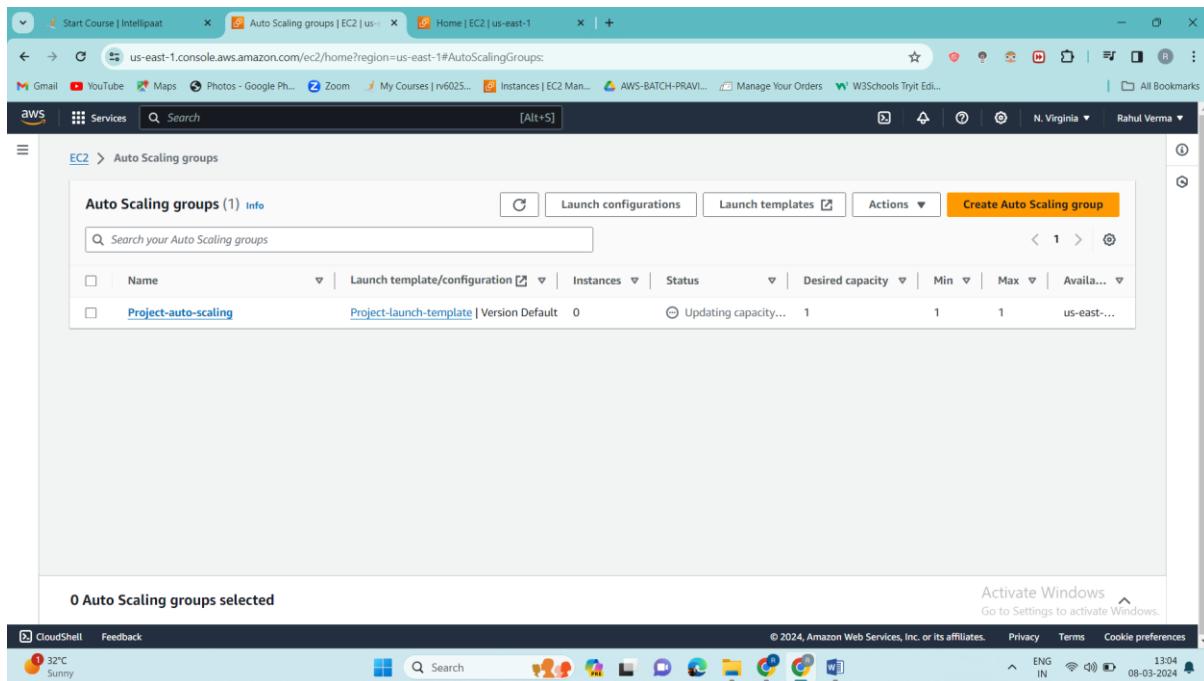
Select target groups

project-target-group | HTTP
Application Load Balancer: aws-foundation-alb

Activate Windows
Go to Settings to activate Windows.

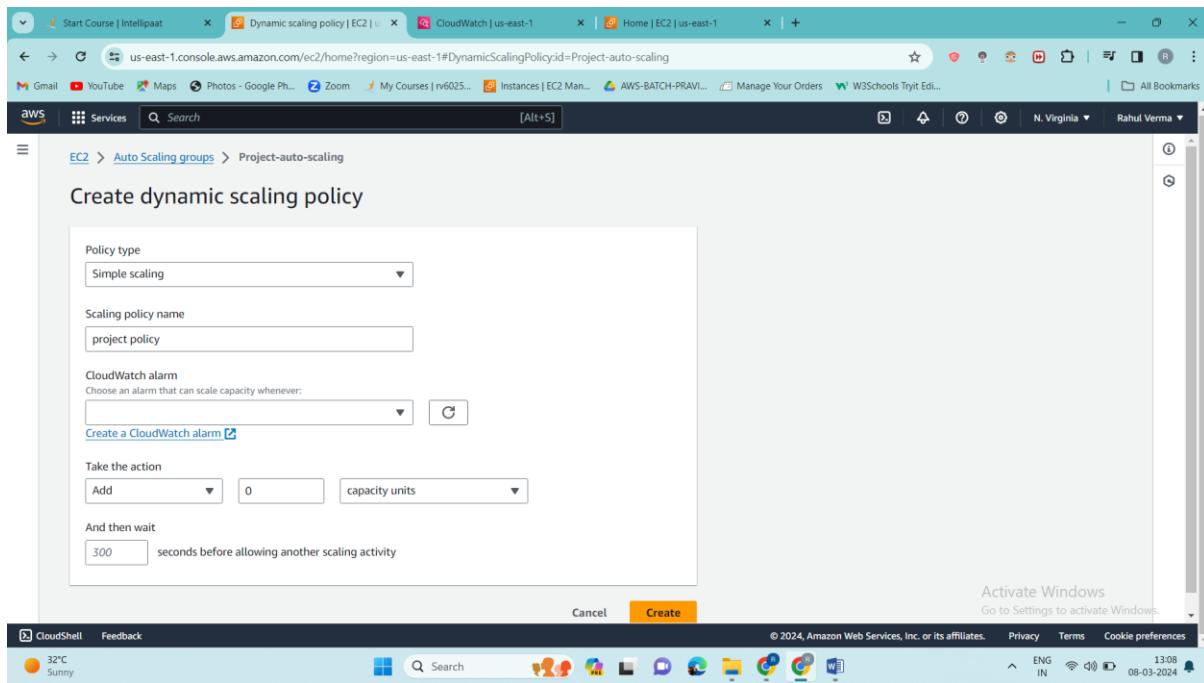
CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences 32°C Sunny 12:53 08-03-2024

Now just create your auto scaling group



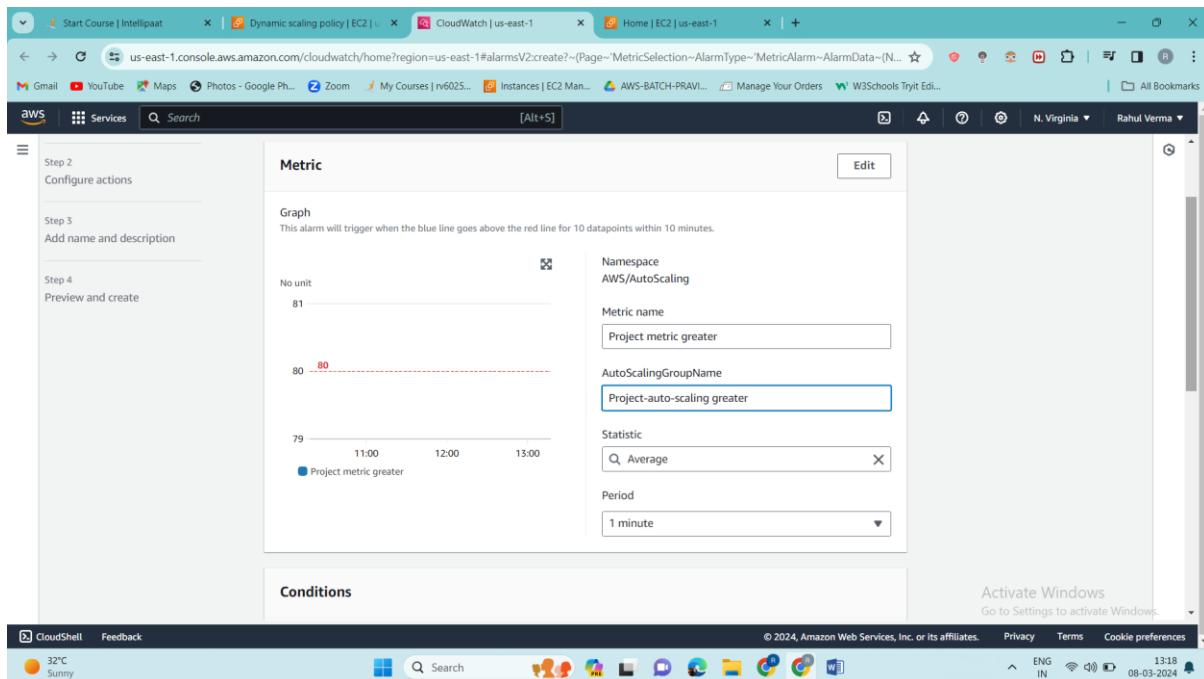
The screenshot shows the AWS CloudWatch Metrics Metrics Insights page. The top navigation bar includes links for Start Course, Auto Scaling groups, Home, and EC2. The main content area displays a table of metrics with columns for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Available. A single row is selected, labeled 'Project-auto-scaling' with a status of '0'. The bottom of the page shows a summary: '0 Auto Scaling groups selected' and a 'Create Auto Scaling group' button.

Let's create simple dynamic policy for our Autoscaling



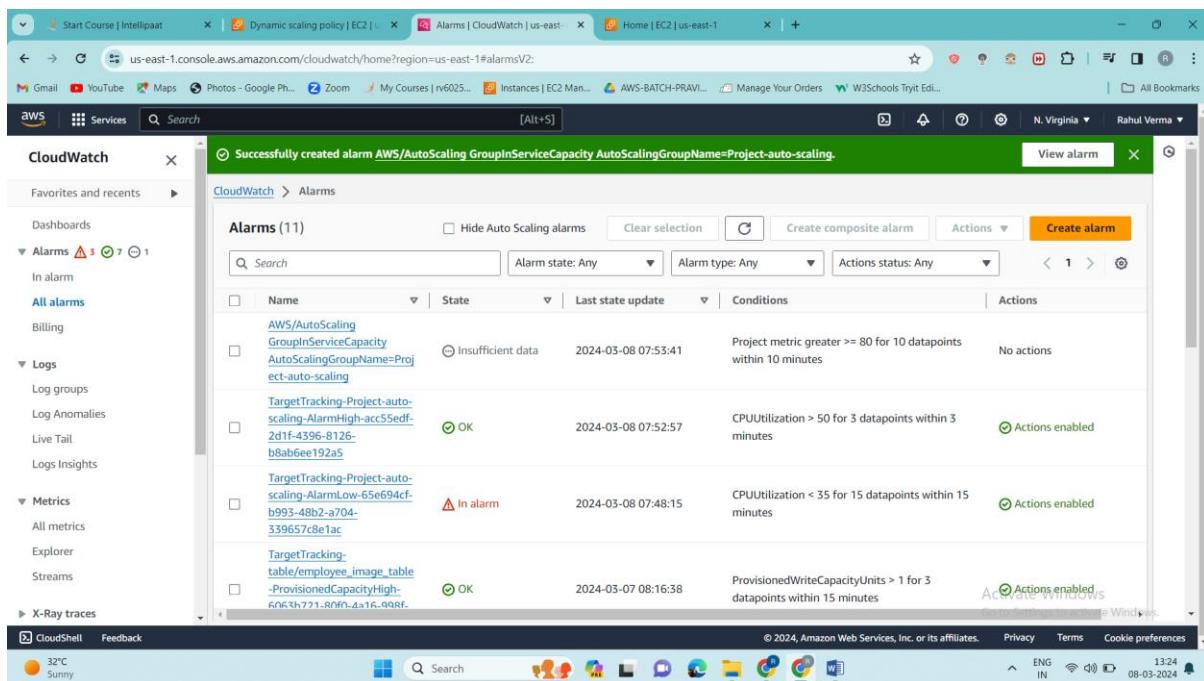
The screenshot shows the AWS CloudWatch Metrics Metrics Insights page. The top navigation bar includes links for Start Course, Dynamic scaling policy, CloudWatch, Home, and EC2. The main content area displays a table of metrics with columns for Name, Launch template/configuration, Instances, Status, Desired capacity, Min, Max, and Available. A single row is selected, labeled 'Project-auto-scaling' with a status of '0'. The bottom of the page shows a summary: '0 Auto Scaling groups selected' and a 'Create Auto Scaling group' button.

And we need one cloud watch for this



The screenshot shows the AWS CloudWatch Metrics console. A new alarm is being created. The 'Metric' tab is active, showing a graph of a metric over time (11:00 to 13:00) with a threshold line at 80. The 'Conditions' tab is also visible. The alarm is named 'Project metric greater' and is set to trigger when the metric crosses the threshold.

So whenever it goes above => 80 % it will trigger the alarm

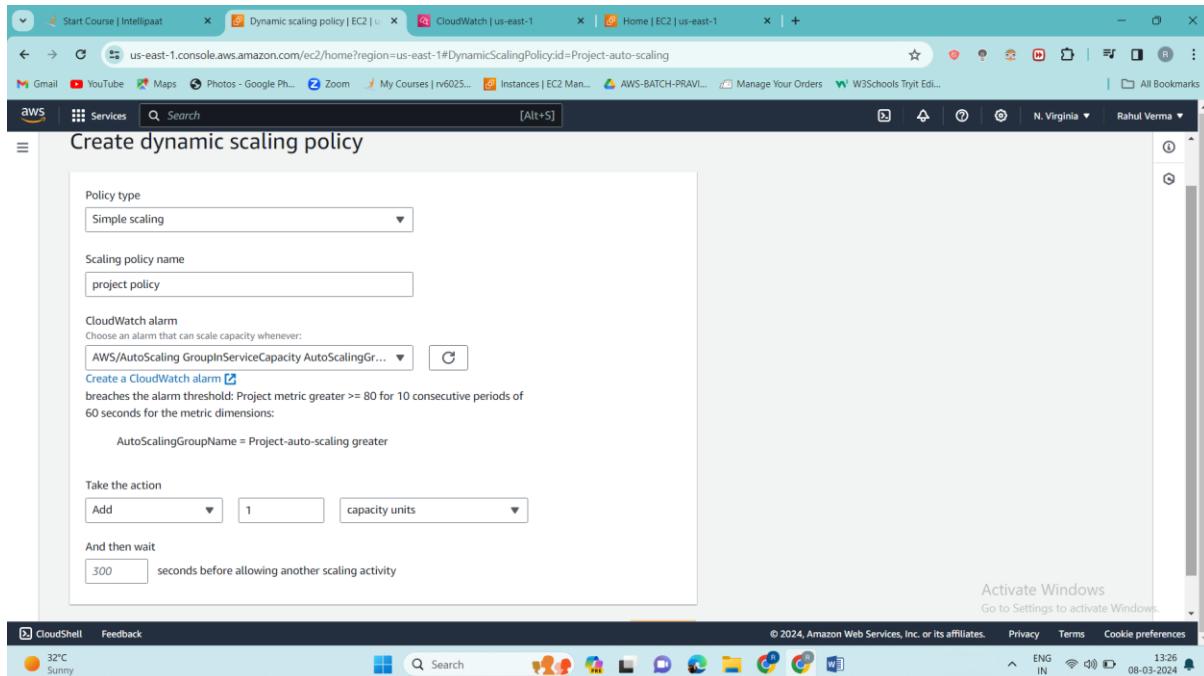


The screenshot shows the AWS CloudWatch Alarms console. A new alarm has been successfully created, named 'AWS/AutoScaling GroupInServiceCapacity AutoScalingGroupName=Project-auto-scaling'. The alarm is currently in an 'Insufficient data' state. The table lists the alarm details, including its name, state, last state update, conditions, and actions.

Name	State	Last state update	Conditions	Actions
AWS/AutoScaling GroupInServiceCapacity AutoScalingGroupName=Project-auto-scaling	Insufficient data	2024-03-08 07:53:41	Project metric greater >= 80 for 10 datapoints within 10 minutes	No actions
TargetTracking-Project-autoscaling-AlarmHigh-ac55edf-2d1f-4396-8126-b8ab6ee192a5	OK	2024-03-08 07:52:57	CPUUtilization > 50 for 3 datapoints within 3 minutes	Actions enabled
TargetTracking-Project-autoscaling-AlarmLow-65e694cf-b693-48b2-a704-339657c8e1ac	In alarm	2024-03-08 07:48:15	CPUUtilization < 35 for 15 datapoints within 15 minutes	Actions enabled
TargetTracking-table/employee_image_table-ProvisionedCapacityHigh-606f4771-R0fN-4a16-99RF-	OK	2024-03-07 08:16:38	ProvisionedWriteCapacityUnits > 1 for 3 datapoints within 15 minutes	Actions enabled

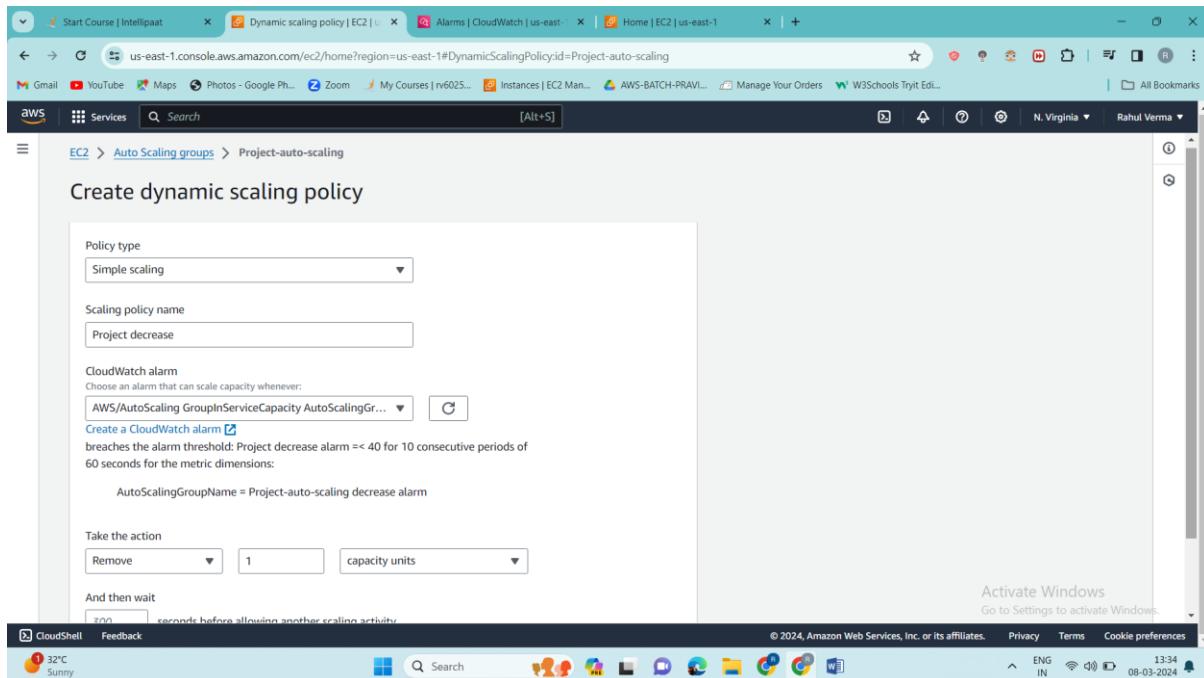
And now will select this cloud watch to our scaling policy

Whenever it's CPU utilization is $\geq 80\%$ it add one instance



The screenshot shows the AWS CloudWatch Metrics console with the URL <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#DynamicScalingPolicy:id=Project-auto-scaling>. The page is titled "Create dynamic scaling policy". The "Policy type" is set to "Simple scaling". The "Scaling policy name" is "project policy". Under "CloudWatch alarm", a dropdown menu shows "AWS/AutoScaling GroupInServiceCapacity AutoScalingGr...". A link "Create a CloudWatch alarm" is available. Below it, a note states: "breaches the alarm threshold: Project metric greater ≥ 80 for 10 consecutive periods of 60 seconds for the metric dimensions: AutoScalingGroupName = Project-auto-scaling greater". The "Take the action" section shows "Add" selected, "1" instances, and "capacity units". The "And then wait" section shows "300" seconds. The bottom right corner of the browser window shows "Activate Windows" and "Go to Settings to activate Windows".

Similarly will add one more, where condition will be whenever cpu utilization goes ≤ 40 it will decrease instance

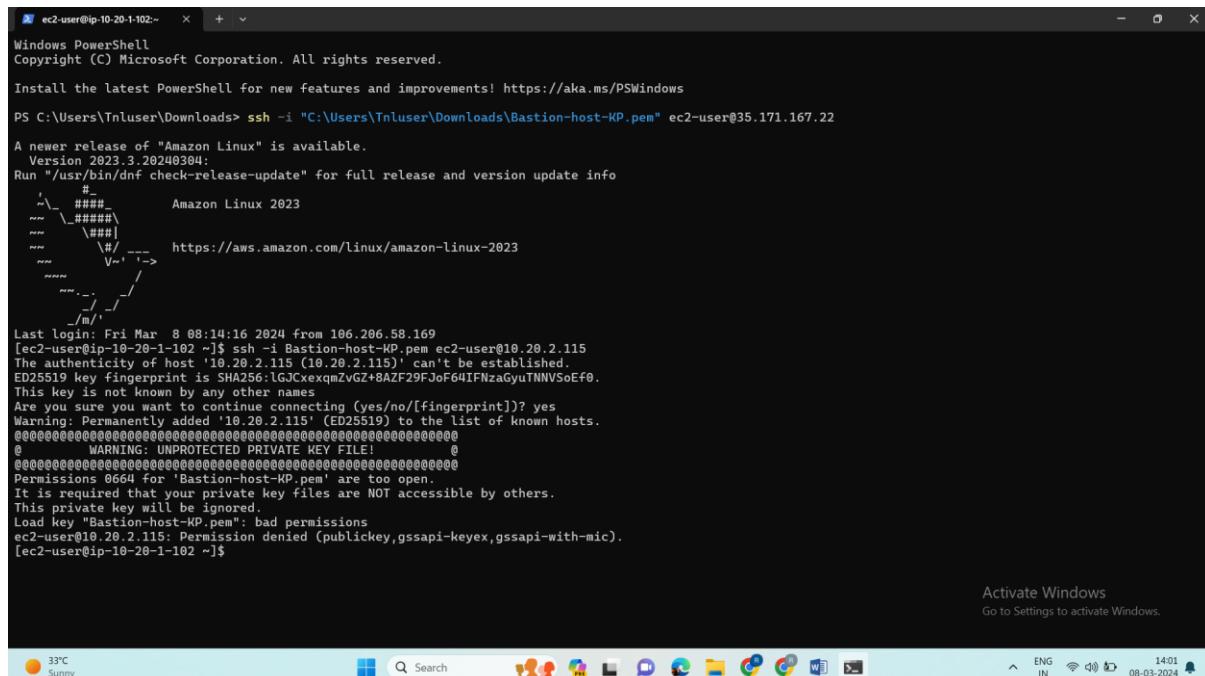


The screenshot shows the AWS CloudWatch Metrics console with the URL <https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#DynamicScalingPolicy:id=Project-auto-scaling>. The page is titled "Create dynamic scaling policy". The "Policy type" is set to "Simple scaling". The "Scaling policy name" is "Project decrease". Under "CloudWatch alarm", a dropdown menu shows "AWS/AutoScaling GroupInServiceCapacity AutoScalingGr...". A link "Create a CloudWatch alarm" is available. Below it, a note states: "breaches the alarm threshold: Project decrease alarm ≤ 40 for 10 consecutive periods of 60 seconds for the metric dimensions: AutoScalingGroupName = Project-auto-scaling decrease alarm". The "Take the action" section shows "Remove" selected, "1" instances, and "capacity units". The "And then wait" section shows "300" seconds. The bottom right corner of the browser window shows "Activate Windows" and "Go to Settings to activate Windows".

So our autoscaling created one instance now

Now have logged in to our bastion host and from this will login to our autoscaling instance

And we are connected to our private instance



```
ec2-user@ip-10-20-1-102:~ x + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

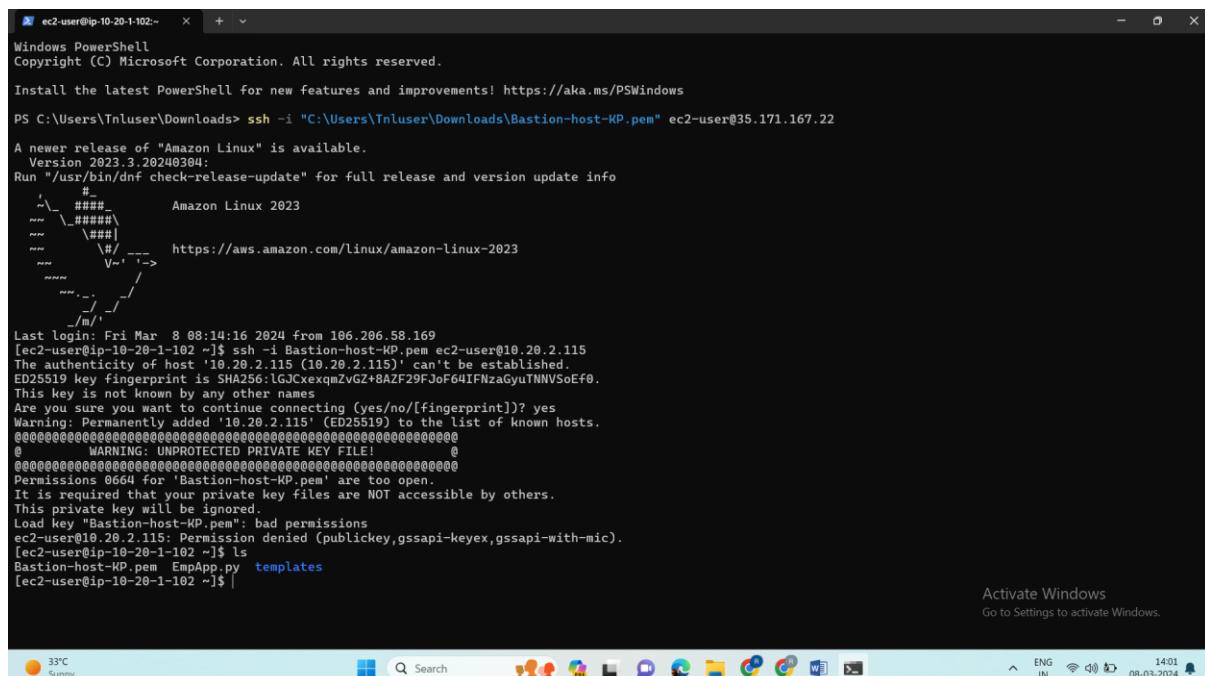
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Tnluser\Downloads> ssh -i "C:\Users\Tnluser\Downloads\Bastion-host-KP.pem" ec2-user@35.171.167.22
A newer release of "Amazon Linux" is available.
Version 2023.3.20240304:
Run "/usr/bin/dnf check-release-update" for full release and version update info
#
`\_ #####_ Amazon Linux 2023
~~ \#####\ \
~~ \###\ \
~~ \#/ ---> https://aws.amazon.com/linux/amazon-linux-2023
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
Last login: Fri Mar  8 08:14:16 2024 from 106.206.58.169
[ec2-user@ip-10-20-1-102 ~]$ ssh -i Bastion-host-KP.pem ec2-user@10.20.2.115
The authenticity of host '10.20.2.115' (10.20.2.115) can't be established.
ED25519 key fingerprint is SHA256:LGJxexqmZvGZ+8AZF29FJoF64IFNzaGyuTNNVSoEf0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.20.2.115' (ED25519) to the list of known hosts.
@         WARNING: UNPROTECTED PRIVATE KEY FILE! @
Permissions 0664 for 'Bastion-host-KP.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "Bastion-host-KP.pem": bad permissions
ec2-user@10.20.2.115: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-20-1-102 ~]$
```

Activate Windows
Go to Settings to activate Windows.

33°C Sunny 14:01 08-03-2024

And let's check everything is there or not



```
ec2-user@ip-10-20-1-102:~ x + v
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

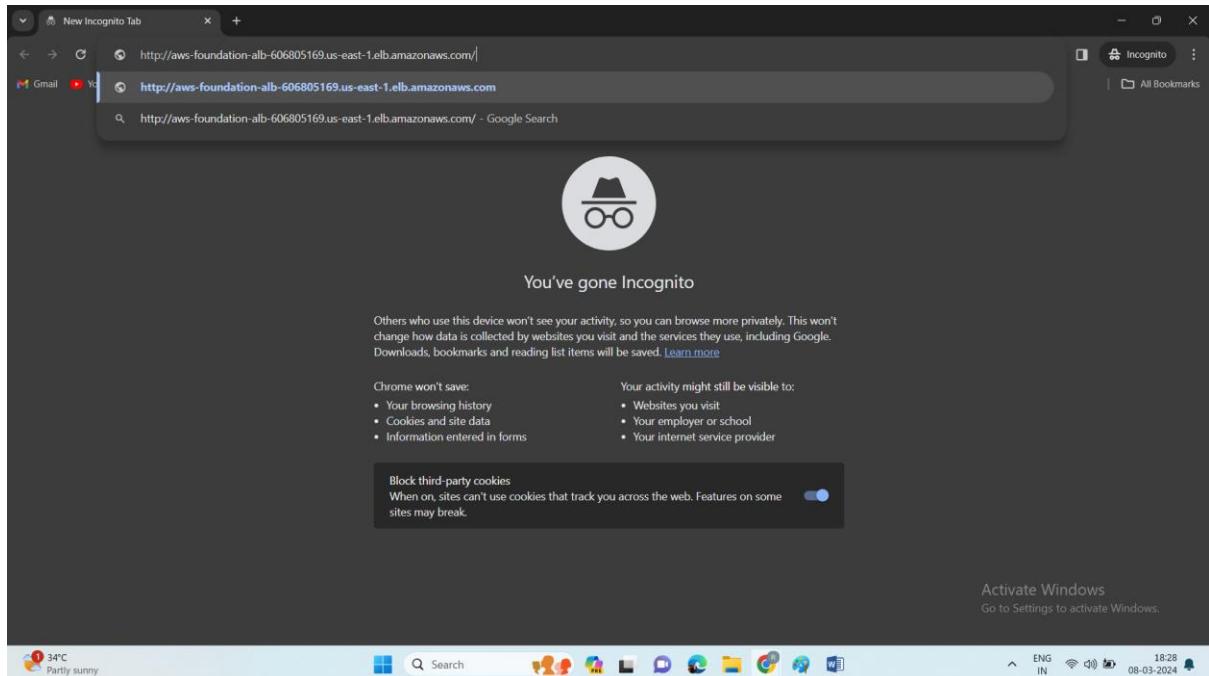
PS C:\Users\Tnluser\Downloads> ssh -i "C:\Users\Tnluser\Downloads\Bastion-host-KP.pem" ec2-user@35.171.167.22
A newer release of "Amazon Linux" is available.
Version 2023.3.20240304:
Run "/usr/bin/dnf check-release-update" for full release and version update info
#
`\_ #####_ Amazon Linux 2023
~~ \#####\ \
~~ \###\ \
~~ \#/ ---> https://aws.amazon.com/linux/amazon-linux-2023
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
~~ \#/ --->
Last login: Fri Mar  8 08:14:16 2024 from 106.206.58.169
[ec2-user@ip-10-20-1-102 ~]$ ssh -i Bastion-host-KP.pem ec2-user@10.20.2.115
The authenticity of host '10.20.2.115' (10.20.2.115) can't be established.
ED25519 key fingerprint is SHA256:LGJxexqmZvGZ+8AZF29FJoF64IFNzaGyuTNNVSoEf0.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.20.2.115' (ED25519) to the list of known hosts.
@         WARNING: UNPROTECTED PRIVATE KEY FILE! @
Permissions 0664 for 'Bastion-host-KP.pem' are too open.
It is required that your private key files are NOT accessible by others.
This private key will be ignored.
Load key "Bastion-host-KP.pem": bad permissions
ec2-user@10.20.2.115: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-20-1-102 ~]$ ls
Bastion-host-KP.pem  EmpApp.py  templates
[ec2-user@ip-10-20-1-102 ~]$
```

Activate Windows
Go to Settings to activate Windows.

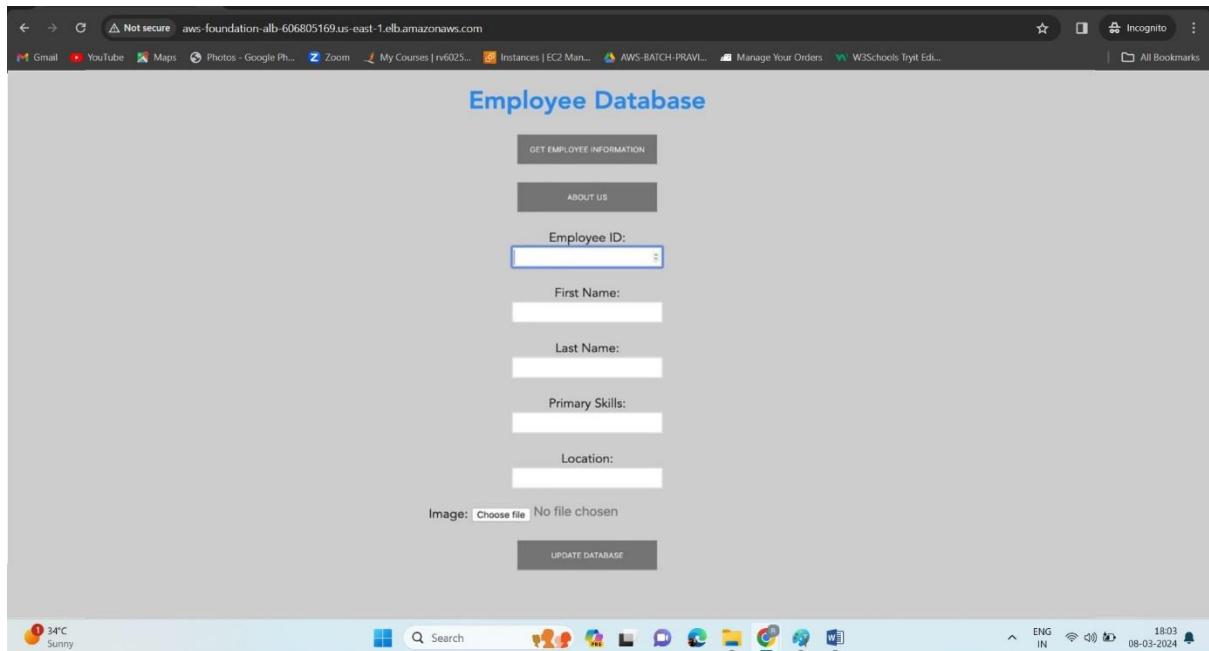
33°C Sunny 14:01 08-03-2024

Now we have to configure the files which is present in our ec2

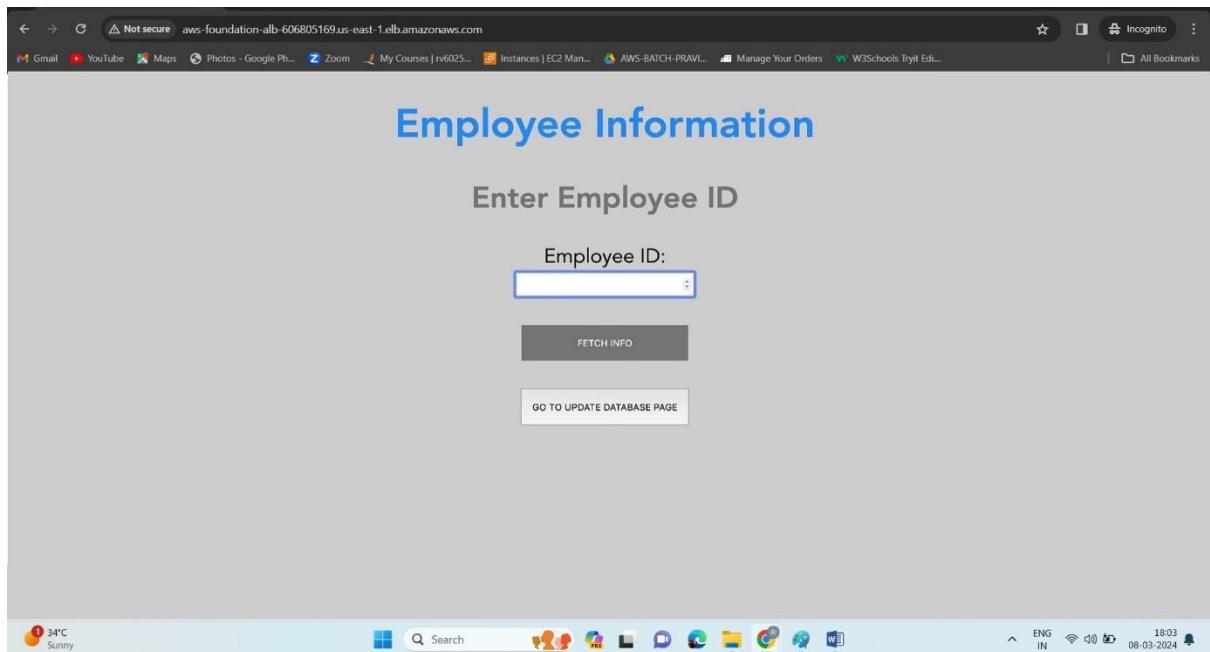
We have to add our loadbalancer dns that's all now let's check our site is working or not by copy pasting load balancer dns in incognito mode



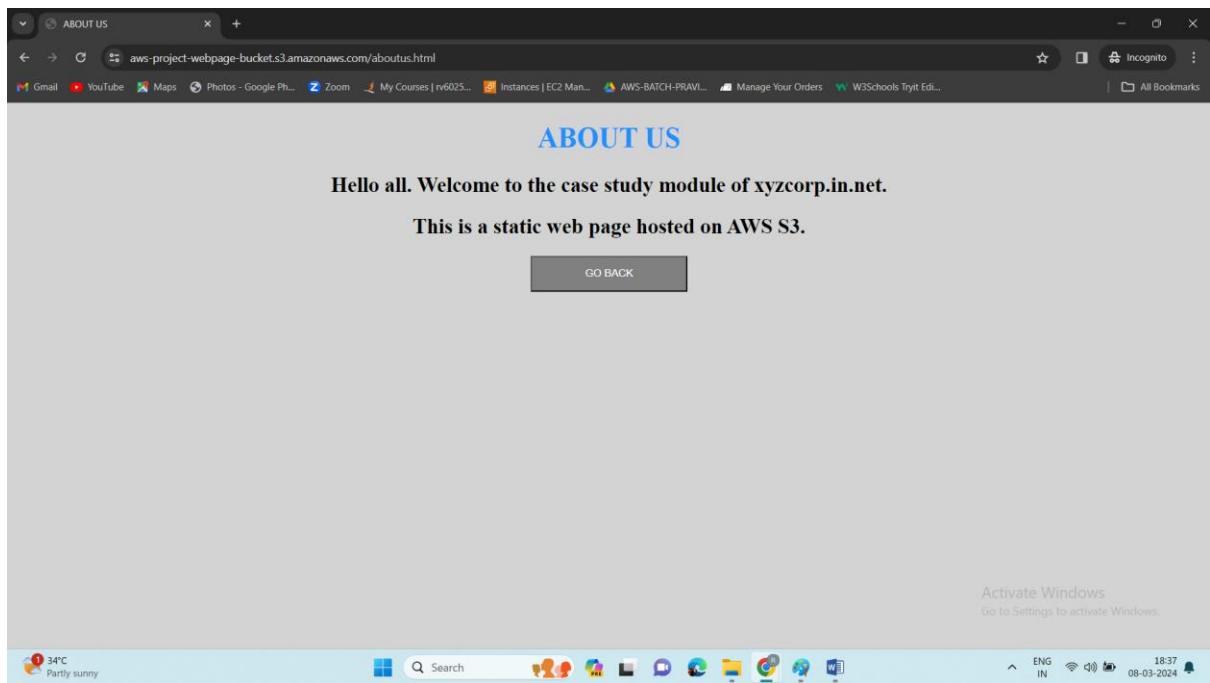
And our application is running



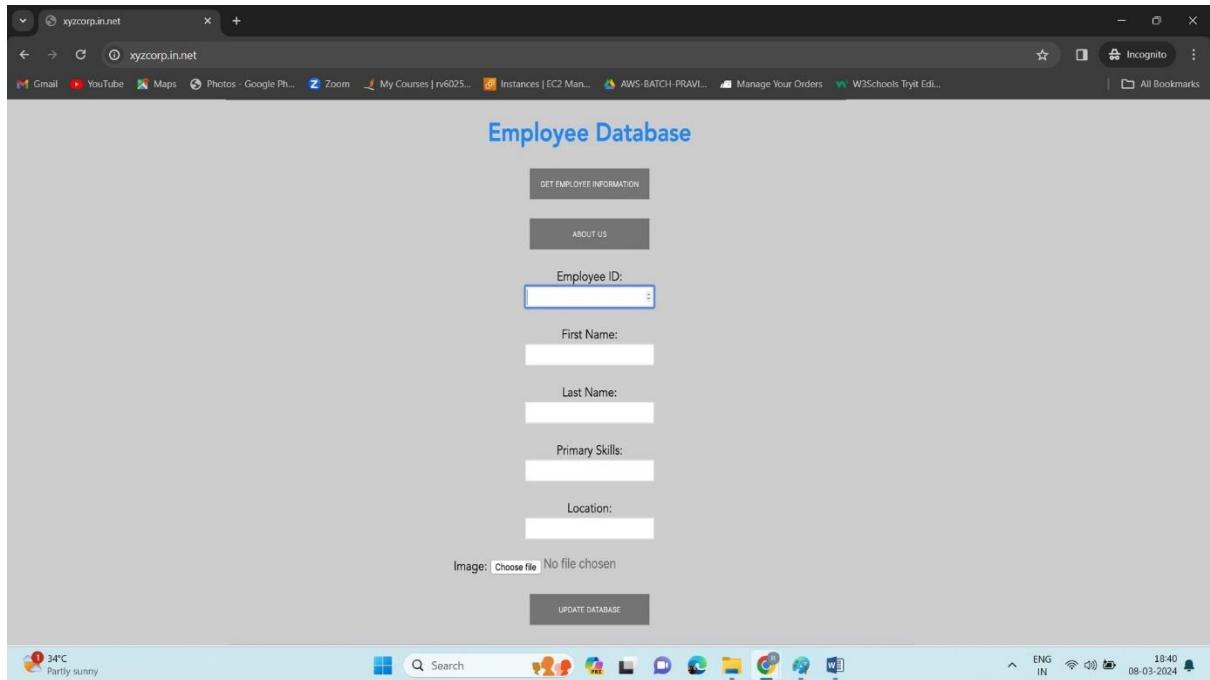
And now let's click it Get employee information



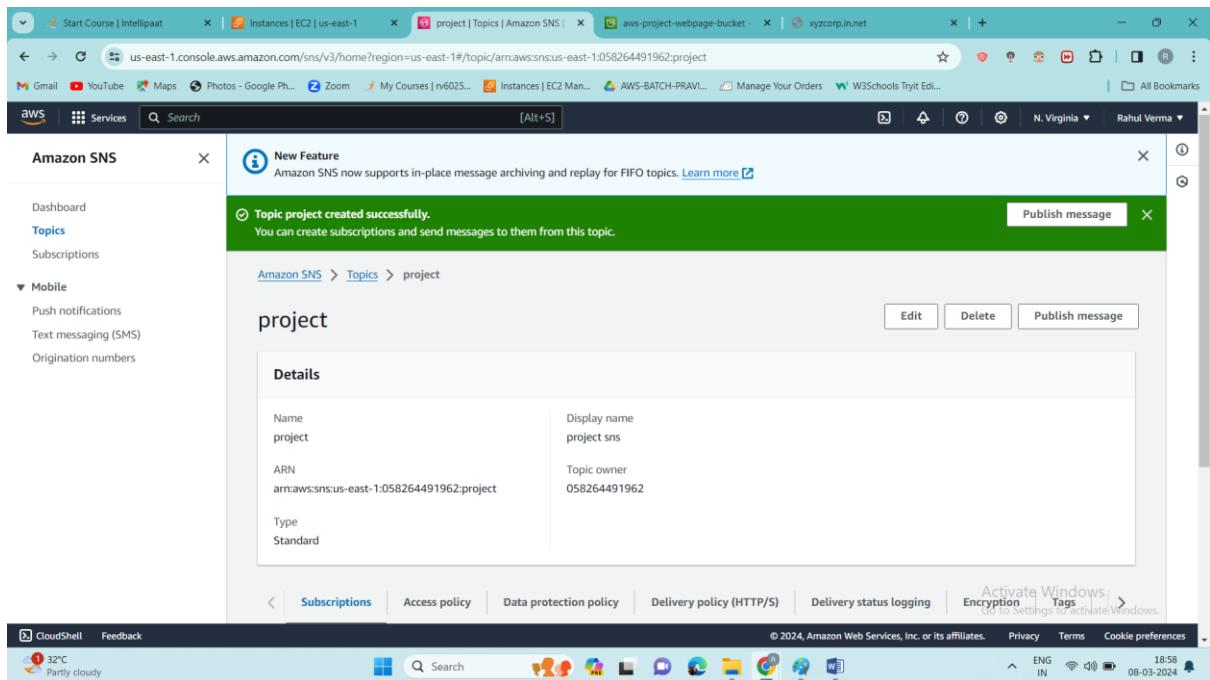
And similarly let's click on About us page



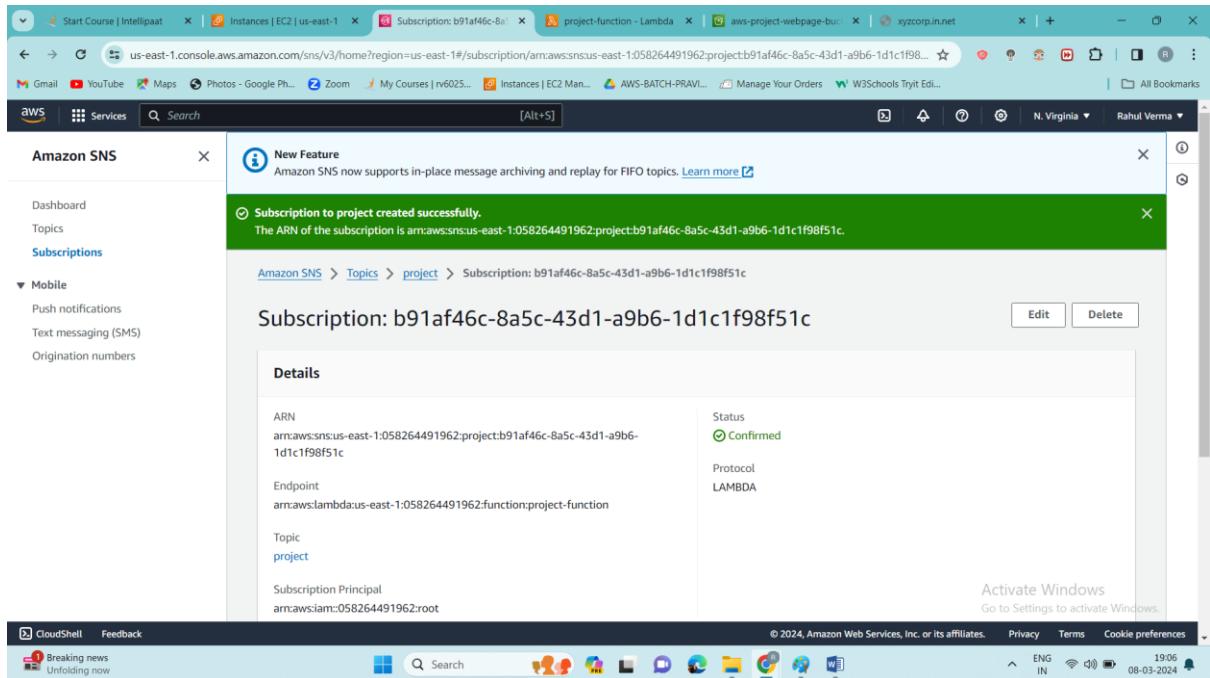
And now we have propagate our load balancer with our hosted zone so let's check if our hosted zone is working or not



And now will create SNS so that we will receive message to our email id whenever and image is uploaded.



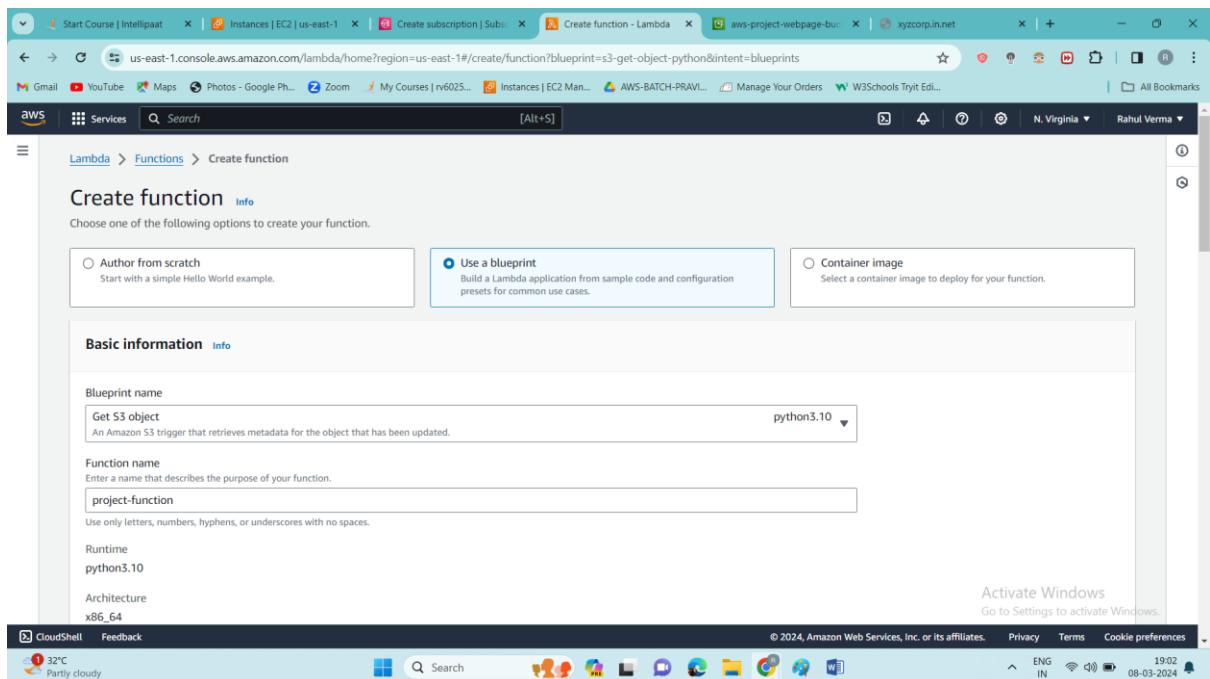
Subscription is done



The screenshot shows the AWS SNS console with a successful subscription creation message. The message states: "Subscription to project created successfully. The ARN of the subscription is arn:aws:sns:us-east-1:058264491962:project:b91af46c-8a5c-43d1-a9b6-1d1c1f98f51c." Below this, the "Subscription" details are listed:

Details	Value
ARN	arn:aws:sns:us-east-1:058264491962:project:b91af46c-8a5c-43d1-a9b6-1d1c1f98f51c
Status	Confirmed
Protocol	LAMBDA
Topic	project
Subscription Principal	arn:aws:iam::058264491962:root

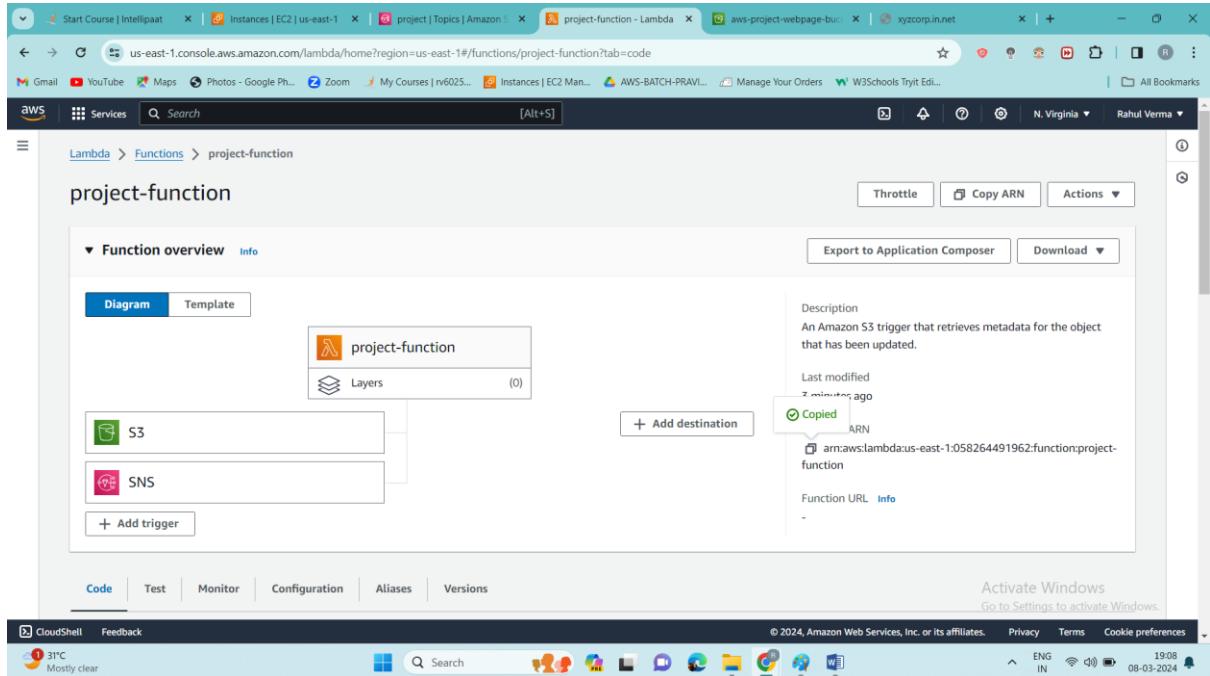
And we have created one Lambda function so that we can create subscription successfully



The screenshot shows the AWS Lambda console with the "Create function" wizard. The "Use a blueprint" option is selected. The "Basic information" section includes:

- Blueprint name: Get S3 object
- Function name: project-function
- Runtime: python3.10
- Architecture: x86_64

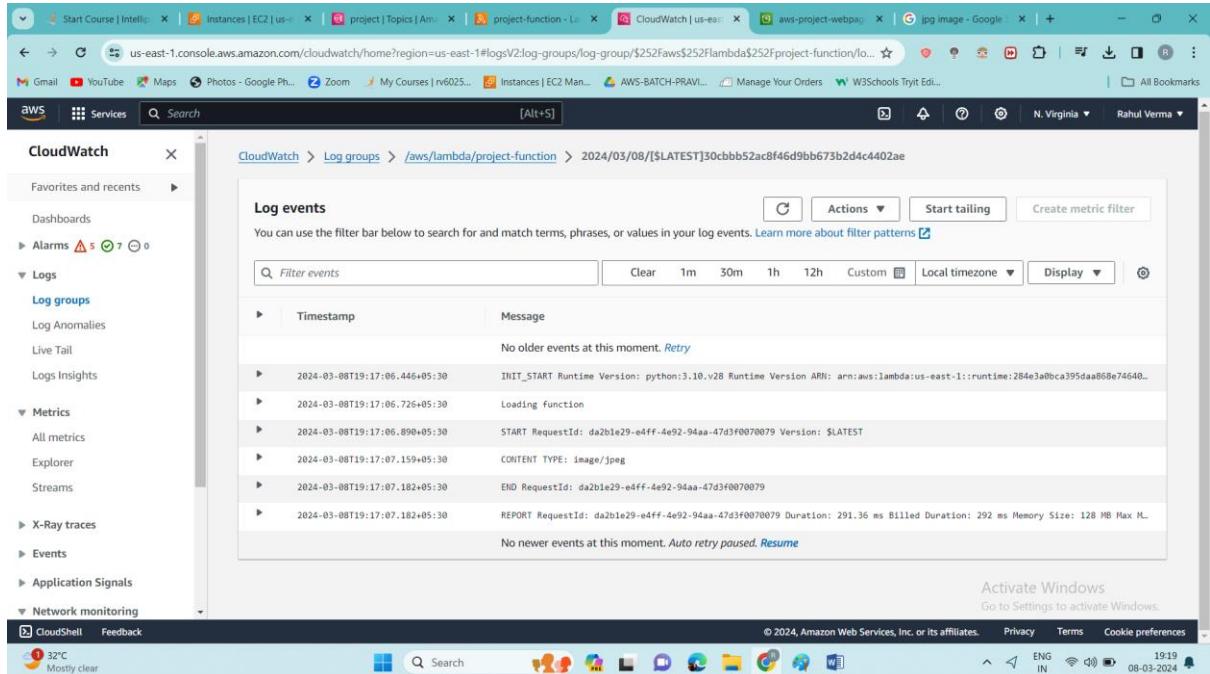
And our lambda function is also created now



The screenshot shows the AWS Lambda console. The left sidebar shows 'Lambda > Functions > project-function'. The main area is titled 'project-function'. It shows a 'Function overview' tab with a 'Diagram' view. The diagram shows the function 'project-function' with a 'Layers' section (0). It has triggers for 'S3' and 'SNS'. Below the diagram are buttons for '+ Add destination', '+ Add trigger', and '+ Add destination'. On the right, there is a 'Description' section: 'An Amazon S3 trigger that retrieves metadata for the object that has been updated.' Below that is 'Last modified' (2 minutes ago) and a green 'Copied' message with the ARN: arn:aws:lambda:us-east-1:058264491962:function:project-function. There are also 'Function URL' and 'Info' buttons. At the bottom, there are tabs for 'Code', 'Test', 'Monitor', 'Configuration', 'Aliases', and 'Versions'. A 'CloudShell' tab is open at the bottom left. The status bar at the bottom right shows 'Activate Windows' and the date '08-03-2024'.

And now let's upload an image

And you can see image is uploaded



The screenshot shows the AWS CloudWatch console. The left sidebar shows 'CloudWatch' with sections for 'Logs', 'Metrics', 'X-Ray traces', 'Events', 'Application Signals', and 'Network monitoring'. The main area shows the 'Log groups' for the '/aws/lambda/project-function' log group. It displays log events for the date '2024/03/08/[\$LATEST]30cbbb52ac8f46d9bb673b2d4c4402ae'. The log events are:

- 2024-03-08T19:17:06.446+05:30 INIT_START Runtime Version: python:3.10.v28 Runtime Version ARN: arn:aws:lambda:us-east-1::runtime:284e3a0bca395daa868e74640...
- 2024-03-08T19:17:06.726+05:30 Loading Function
- 2024-03-08T19:17:06.890+05:30 START RequestId: da2b1e29-e4ff-4e92-94aa-47d3f0070079 Version: \$LATEST
- 2024-03-08T19:17:07.159+05:30 CONTENT TYPE: image/jpeg
- 2024-03-08T19:17:07.182+05:30 END RequestId: da2b1e29-e4ff-4e92-94aa-47d3f0070079
- 2024-03-08T19:17:07.182+05:30 REPORT RequestId: da2b1e29-e4ff-4e92-94aa-47d3f0070079 Duration: 291.36 ms Billed Duration: 292 ms Memory Size: 128 MB Max M...

The status bar at the bottom right shows 'Activate Windows' and the date '08-03-2024'.