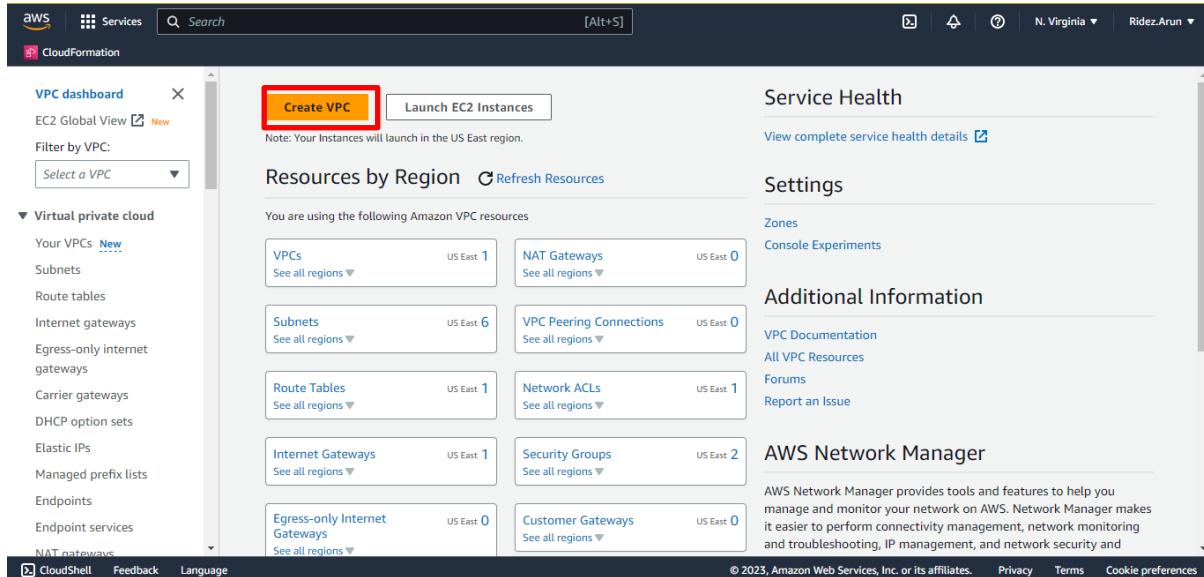


AWS_SNS_PROJECT

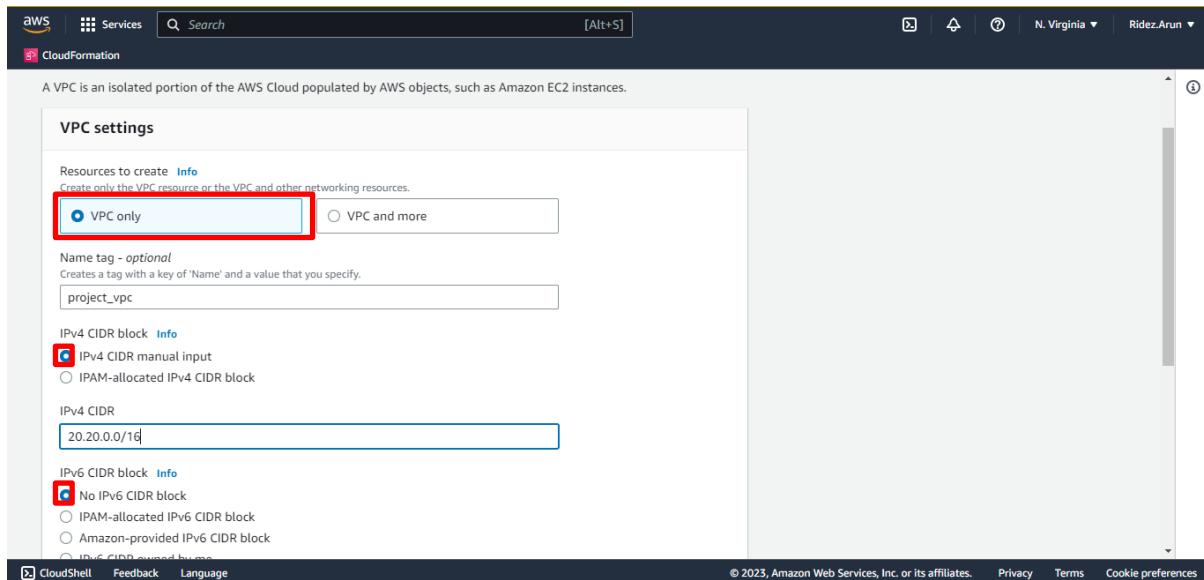
CREATION OF VPC

1) search for vpc in AWS, click create vpc.



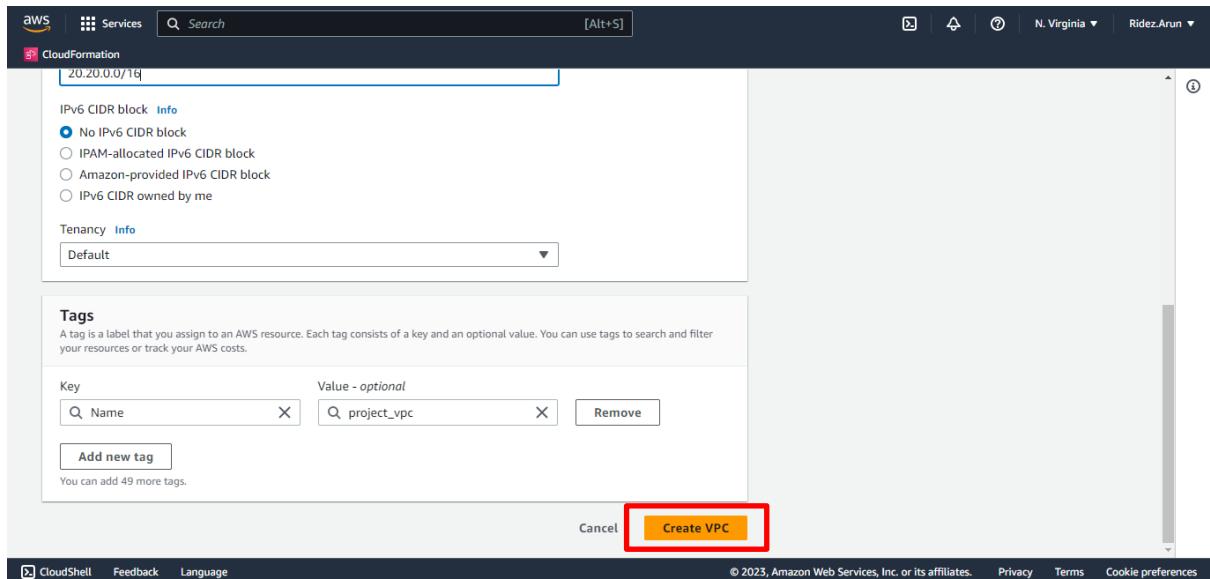
The screenshot shows the AWS VPC dashboard. On the left, there's a sidebar with options like 'Virtual private cloud' (selected), 'Your VPCs' (1), 'Subnets' (6), 'Route tables' (1), 'Internet gateways' (1), 'Egress-only internet gateways' (0), 'Carrier gateways' (0), 'DHCP option sets' (0), 'Elastic IPs' (0), 'Managed prefix lists' (0), 'Endpoints' (0), 'Endpoint services' (0), and 'NAT gateways' (0). The main area has a 'Create VPC' button highlighted with a red box. Below it, there's a section titled 'Resources by Region' showing counts for VPCs (1), Subnets (6), Route Tables (1), Internet Gateways (1), NAT Gateways (0), VPC Peering Connections (0), Network ACLs (1), Security Groups (2), and Customer Gateways (0). To the right, there's a 'Service Health' section and a 'Settings' section with links to 'Zones' and 'Console Experiments'. At the bottom, there's an 'Additional Information' section with links to 'VPC Documentation', 'All VPC Resources', 'Forums', and 'Report an Issue'. The footer includes links for 'CloudShell', 'Feedback', 'Language', and copyright information.

2) choose vpc only,choose ipv4 cidr manual input,choose no ipv6 cidr block, give the Ipv4 as 20.20.0.0/16.

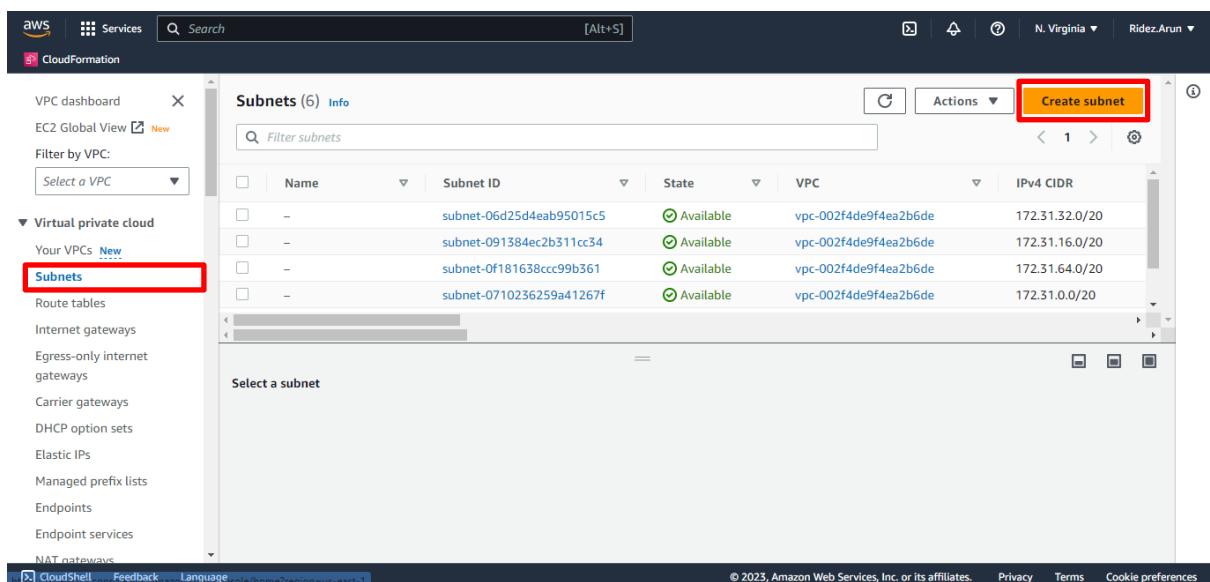


The screenshot shows the 'VPC settings' page. Under 'Resources to create', the 'VPC only' option is selected (highlighted with a red box). There's also an 'Info' link and a note about creating only the VPC resource or the VPC and other networking resources. A 'Name tag - optional' field contains 'project_vpc'. Under 'IPv4 CIDR block', the 'IPv4 CIDR manual input' checkbox is checked (highlighted with a red box), while 'IPAM-allocated IPv4 CIDR block' is unchecked. The 'IPv4 CIDR' field contains '20.20.0.0/16'. Under 'IPv6 CIDR block', the 'No IPv6 CIDR block' checkbox is checked (highlighted with a red box), while 'IPAM-allocated IPv6 CIDR block', 'Amazon-provided IPv6 CIDR block', and 'IPv6 CIDR owned by me' are unchecked. The footer includes links for 'CloudShell', 'Feedback', 'Language', and copyright information.

3) then click create vpc.



4) then click subnet in left side → create subnet.



5) choose the vpc which we created for creation of subnet.

The screenshot shows the 'Create subnet' wizard in the AWS CloudFormation console. In the 'VPC' section, the 'VPC ID' dropdown is set to 'vpc-084f8a5472536f25d (project_vpc)', which is highlighted with a red box. Below it, the 'Associated VPC CIDRs' section shows 'IPv4 CIDRs' as '20.20.0.0/16'. In the 'Subnet settings' section, there is a note: 'Specify the CIDR blocks and Availability Zone for the subnet.' The 'Subnet 1 of 3' section is active, showing 'Subnet name' as 'admin_public1' (highlighted with a red box), 'Availability Zone' as 'US East (N. Virginia) / us-east-1a' (highlighted with a red box), and 'IPv4 CIDR Mask' as '20.20.1.0/24' (highlighted with a red box). The 'Tags - optional' section contains a single tag: 'Name' with value 'admin_public1'. The bottom navigation bar includes 'CloudShell', 'Feedback', 'Language', and links to '© 2023, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

6) Subnet1 should be in public → give name(eg.admin_public1) → choose us-east-1a as availability zone → cidr block as 20.20.1.0/24.

This screenshot continues the 'Create subnet' wizard. The 'Subnet 1 of 3' section is fully visible, showing the configuration for Subnet 1. The 'Subnet name' is 'admin_public1', 'Availability Zone' is 'US East (N. Virginia) / us-east-1a', and 'IPv4 CIDR Mask' is '20.20.1.0/24'. The 'Tags - optional' section shows a single tag 'Name' with value 'admin_public1'. The bottom navigation bar includes 'CloudShell', 'Feedback', 'Language', and links to '© 2023, Amazon Web Services, Inc. or its affiliates.', 'Privacy', 'Terms', and 'Cookie preferences'.

7) Subnet2 should be in public → give name (eg.admin_public2) → choose us-east-1b as availability zone → cidr block as 20.20.2.0/24.

The screenshot shows the AWS CloudFormation console with the 'Subnet 2 of 3' configuration page. The subnet name is set to 'admin_public2'. The availability zone is chosen as 'US East (N. Virginia) / us-east-1b'. The IPv4 CIDR block is set to '20.20.2.0/24'. A single tag 'Name: admin_public2' is added. The page also includes sections for 'Subnet 3 of 3' and navigation links like CloudShell, Feedback, Language, Privacy, Terms, and Cookie preferences.

- 8) Subnet3 should be in private → give name (eg.admin_private) → choose us-east-1a as availability zone → cidr block as 20.20.3.0/24 → click create → the subnet will be created successfully.

The screenshot shows the AWS CloudFormation console with the 'Subnet 3 of 3' configuration page. The subnet name is set to 'admin_private'. The availability zone is chosen as 'US East (N. Virginia) / us-east-1a'. The IPv4 CIDR block is set to '20.20.3.0/24'. A single tag 'Name: admin_private' is added. The page includes a 'Add new subnet' button and navigation links like CloudShell, Feedback, Language, Privacy, Terms, and Cookie preferences.

- 9) Now we need to create a route table → click route tables in left side → here you can find the default route table that automatically created while creating a vpc rename it as admin_routepublic → then click create route table.

The screenshot shows the AWS CloudFormation interface for managing VPC resources. On the left, a sidebar lists various VPC components: 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. The 'Route tables' item is selected and highlighted with a red box. The main content area displays a table titled 'Route tables (2)'. The table has columns for Name, Route table ID, Explicit subnet associations, Edge associations, and Main. It lists two entries: one with a blank name and ID 'rtb-0db2a357558aaae8d', and another with a blank name and ID 'rtb-0ba8b639d96048f84'. Both rows have a 'Yes' value in the Main column. Below the table, a section titled 'Select a route table' is visible.

10) The default route table has been renamed as admin_routepublic → now click create route table.

This screenshot continues from the previous one, showing the 'admin_routepublic' route table selected in the list. A red box highlights the row for 'admin_routepublic'. The 'Create route table' button is also highlighted with a red box. The main content area shows the details for the selected route table, including tabs for Details, Routes, Subnet associations, Edge associations, Route propagation, and Tags. A message at the bottom says 'You can now check network connectivity with Reachability Analyzer' with a 'Run Reachability Analyzer' button.

11) Give name as admin_routeprivate → choose the vpc we created.

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="admin_routeprivate"/>

Add new tag
You can add 49 more tags.

Create route table

12) The route table has been created successfully.

Route tables (3)

Name	Route table ID	Explicit subnet associations	Edge associations	Main
-	rtb-0ba8b639d96048f84	-	-	Yes
admin_routeprivate	rtb-00b9c1558ffadb88f	-	-	No
admin_routepublic	rtb-0db2a357558aae8d	-	-	Yes

13) Select the private routetable → go to subnet association.

VPC dashboard

EC2 Global View New

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs New

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

CloudShell Feedback Language

Route tables (1/3) Info

Find resources by attribute or tag

Name	Route table ID	Explicit subnet associations	Edge associations	Main
admin_routepublic	rtb-0db2a357558aae8d	-	-	Yes
-	rtb-0ba8b639d96048f84	-	-	Yes
admin_routeprivate	rtb-00b9c1558ffadb88f	-	-	No

rtb-00b9c1558ffadb88f / admin_routeprivate

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (0)

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
------	-----------	-----------	-----------

Edit subnet associations

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

14) associate the private subnet → save association.

VPC > Route tables > rtb-00b9c1558ffadb88f > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (1/3)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
admin_private	subnet-0073d7946686422c2	20.20.3.0/24	-	Main (rtb-0db2a357558aae8d)
admin_public1	subnet-09f5835a168a83382	20.20.1.0/24	-	Main (rtb-0db2a357558aae8d /)
admin_public2	subnet-08d80e5bb57fa15ac	20.20.2.0/24	-	Main (rtb-0db2a357558aae8d /)

Selected subnets

subnet-0073d7946686422c2 / admin_private X

Cancel **Save associations**

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

15 As the same select the public route table → go to subnet association.

VPC dashboard

EC2 Global View New

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs New

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

NAT gateways

CloudShell Feedback Language

Route tables (1/3) Info

Find resources by attribute or tag

Name	Route table ID	Explicit subnet associations	Edge associations	Main
admin_routepublic	rtb-0db2a357558aaaae8d	-	-	Yes
-	rtb-0ba8b639d96048f84	-	-	Yes
admin_routeprivate	rtb-00b9c1558ffadb88f	subnet-0073d794668642...	-	No

rtb-0db2a357558aaaae8d / admin_routepublic

Details | Routes | **Subnet associations** | Edge associations | Route propagation | Tags

Explicit subnet associations (0)

Find subnet association

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR
------	-----------	-----------	-----------

Edit subnet associations

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

16) Associate both the public subnet to the public route table → save association.

AWS Services Search [Alt+S]

CloudFormation

VPC > Route tables > rtb-0db2a357558aaaae8d > Edit subnet associations

Edit subnet associations

Change which subnets are associated with this route table.

Available subnets (2/3)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
admin_private	subnet-0073d7946686422c2	20.20.3.0/24	-	rtb-00b9c1558ffadb88f / admin_routepublic
<input checked="" type="checkbox"/> admin_public1	subnet-09f5835a168a83382	20.20.1.0/24	-	Main (rtb-0db2a357558aaaae8d / admin_routepublic)
<input checked="" type="checkbox"/> admin_public2	subnet-08d80e5bb57fa15ac	20.20.2.0/24	-	Main (rtb-0db2a357558aaaae8d / admin_routepublic)

Selected subnets

subnet-09f5835a168a83382 / admin_public1 X | subnet-08d80e5bb57fa15ac / admin_public2 X

Cancel Save associations

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

17) Now create an internet gateway → click internet gateway in left panel → click create internet gateway.

VPC dashboard

EC2 Global View New

Filter by VPC: Select a VPC

Virtual private cloud

Your VPCs New

Subnets

Route tables

Internet gateways

Egress-only internet gateways

Carrier gateways

DHCP option sets

Elastic IPs

Managed prefix lists

Endpoints

Endpoint services

Internet gateways (1/1) Info

Actions Create internet gateway

Name	Internet gateway ID	State	VPC ID	Owner
-	igw-03e268ca55c4976e9	Attached	vpc-002f4de9f4ea2b6de	25850646

igw-03e268ca55c4976e9

Details Tags

Details

Internet gateway ID State VPC ID Owner

18) give name to internet gateway(eg.admin_internetgateway) → click create.

Internet gateway settings

Name tag

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

admin_internetgateway

Tags - optional

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

Name admin_internetgateway Remove

Add new tag

You can add 49 more tags.

Cancel Create internet gateway

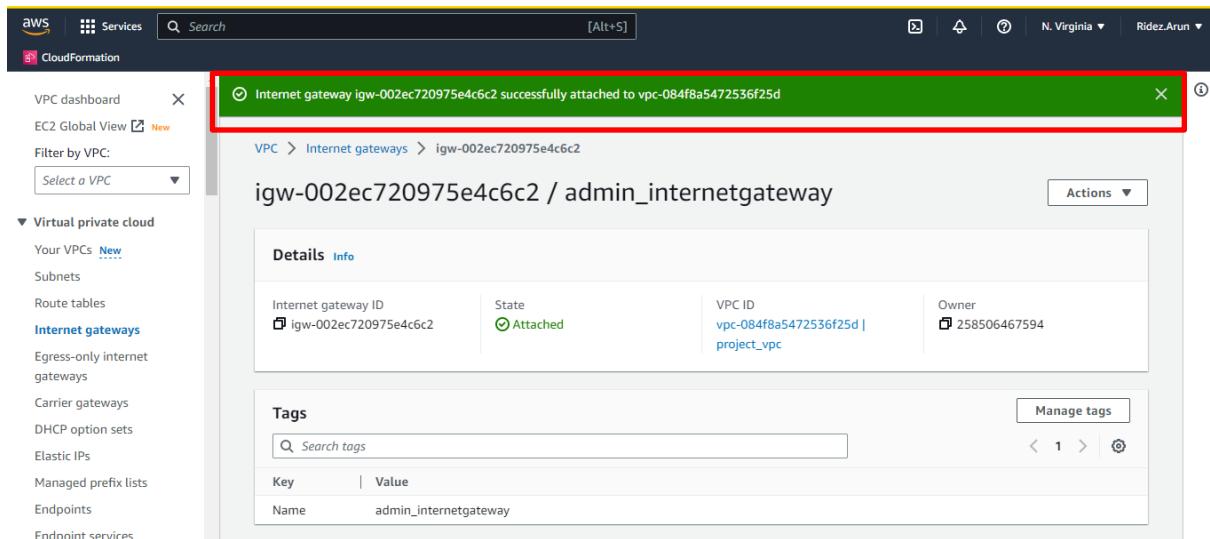
19) It will ask to attach to a vpc → click attach to a vpc.

The screenshot shows the AWS VPC Internet Gateways page. A message at the top states: "The following internet gateway was created: igw-002ec720975e4c6c2 - admin_internetgateway. You can now attach to a VPC to enable the VPC to communicate with the internet." A red box highlights the "Attach to a VPC" button. Below this, the internet gateway details are shown: ID igw-002ec720975e4c6c2, State Detached, VPC ID -, Owner 258506467594. The Tags section shows a single tag named "admin_internetgateway".

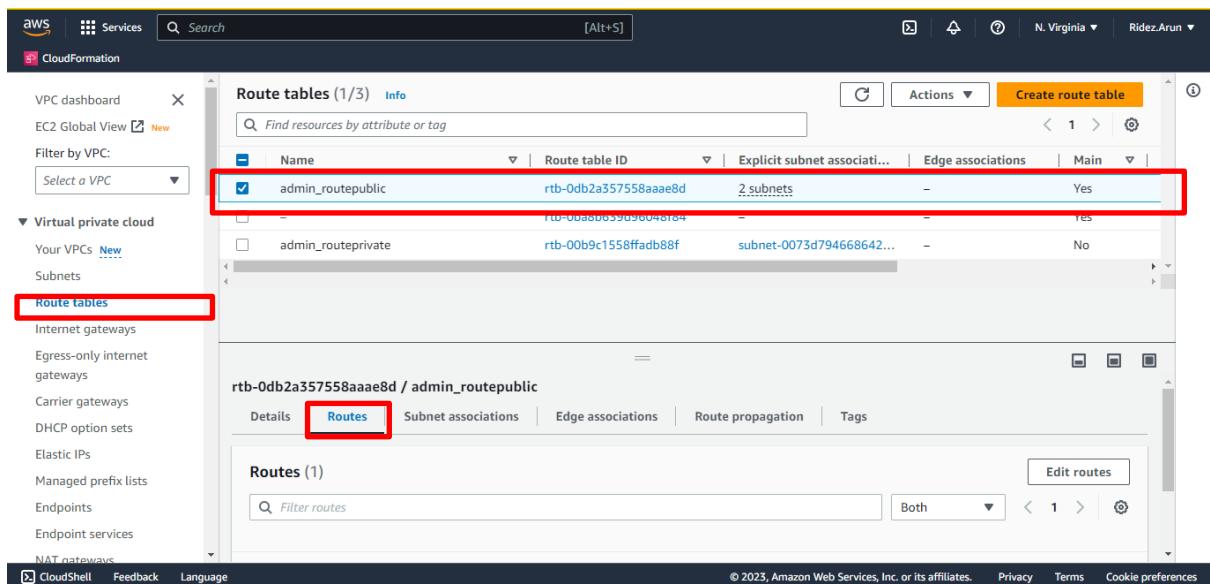
20) choose the vpc we created → click attach.

The screenshot shows the "Attach to VPC" dialog box. It asks to attach an internet gateway to a VPC. A red box highlights the search bar where "vpc-084f8a5472536f25d" is typed. Another red box highlights the "Attach internet gateway" button.

21) The internet gateway has been created and attached to vpc successfully.



22) Go to route table again → select the public route table → go to routes → click edit routes.



23) Select the destination as 0.0.0.0/0 and target as internet gateway and attach it → save changes.

The screenshot shows the 'Edit routes' page for a specific route table. A new route is being added with the destination '0.0.0.0/0' and target 'igw-002ec720975e4c6c2'. The 'Save changes' button is highlighted with a red box.

24) Now we need to create NAT gateway → click NAT gateway in left panel → click create NAT gateway.

The screenshot shows the 'NAT gateways' section of the AWS VPC console. The 'NAT gateways' link in the left sidebar is highlighted with a red box. The 'Create NAT gateway' button in the top right corner is also highlighted with a red box.

25) Give name to NAT gateway (eg.admin_NAT) → subnet as admin_public1 → connectivity type should be in public.

The screenshot shows the 'Create NAT gateway' wizard in the AWS CloudFormation console. The current step is 'NAT gateway settings'. The form includes fields for 'Name - optional' (containing 'admin_nat'), 'Subnet' (selected as 'subnet-09f5835a168a83382 (admin_public1)'), and 'Connectivity type' (set to 'Public'). The 'CloudShell', 'Feedback', and 'Language' buttons are at the bottom left, and the '© 2023, Amazon Web Services, Inc. or its affiliates.' and 'Cookie preferences' links are at the bottom right.

26) Click allocate elastic ip→click create.

The screenshot shows the 'Allocate Elastic IP' step in the AWS CloudFormation console. It displays an elastic IP allocation ID 'eipalloc-0c0af4e83af0f3793' and an 'Allocate Elastic IP' button, which is highlighted with a red box. Below this, there's a 'Tags' section where a tag 'Name' is added with value 'admin_nat'. At the bottom, there are 'Cancel' and 'Create NAT gateway' buttons, with 'Create NAT gateway' also highlighted with a red box.

27) the NAT gateway has been created successfully.

The screenshot shows the AWS CloudFormation VPC dashboard. A green success message at the top states: "NAT gateway nat-04a7c59f512ab48e1 | admin_nat was created successfully." Below this, a table lists the NAT gateway details:

ARN	Type	Subnet	Status
arn:aws:ec2:us-east-1:258506467594:natgateway/nat-04a7c59f512ab48e1	NAT gateway	subnet-09f5835a168a83382 / admin_public	Created
	VPC	vpc-084f8a5472536f25d / project_vpc	Tuesday, May 23, 2023 at 17:09:33 GMT+5:30
			Deleted

Below the table, there are tabs for "Secondary IPv4 addresses", "Monitoring", and "Tags". The "Secondary IPv4 addresses" tab shows a message: "Secondary IPv4 addresses are not available for this nat gateway." At the bottom, there are links for "CloudShell", "Feedback", and "Language".

28) Again go the route table → select the private route table → go to routes → click edit routes.

The screenshot shows the AWS CloudFormation Route tables page. A red box highlights the "Route tables" link in the navigation sidebar. The main table lists route tables:

Name	Route table ID	Explicit subnet associations	Edge associations	Main
admin_routeprivate	rtb-00b9c1558ffadb88f	subnet-0073d794668642...	-	No
admin_routepublic	rtb-0db2a357558aae8d	2 subnets	-	Yes
-	rtb-0ba8b639d96048f84	-	-	Yes

Below the table, a detailed view of the selected route table "rtb-00b9c1558ffadb88f / admin_routeprivate" is shown. A red box highlights the "Routes" tab. The "Routes" section displays one route:

Destination	Target
0.0.0.0/0	NAT gateway

At the bottom, there are tabs for "Details", "Routes", "Subnet associations", "Edge associations", "Route propagation", and "Tags". There is also a "Edit routes" button.

29) Select the destination as 0.0.0.0/0 and target as NAT gateway then attach NAT gateway → save changes.

VPC > Route tables > rtb-00b9c1558ffadb88 > Edit routes

Edit routes

Destination	Target	Status	Propagated
20.20.0.0/16	local	Active	No
0.0.0.0/0	nat-04a7c59f512ab48e1	-	No

Add route

Cancel Preview Save changes

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

30) Now we need to create a security group → select security group in left panel → click create security group.

Egress-only internet gateways
Carrier gateways
DHCP option sets
Elastic IPs
Managed prefix lists
Endpoints
Endpoint services
NAT gateways
Peering connections

▼ Security
Network ACLs
Security groups **highlighted with red box**
▼ DNS firewall
Rule groups
Domain lists
▼ Network Firewall

Security Groups Info Actions Export security groups to CSV Create security group **highlighted with red box**

Name Security group ID Security group name VPC ID Description

https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1 © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

31) Give name to security group → choose our vpc → click create.

Create security group [Info](#)

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.

Basic details

Security group name [Info](#)
admin_security

Description [Info](#)
Allows SSH access to developers

VPC [Info](#)
Q vpc-084f8a5472536f25d

Inbound rules [Info](#)

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Give inbound and outbound type as all traffic → source as anywhere → click create.

Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info
All traffic	All	All	Anywhere	0.0.0.0/0

Add rule Delete

Outbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Destination Info	Description - optional Info
All traffic	All	All	Custom	0.0.0.0/0

Add rule Delete

32) Now go back to subnet → choose the public subnet1 → actions → edit subnet settings.

VPC dashboard EC2 Global View New Filter by VPC: Select a VPC Virtual private cloud Your VPCs New Subnets Route tables Internet gateways Egress-only internet gateways Carrier gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways CloudShell Feedback Language

Subnets (1/9) Info

	Name	Subnet ID	Status	VPC ARN
<input type="checkbox"/>	admin_private	subnet-0073d7946686422c2	Available	vpc-004f8a5472536f
<input checked="" type="checkbox"/>	admin_public1	subnet-09f5835a168a83382	Available	vpc-084f8a5472536f
<input type="checkbox"/>	admin_public2	subnet-08d80e5bb57fa15ac	Available	vpc-084f8a5472536f

Actions ▾ Create subnet View details Create flow log Edit subnet settings Edit network ACL association Edit route table association Edit CIDR reservations

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

33) Enable the auto-assign ip → click save.

VPC > Subnets > subnet-09f5835a168a83382 > Edit subnet settings

Edit subnet settings Info

Subnet

Subnet ID	Name
subnet-09f5835a168a83382	admin_public1

Auto-assign IP settings Info

Enable the auto-assign IP settings to automatically request a public IPv4 or IPv6 address for a new network interface in this subnet.

Enable auto-assign public IPv4 address Info

Enable auto-assign customer-owned IPv4 address Info

Option disabled because no customer owned pools found.

Resource-based name (RBN) settings Info

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

34) As the same select the public subnet2 → actions → edit subnet settings.

VPC dashboard EC2 Global View New Filter by VPC: Select a VPC Virtual private cloud Your VPCs New Subnets Route tables Internet gateways Egress-only internet gateways Carrier gateways DHCP option sets Elastic IPs Managed prefix lists Endpoints Endpoint services NAT gateways

Subnets (1/9) Info Filter subnets Actions ▾ Create subnet View details Create flow log Edit subnet settings Edit IPv6 CIDRs Edit network ACL association Edit route table association Edit CIDR reservations Edit CIDR reservations 20.20.2.0/24

subnet-08d80e5bb57fa15ac / admin_public2

Details Flow logs Route table Network ACL CIDR reservations Sharing Tags

Subnet ID	Subnet ARN	State	IPv4 CIDR
subnet-08d80e5bb57fa15ac	vpc-084f8a5472536f	Available	20.20.2.0/24

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

35) Enable the auto-assign ip → click save.

VPC > Subnets > subnet-09f5835a168a83382 > Edit subnet settings

Edit subnet settings

Subnet

Subnet ID	Name
subnet-09f5835a168a83382	admin_public1

Auto-assign IP settings

Enable the auto-assign IP settings to automatically request a public IPv4 or IPv6 address for a new network interface in this subnet.

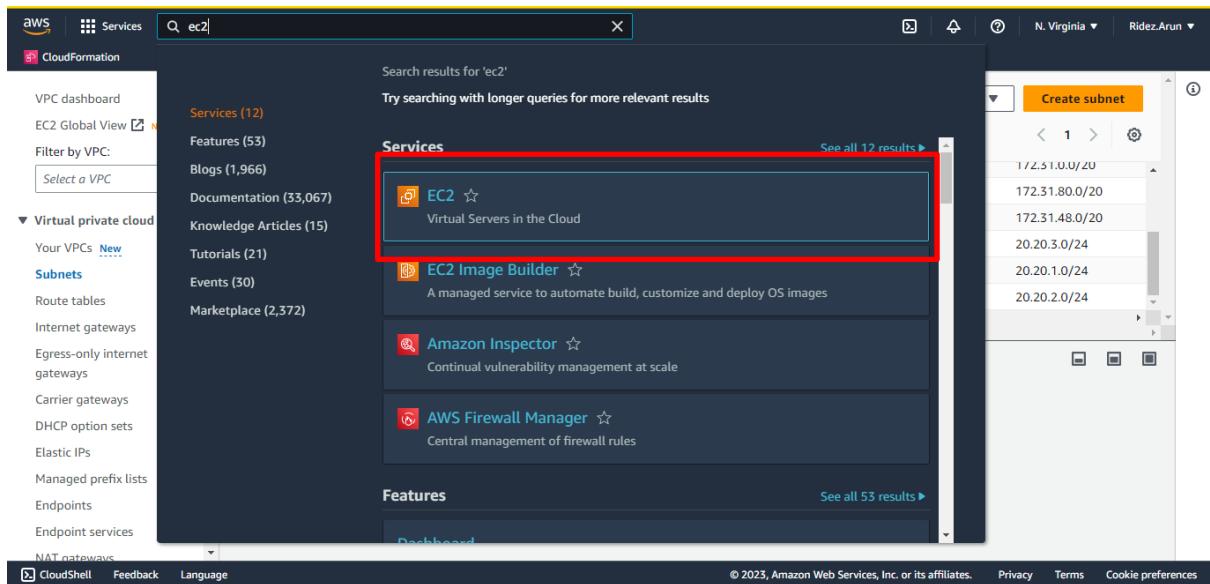
Enable auto-assign public IPv4 address

Enable auto-assign customer-owned IPv4 address

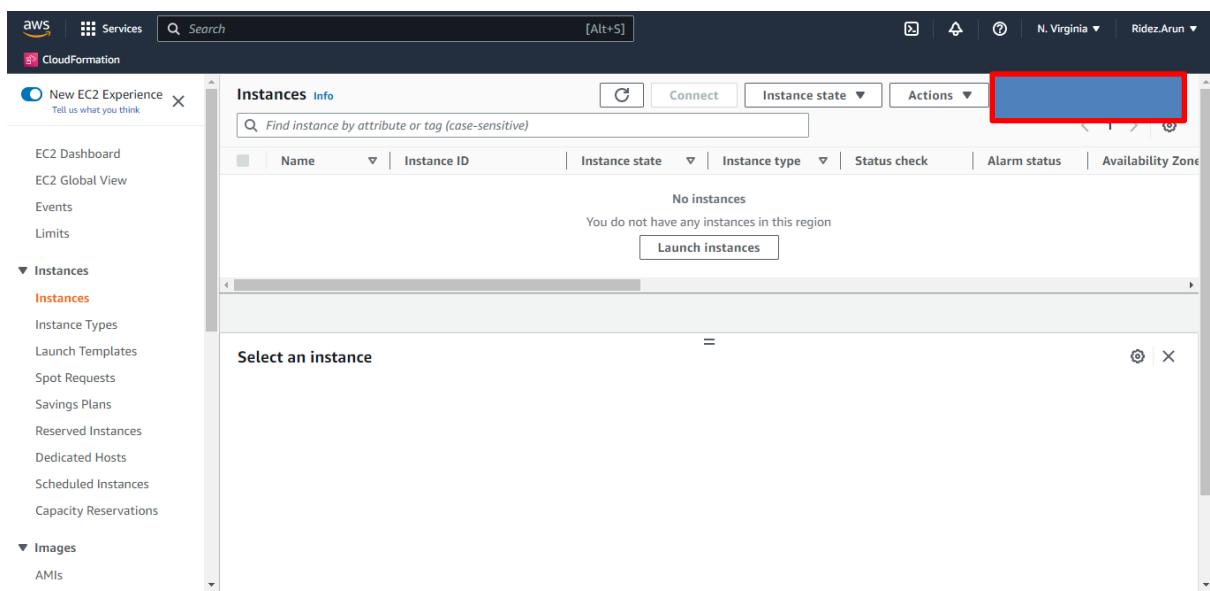
Resource-based name (RBN) settings

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

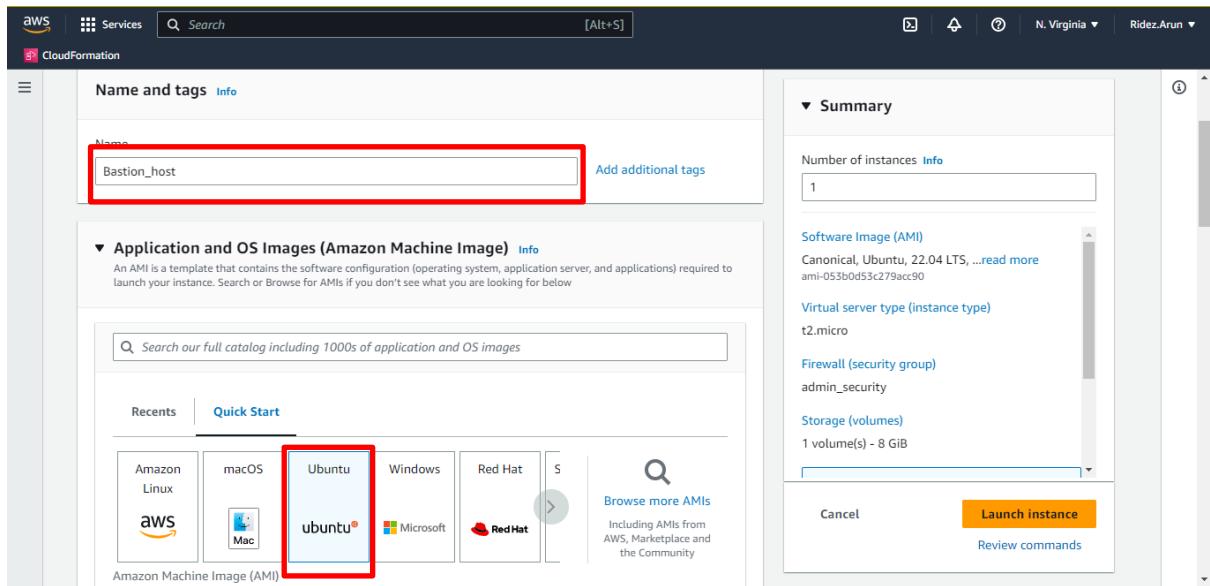
36) Search for ec2 in aws.



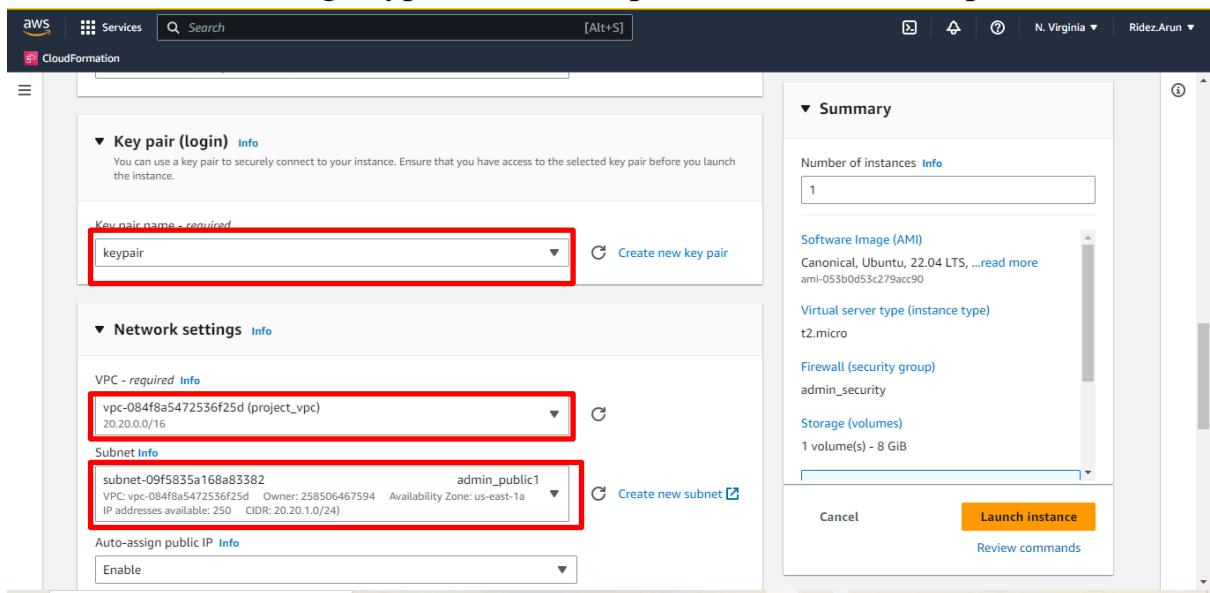
37) Click launch instance.



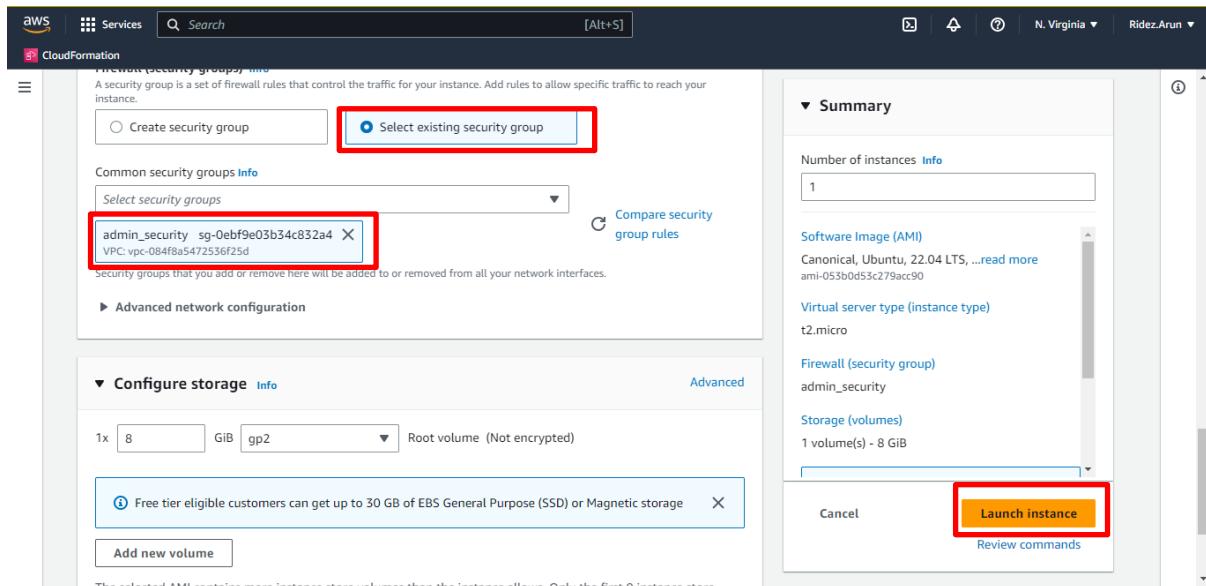
38) Give name→choose Ubuntu.



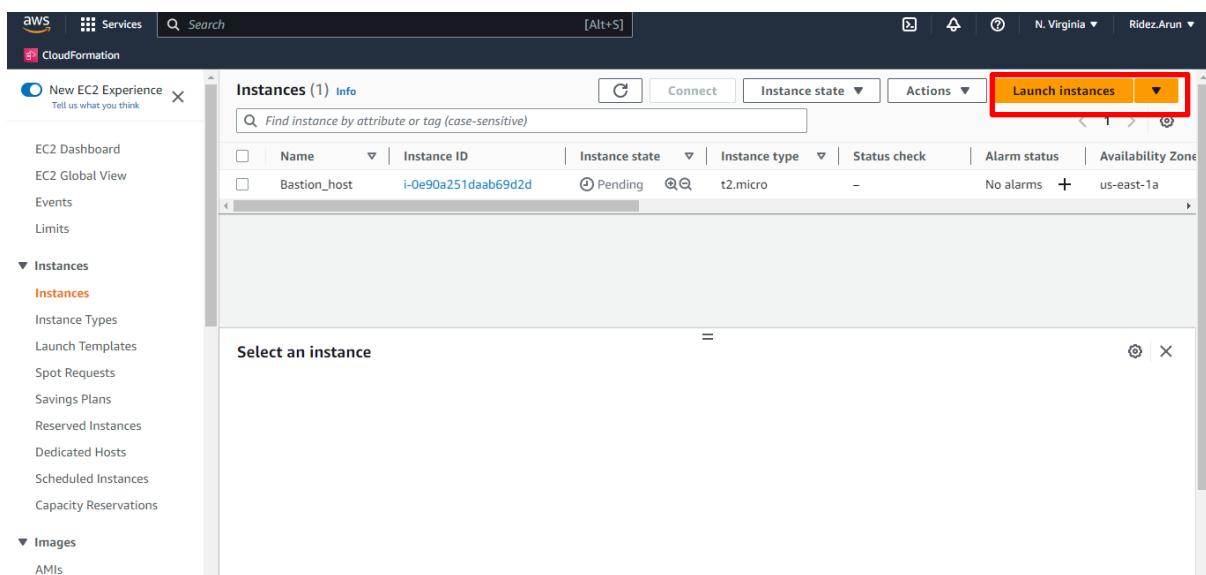
39) Select the existing keypair → choose vpc we created → choose public subnet1.



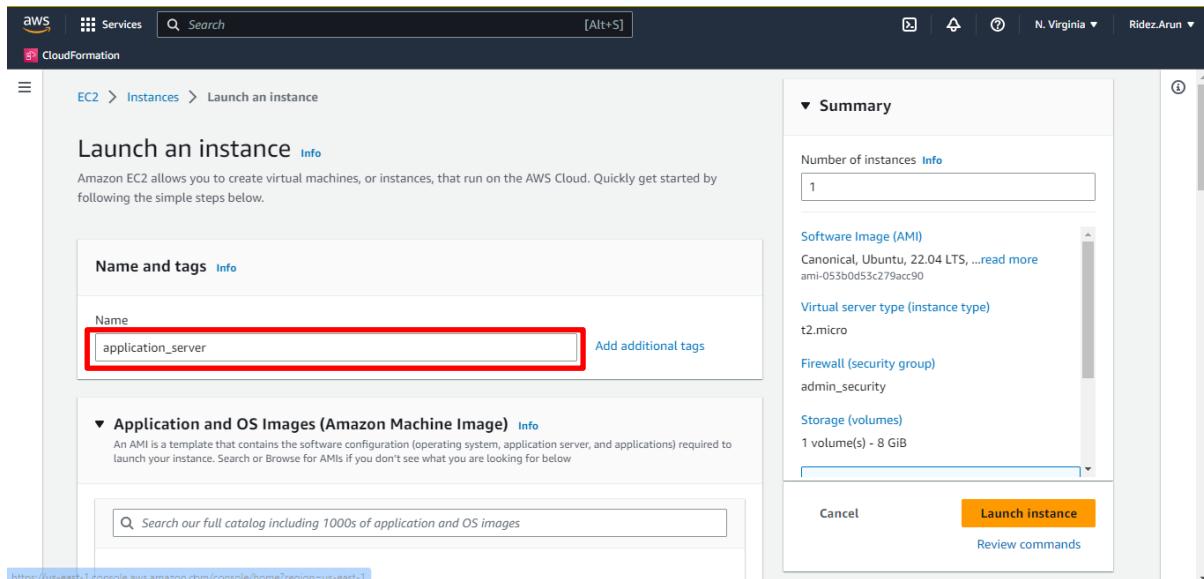
40) Choose the existing security group → select the security group we created → launch instance.



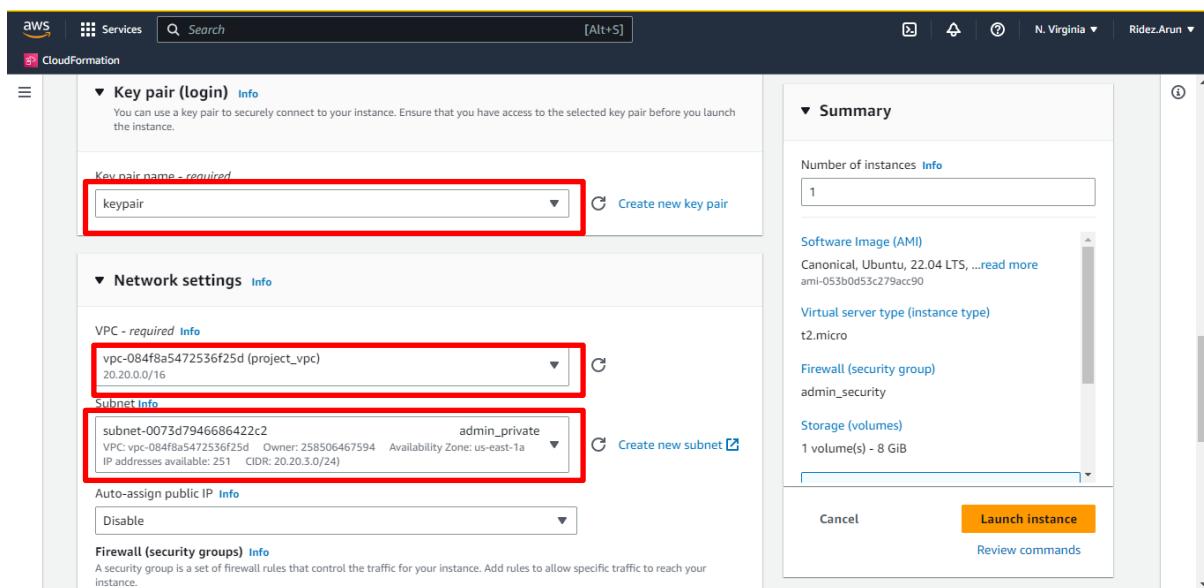
41) As the same create another instance → click launch instance.



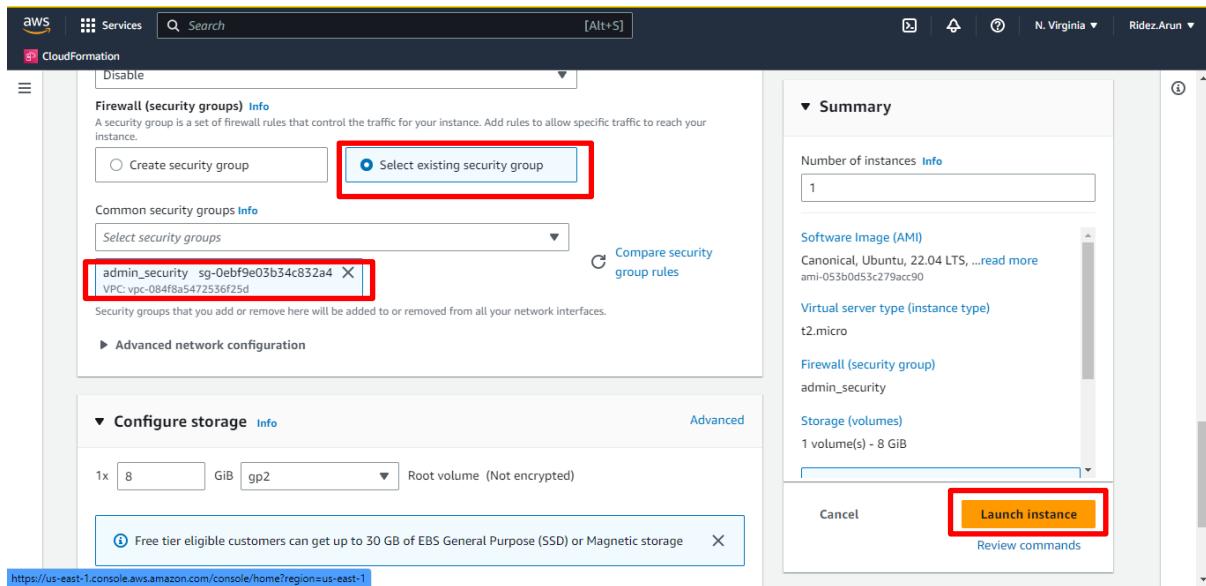
42) Give name → choose Ubuntu.



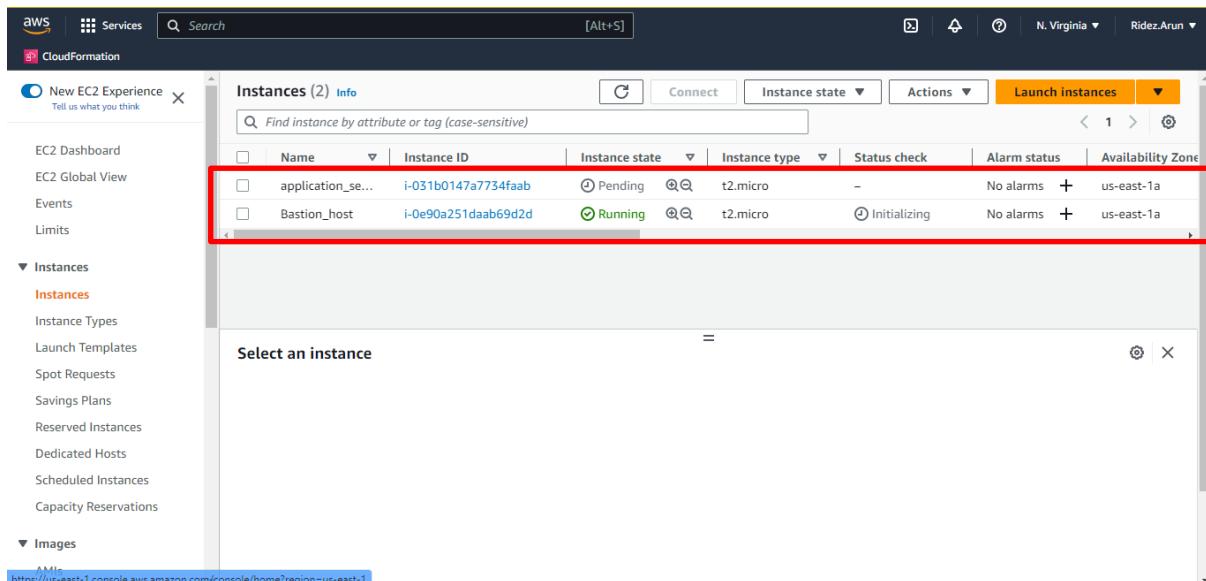
43) Choose the existing keypair → choose vpc we created → choose private subnet.



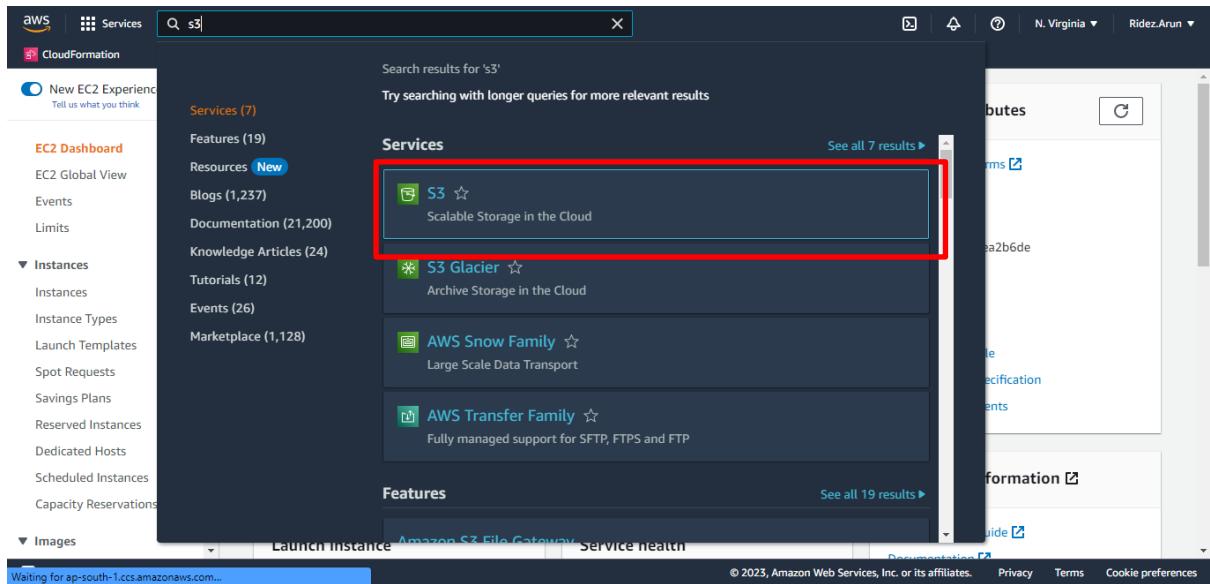
44) Select existing security group → choose the security group we created → launch instance.



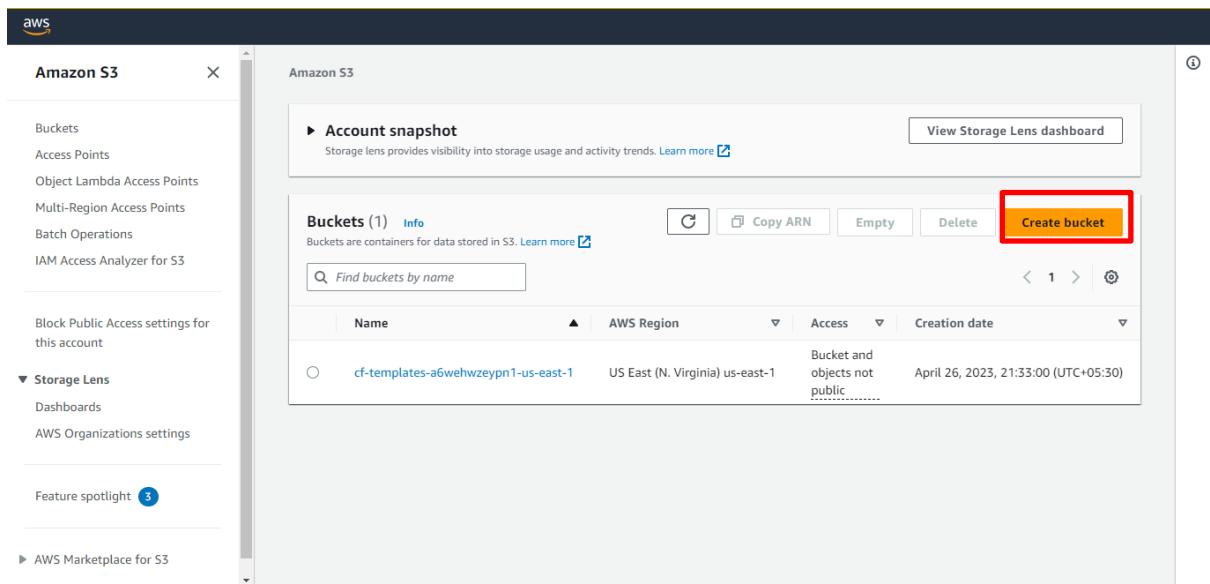
45) Both private and public instance has been created successfully.



46) Search for s3 in aws.



47) Click create bucket.



48) Give bucket name → choose us-east-1 as Availability region.

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name: admin_bucket
Bucket name must be globally unique and must not contain spaces or uppercase letters. See [rules for bucket naming](#).

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

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

[Choose bucket](#)

Object Ownership

<https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1>

49) Select ACLs enabled → choose bucket owner preferred.

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner preferred
If new objects written to this bucket specify the bucket-owner-full-control canned ACL, they are owned by the bucket owner. Otherwise, they are owned by the object writer.

Object writer
The object writer remains the object owner.

If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. [Learn more](#)

<https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1>

50) Disable block all public access.

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

Warning Turning off block all public access might result in this bucket and the objects within becoming public

CloudShell Feedback Language [Role Home](#) [Region: US-East-1](#)

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

51) Disable versioning.

Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

Disable

Enable

Tags (0) - optional

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

No tags associated with this bucket.

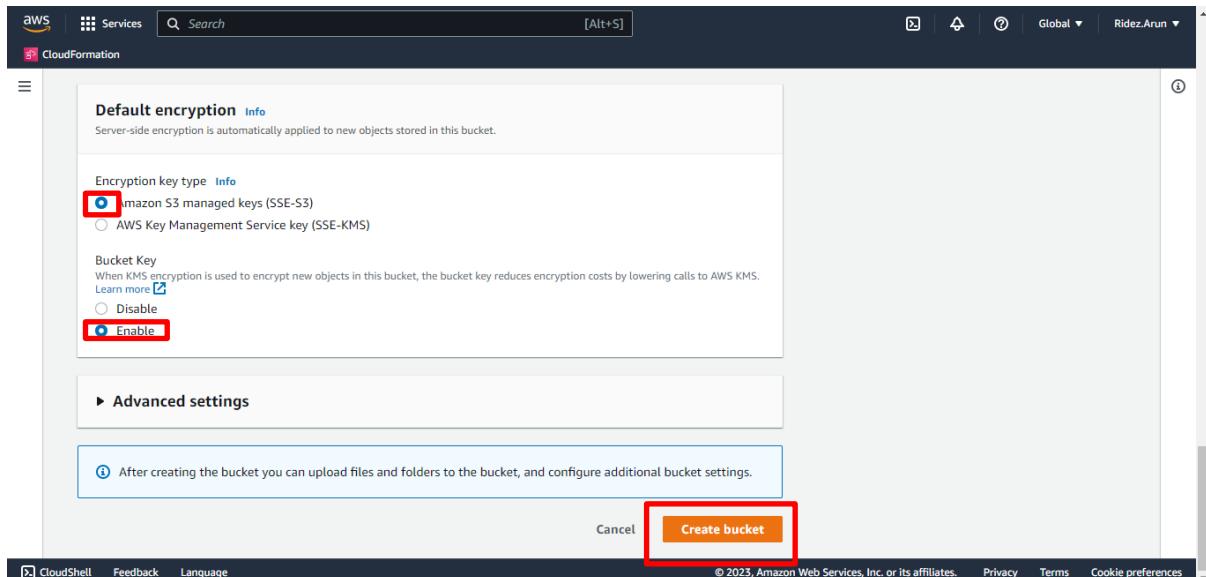
[Add tag](#)

Default encryption [Info](#)

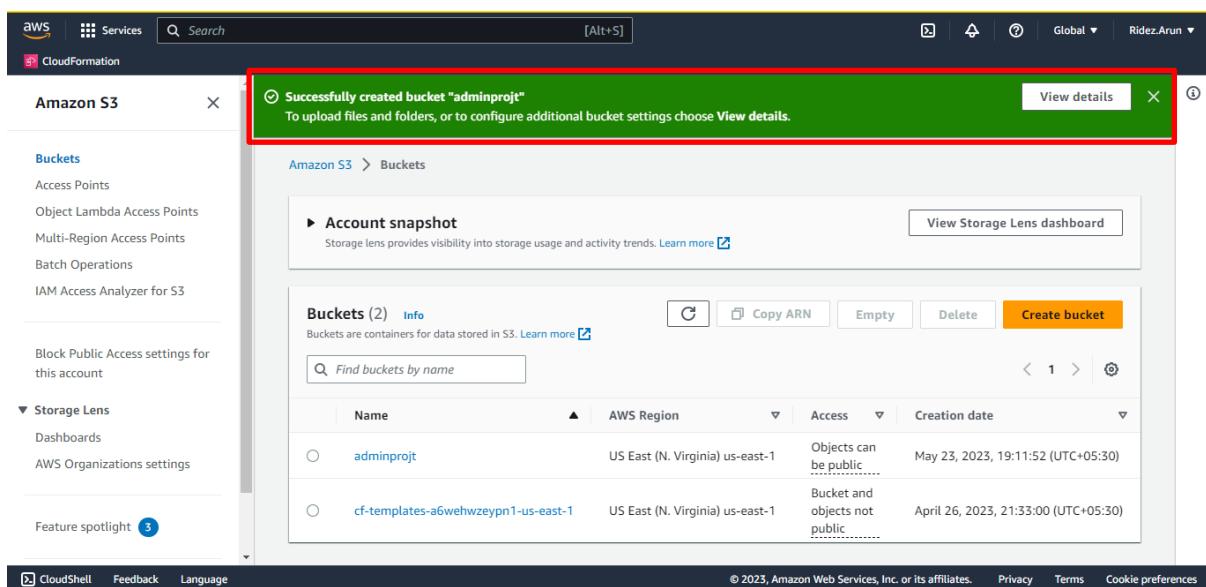
https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

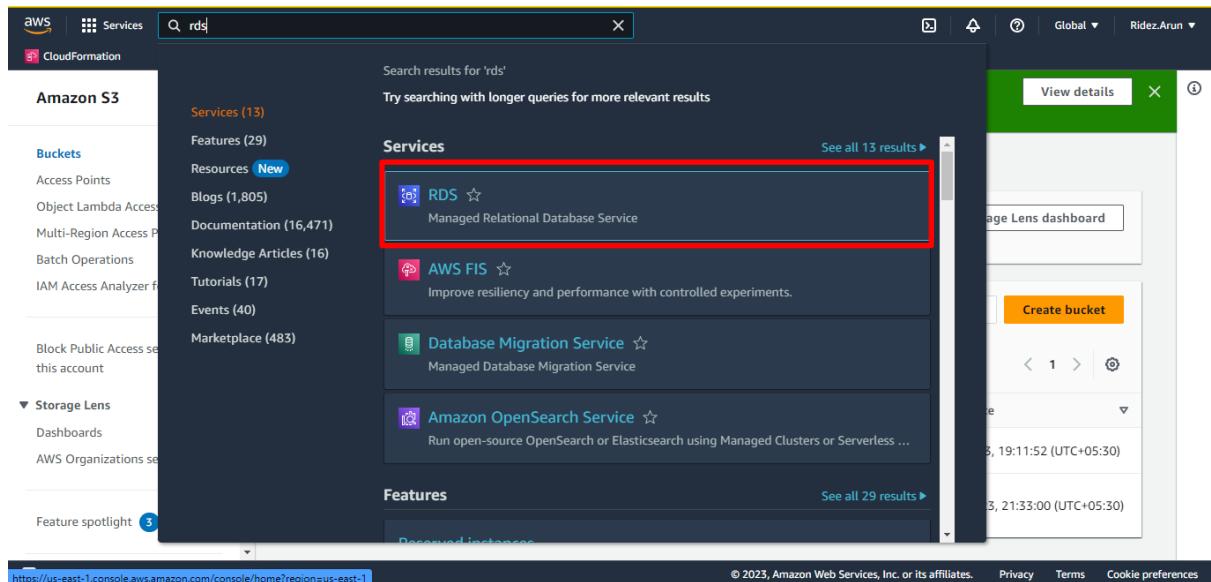
52) Choose amazon s3 → enable bucket key → create bucket.



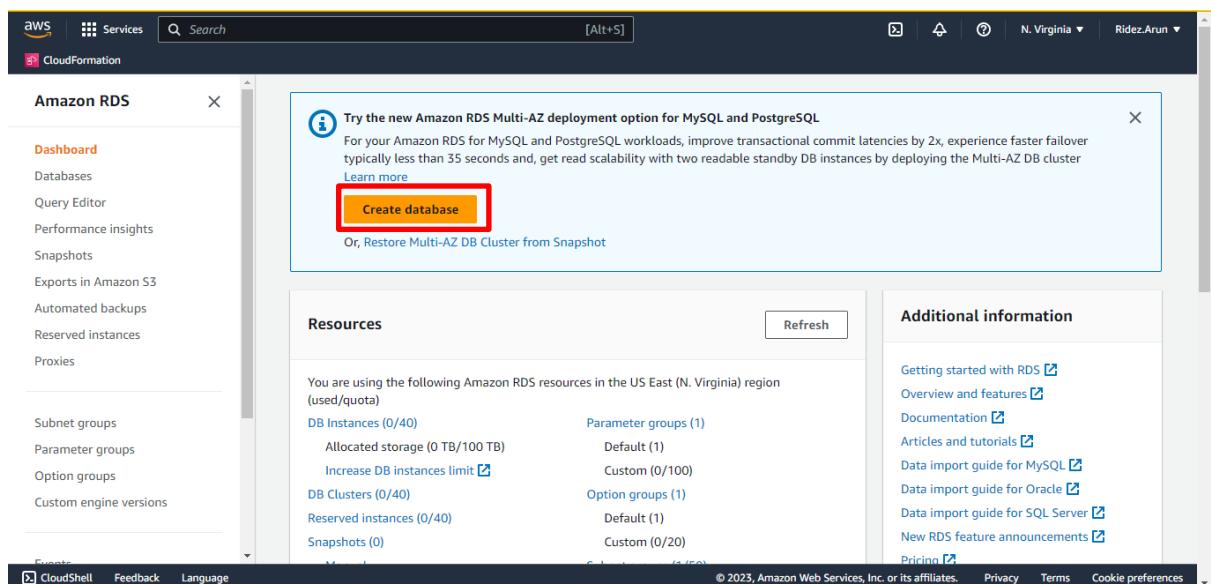
53) Bucket has been created successfully.



54) Now search for rds.



55) Click create database.



56) Choose standard create → choose mysql.

The screenshot shows the 'Create database' wizard in the AWS RDS console. In the 'Choose a database creation method' step, the 'Standard create' option is selected and highlighted with a red box. In the 'Engine options' step, the 'MySQL' engine type is selected and highlighted with a red box.

57) Choose free tier.

The screenshot shows the 'Create database' wizard in the AWS RDS console. In the 'Templates' step, the 'Free tier' option is selected and highlighted with a red box. Below it, under 'Availability and durability', the 'Multi-AZ DB Cluster - new' deployment option is selected.

58) Give name to identifier→give user name(rg.admin).

DB instance identifier [Info](#)
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username [Info](#)
Type a login ID for the master user of your DB instance.
1 to 16 alphanumeric characters. First character must be a letter.

Manage master credentials in AWS Secrets Manager
Manage master user credentials in Secrets Manager. RDS can generate a password for you and manage it throughout its lifecycle.

ⓘ If you manage the master user credentials in Secrets Manager, some RDS features aren't supported. [Learn more](#)

Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password.

© 2023, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

59) Give password(eg.admin123).

Auto generate a password
Amazon RDS can generate a password for you, or you can specify your own password.

Master password [Info](#)
Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), " (double quote) and @ (at sign).

Confirm master password [Info](#)

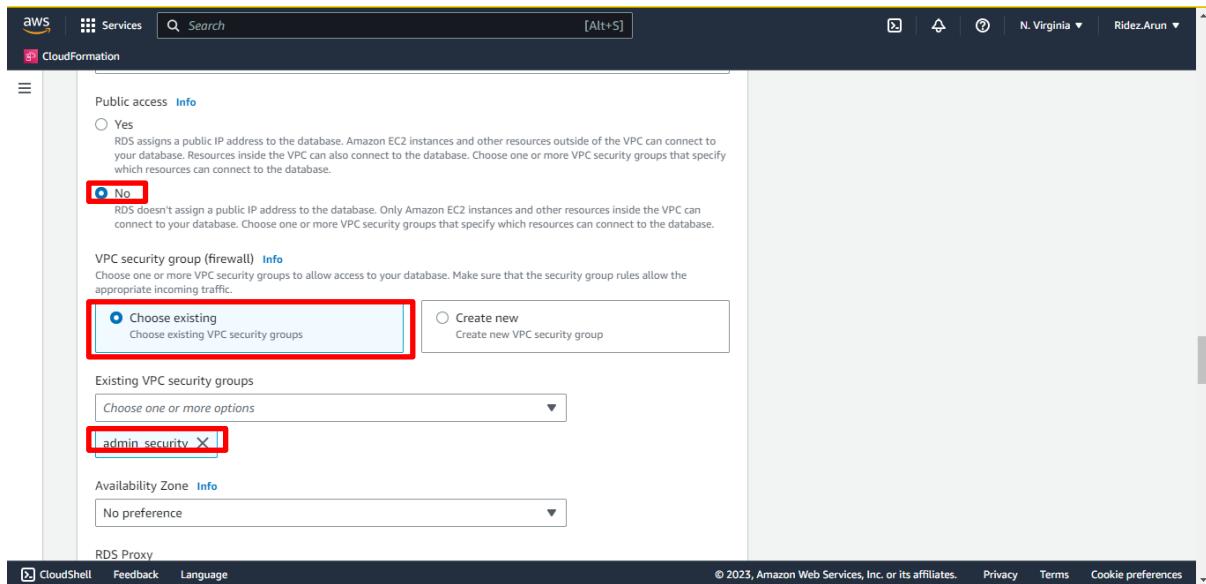
Instance configuration
The DB instance configuration options below are limited to those supported by the engine that you selected above.

Amazon RDS Optimized Writes - new [Info](#)
 Show instance classes that support Amazon RDS Optimized Writes

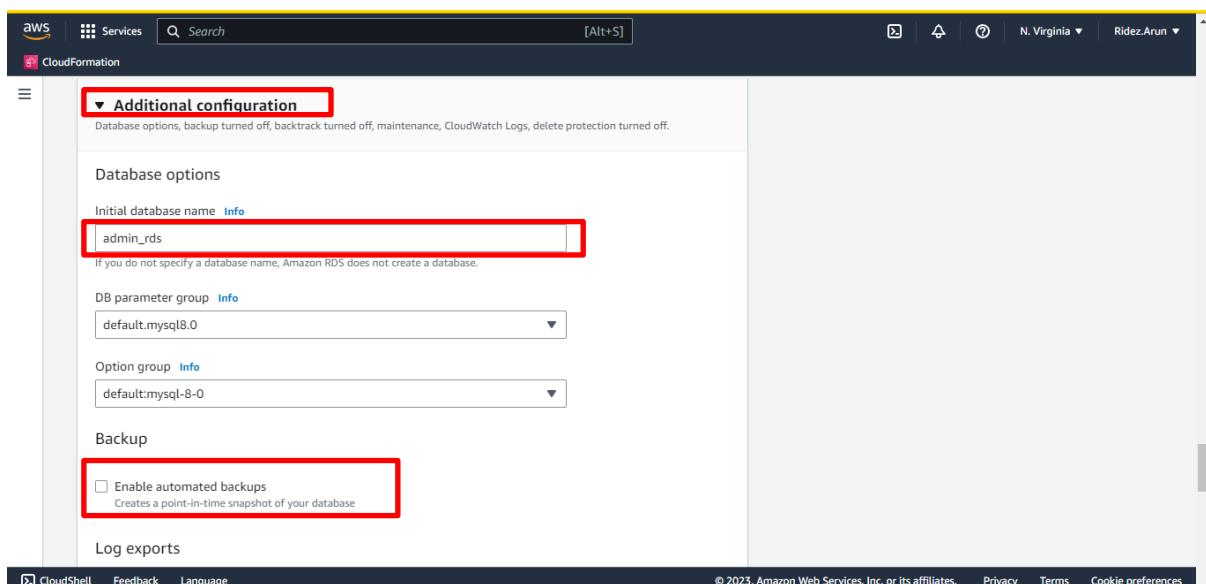
DB instance class [Info](#)
 Standard classes (includes m classes)
 Memory optimized classes (includes r and x classes)

<https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1> © 2023, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

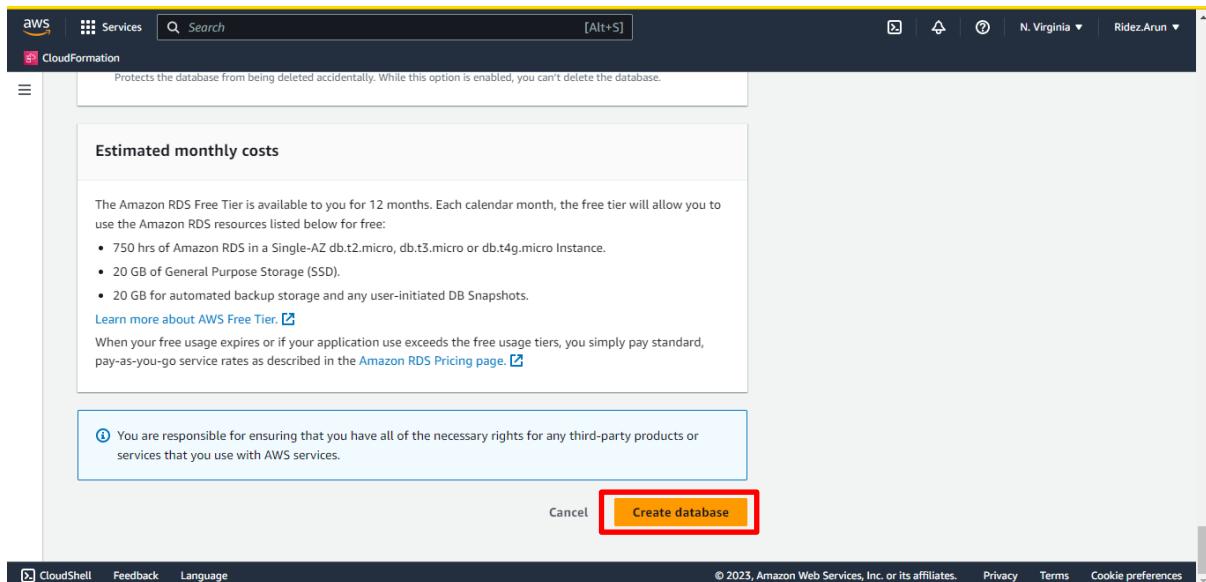
60) Select public access as no → add security group we created.



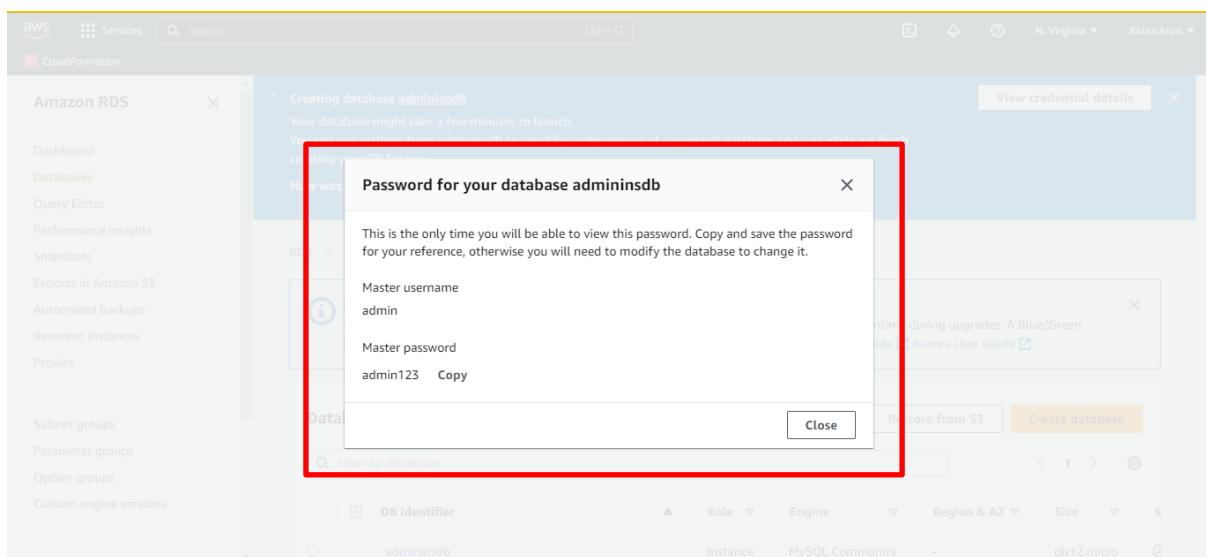
61) Click additional configuration → give database name(rg.admin_rds) → disable backup.



62) Click create database.



63) click view details → copy the details and paste it in note pad → close.



64) Use chatgpt for about us code → type generate about us html code, it will generate, once completed → copy code.

The screenshot shows a web browser window for chat.openai.com. On the left, there's a sidebar with a '+ New chat' button and a list of recent conversations: 'About Us HTML Code', 'Django eCommerce Solution.', 'Model assists user.', 'E-Commerce Application Bene...', and 'CMGFP Career Goals Fit.'. Below this is an 'Upgrade to Plus' button and a user profile 'GAYATHRI K'. The main area contains a message from the AI: 'Sure! Here's an example of an HTML code for an "About Us" page:' followed by a red-bordered code block. The code is as follows:

```
<!DOCTYPE html>
<html>
<head>
    <title>About Us</title>
</head>
<body>
    <header>
        <h1>About Us</h1>
    </header>

    <nav>
        <!-- Navigation links go here -->
    </nav>

    <section>
        <h2>Our Story</h2>
        <p>Lorem ipsum dolor si...</p>
    </section>

```

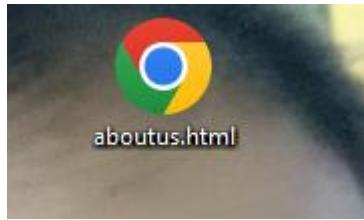
65) Paste the copied code in note pad and save it in desktop by giving name as aboutus.html.

The screenshot shows a Windows Notepad window titled '*aboutus.txt - Notepad'. The window contains the same HTML code as the AI response, with a red border around the code block. The code defines an 'About Us' page with a header, navigation, and a section about the company's story. It also includes a footer with a copyright notice for 2023.

```
<ul>
    <li>John Doe - CEO</li>
    <li>Jane Smith - CTO</li>
    <li>Mike Johnson - CFO</li>
</ul>
</section>

<section>
    <h2>Company Values</h2>
    <ul>
        <li>Integrity</li>
        <li>Innovation</li>
        <li>Customer Focus</li>
        <li>Teamwork</li>
    </ul>
</section>

<footer>
    <p>&copy; 2023 Your Company. All rights reserved.</p>
</footer>
</body>
</html>
```



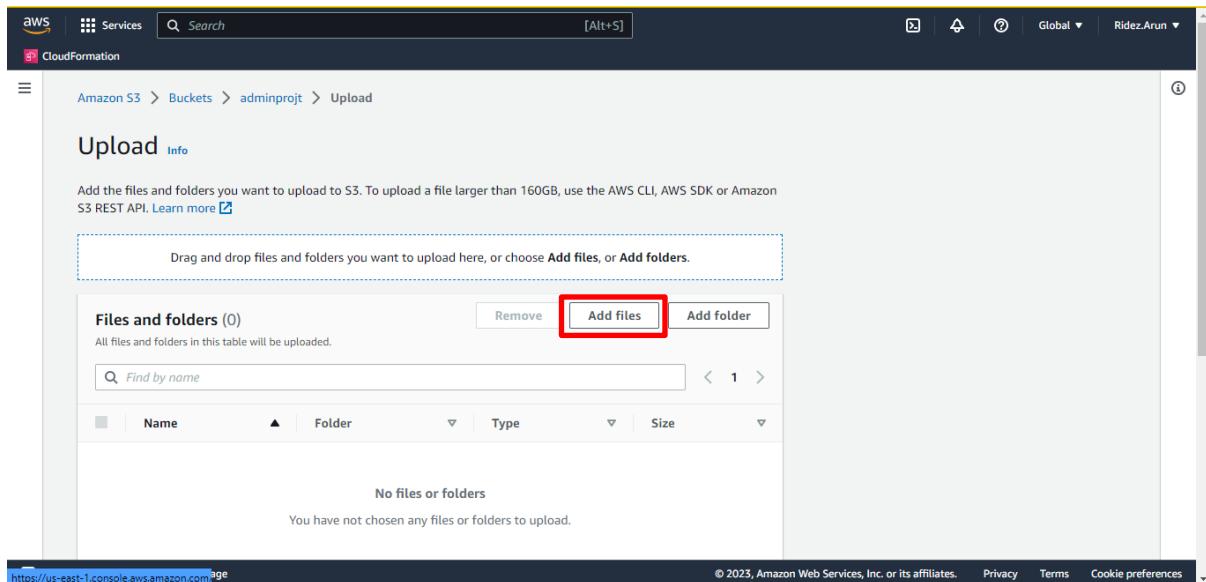
66) Go to s3 bucket → double tap the bucket we created.

A screenshot of the AWS S3 Buckets page. The left sidebar shows 'Buckets' and other options like 'Storage Lens'. The main area shows an 'Account snapshot' and a table of buckets. The 'adminproj' bucket is selected and highlighted with a red box. The table data is as follows:

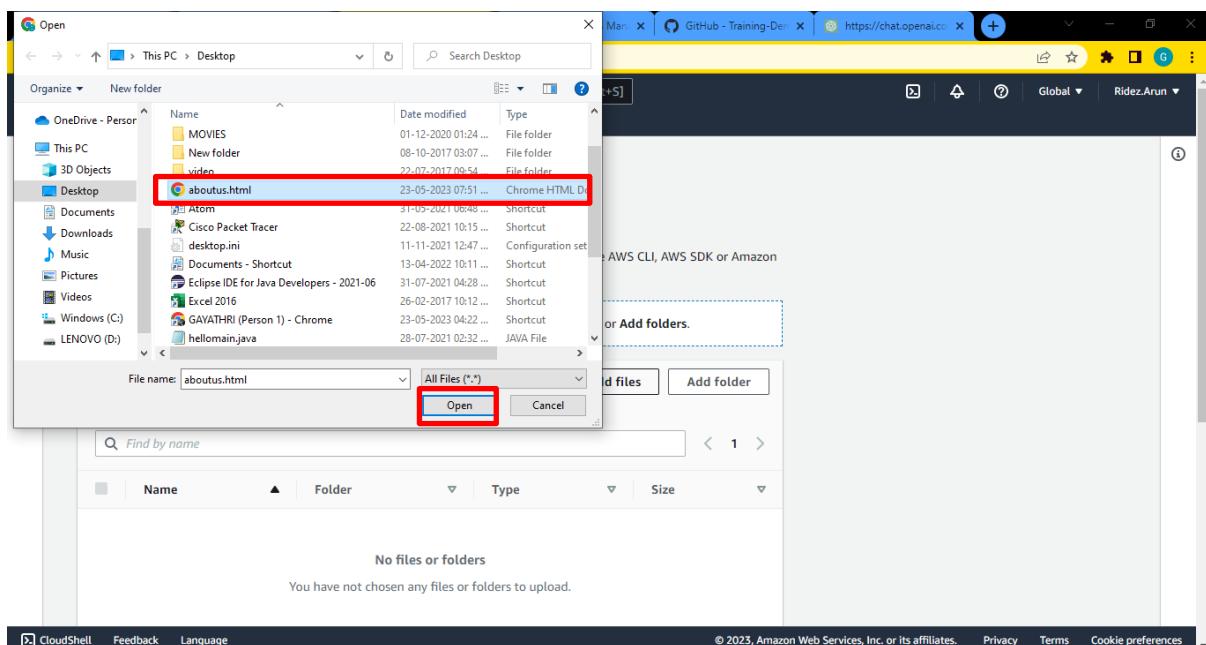
67) Click upload.

A screenshot of the AWS S3 Bucket 'adminproj' page. The left sidebar shows 'Objects' and other tabs like 'Properties', 'Permissions', 'Metrics', 'Management', and 'Access Points'. The main area shows an 'Objects (0)' section with a 'Create folder' and 'Upload' button. The 'Upload' button is highlighted with a red box. The table at the bottom shows 'No objects'.

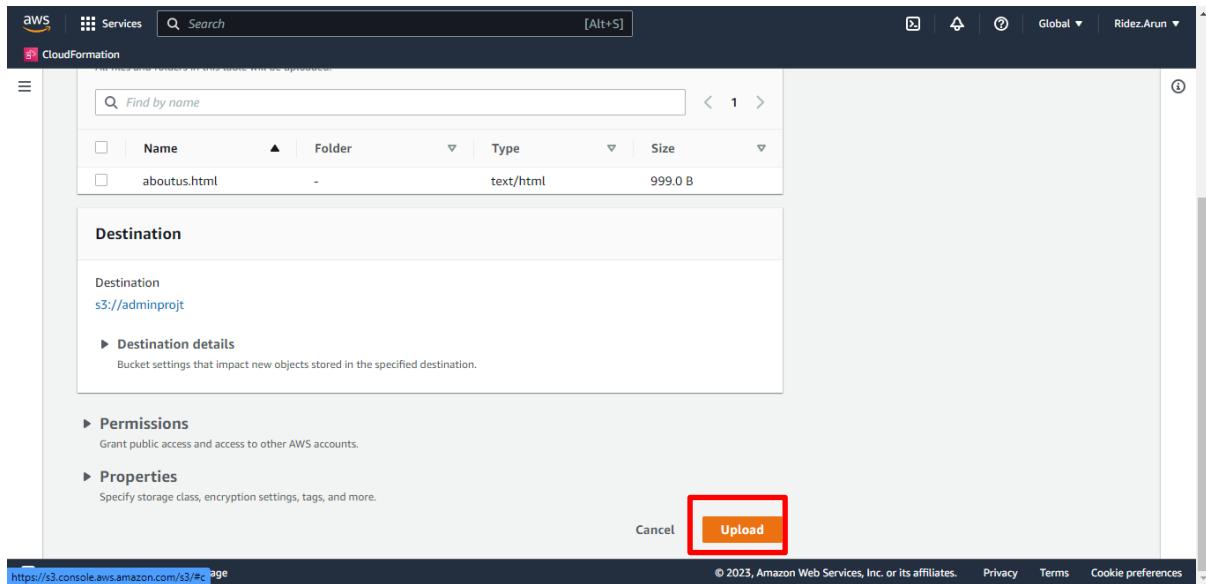
68) Click add files.



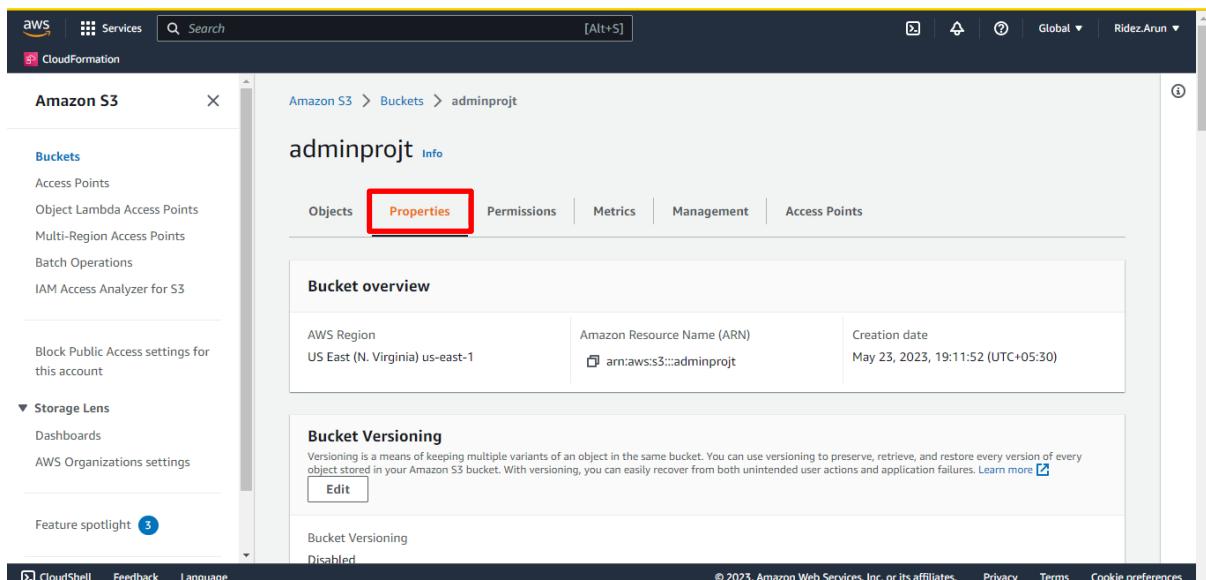
69) Choose the aboutus.html file → click open.



70) click upload.



71) In s3 bucket go to properties.



72) Go to static website hosting → click edit.

The screenshot shows the AWS S3 Bucket Properties page for a bucket named 'adminproj'. The left sidebar includes options like Buckets, Storage Lens, and Feature spotlight. The main content area displays settings for Object Lock (disabled), Requester pays (disabled), and Static website hosting. The 'Static website hosting' section is highlighted with a red box, and the 'Edit' button is also highlighted with a red box.

73) Enable static website hosting → choose host a static website.

The screenshot shows the 'Edit static website hosting' configuration page for the 'adminproj' bucket. It features sections for 'Static website hosting' (with 'Enable' selected) and 'Hosting type' (with 'Host a static website' selected). A note at the bottom 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.' The 'Enable' radio button and the 'Host a static website' radio button are both highlighted with red boxes.

74) Then in index and error document paste aboutus.html file name.

The screenshot shows the AWS S3 console with the 'Amazon S3' service selected. On the left, there's a sidebar with options like 'Buckets', 'Storage Lens', and 'Feature spotlight'. The main area is titled 'Using Amazon S3 Block Public Access' and contains sections for 'Index document' (containing 'aboutus.html') and 'Error document - optional' (also containing 'aboutus.html'). Below these are 'Redirection rules - optional' and a JSON editor. At the bottom right of the main area, there are 'Cancel' and 'Save changes' buttons, with 'Save changes' being highlighted by a red box.

75) Click save changes.

This screenshot shows the same AWS S3 console interface as the previous one, but the main content area is now focused on 'Redirection rules - optional'. It displays a single rule entry '1'. At the bottom right of this area, there are 'Cancel' and 'Save changes' buttons, with 'Save changes' being highlighted by a red box.

76) Now open policy generator in Google→type aws policy generator.

77) Go back to s3 bucket → open our s3 bucket → copy the arn.

78) Get back to policy generator → type as s3 → effec-allow → principal as * → select all actions → paste the copied ARN here → click add statement.

AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#).

Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an [IAM Policy](#), an [S3 Bucket Policy](#), an [SNS Topic Policy](#), a [VPC Endpoint Policy](#), and an [SQS Queue Policy](#).

Select Type of Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See a [description of elements that you can use in statements](#).

Effect Allow Deny

Principal

Use a comma to separate multiple values.

AWS Service All Services ("*")

Use multiple statements to add permissions for more than one service.

Actions All Actions ("*")

Amazon Resource Name (ARN)

ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${KeyName}.

Use a comma to separate multiple values.

Add Conditions (Optional)

79) Click generate policy.

Actions All Actions ("*")

Amazon Resource Name (ARN)

ARN should follow the following format: arn:aws:s3:::\${BucketName}/\${KeyName}.

Add Conditions (Optional)

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
*	Allow	s3::*	arn:aws:s3:::adminprojt	None

Step 3: Generate Policy

A *policy* is a document (written in the [Access Policy Language](#)) that acts as a container for one or more statements.

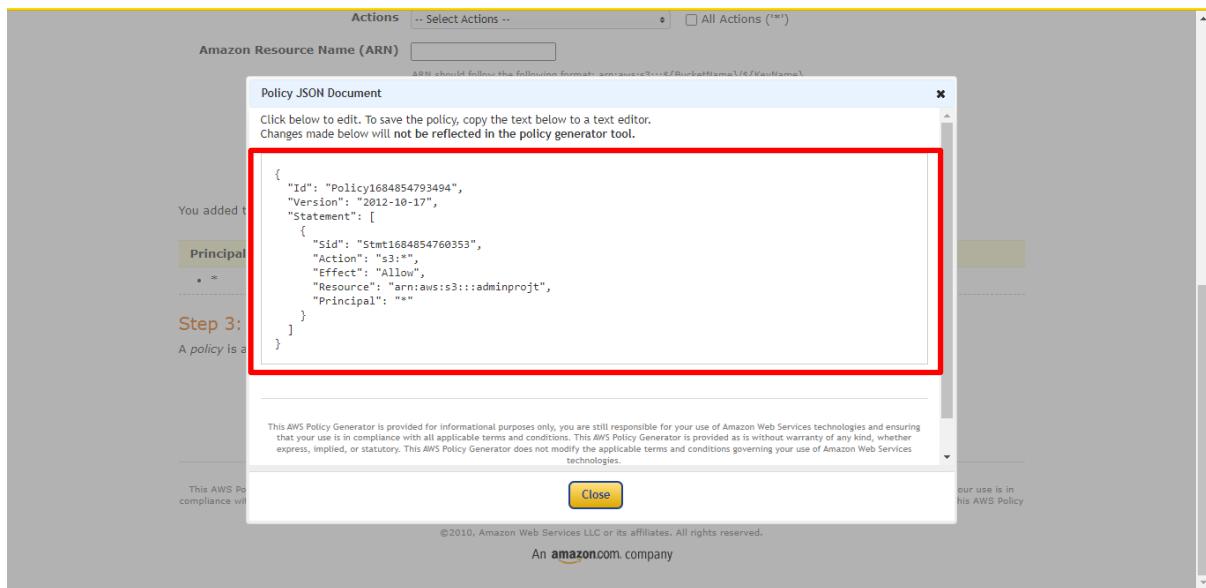
[Start Over](#)

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that your use is in compliance with all applicable terms and conditions. This AWS Policy Generator is provided [as is](#) without warranty of any kind, whether express, implied, or statutory. This AWS Policy Generator does not modify the applicable terms and conditions governing your use of Amazon Web Services technologies.

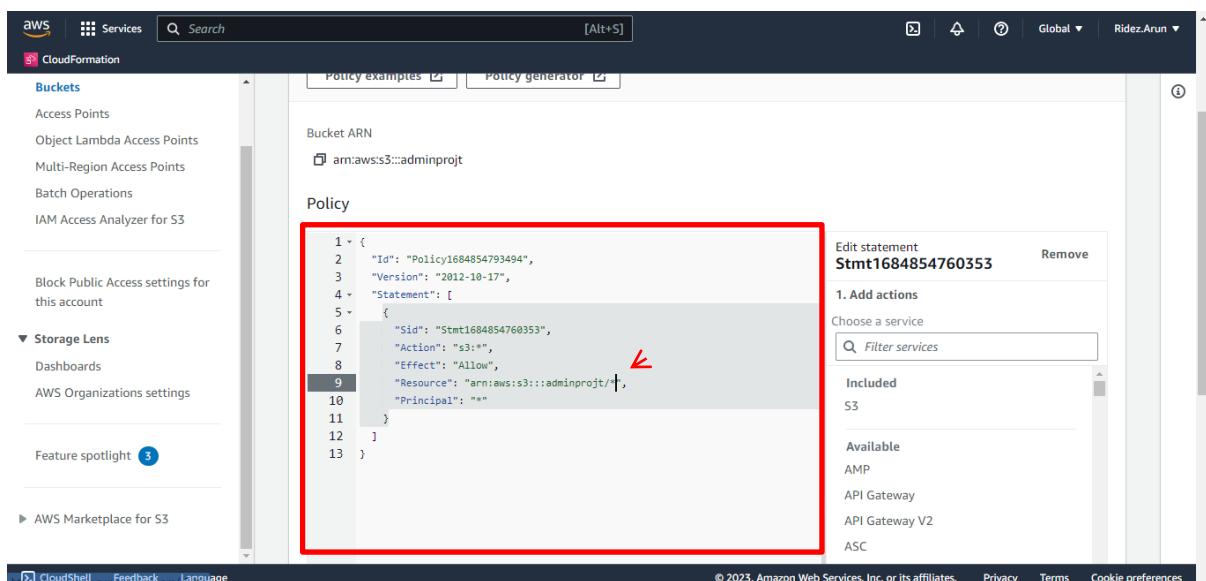
©2010, Amazon Web Services LLC or its affiliates. All rights reserved.

An [amazon.com](#) company

80) copy the generated policy.



81) Go back to our s3 bucket → permissions → edit bucket permissions → paste the copied code over there → type /* at the end of resources → click save.



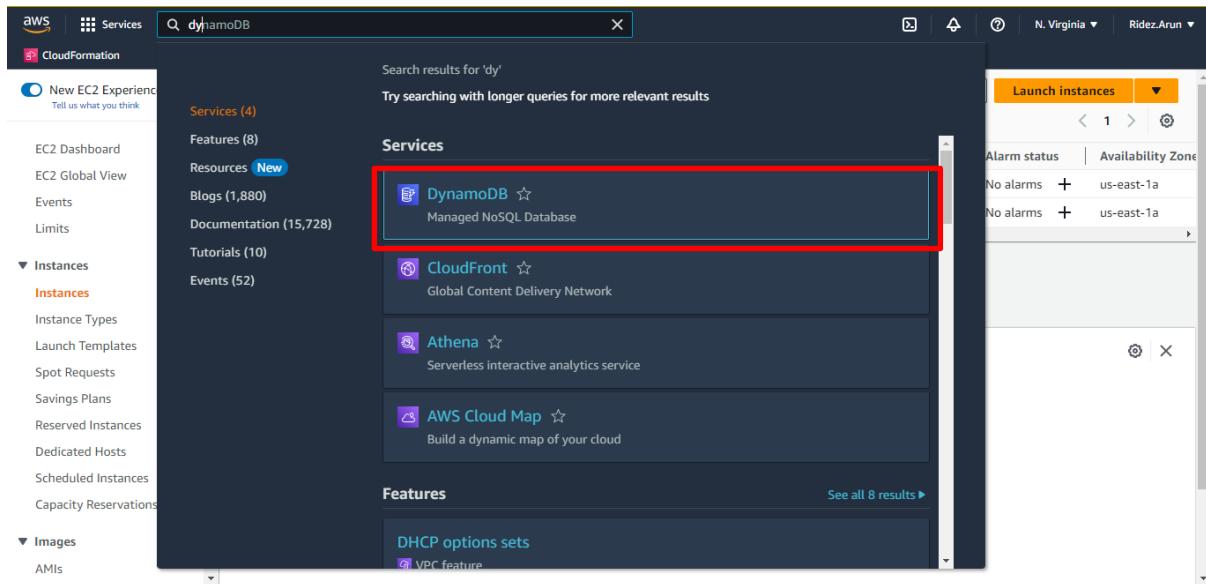
82) Go to properties → go to static website hosting → copy the endpoint paste it in browser.

The screenshot shows the AWS S3 console with the 'Static website hosting' section highlighted by a red box. Under 'Hosting type', it is set to 'Bucket hosting'. Below that, the 'Bucket website endpoint' is listed as <http://adminproj.s3-website-us-east-1.amazonaws.com>.

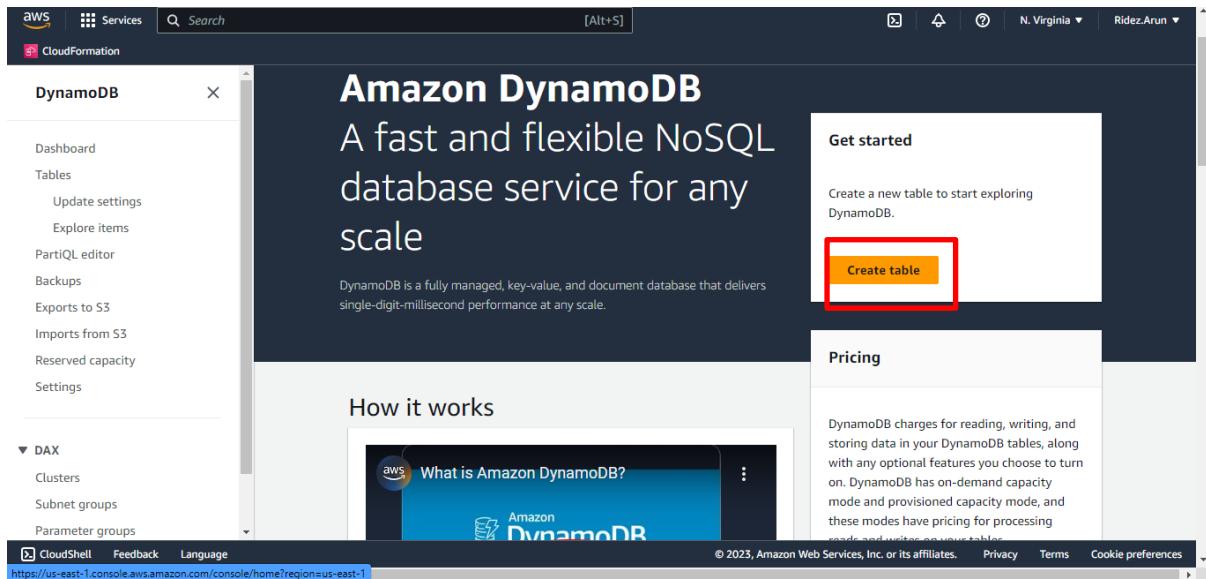
83) Once you paste the endpoint the about us page will run.

The screenshot shows a web browser displaying the 'About Us' page from the static website endpoint. The URL in the address bar is <http://adminproj.s3-website-us-east-1.amazonaws.com>. The page content includes sections for 'About Us', 'Our Story', 'Our Team', and 'Company Values'. A Google Translate overlay is visible in the top right corner.

84) Now search for dynamoDB in aws.



85) Click create table.



86) before creating a table Click this link <http://github.com/Training-Demo/AWS-Project/blob/main/EmpApp.py> open EmpApp.py file .

The screenshot shows a GitHub repository page. The URL `github.com/Training-Demo/AWS-Project` is highlighted in a red box at the top. The repository name is `Training-Demo / AWS-Project`. The repository details show 1 branch, 0 tags, and 4 commits. The code editor displays the `EmpApp.py` file, which contains Python code for interacting with S3 and DynamoDB. A red box highlights the line where the table name is specified.

87) Scroll the code here you will find table name copy the table name ‘employee_image_table’ and ‘empid’.

The screenshot shows the `EmpApp.py` file in the GitHub code editor. The line of code highlighted by a red box is:

```
86     TableName='employee_image_table',
```

A red arrow points from the text 'Table name' to this line. Another red arrow points from the text 'Partition key' to the `'empid':` part of the code.

88) Paste ‘employee_image_table’ in table name and ‘empid’ in partition key.

Screenshot of the AWS CloudFormation console showing the creation of a new DynamoDB table named "employee_image_table". The table has a partition key "empid" (Number) and no sort key.

Table details

Table name: employee_image_table

Partition key: empid (Number)

Sort key - optional: (String)

<https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1>

89) Click create table.

Screenshot of the AWS CloudFormation console showing the configuration of a new DynamoDB table. The table has the following settings:

Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Table class	DynamoDB Standard	Yes
Deletion protection	Off	Yes

Tags

No tags are associated with the resource.

Create table

<https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1>

90) The table has been created successfully.

The screenshot shows the AWS DynamoDB service page. On the left, there's a navigation sidebar with options like Dashboard, Tables, Update settings, Explore items, PartiQL editor, Backups, Exports to S3, Imports from S3, Reserved capacity, and Settings. Below that is a section for DAX with Clusters, Subnet groups, and Parameter groups. At the bottom of the sidebar is a CloudShell link. The main content area is titled 'DynamoDB > Tables'. It shows a table with one item, 'Tables (1) Info'. The table has columns: Name, Status, Partition key, Sort key, Indexes, Deletion protection, and Read capacity mode. The single row is highlighted with a red box. The row details are: Name is 'employee_image_table', Status is 'Active', Partition key is 'empid (N)', Sort key is null, Indexes are off, Deletion protection is off, and Read capacity mode is 'Provisioned with auto scaling'. At the top right of the table view are buttons for Actions, Delete, and Create table.

91) Now we need to create a load balancer → click load balancer in left panel.

The screenshot shows the AWS Lambda service page. On the left, there's a navigation sidebar with sections for Elastic Block Store (Volumes, Snapshots, Lifecycle Manager), Network & Security (Security Groups, Elastic IPs, Placement Groups, Key Pairs, Network Interfaces), Load Balancing (Load Balancers, Target Groups), and Auto Scaling (Launch Configurations, Auto Scaling Groups). The 'Load Balancers' option is highlighted with a red box. The main content area is titled 'Instances (2) Info'. It shows a table with two items. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, and Availability Zone. The two rows are: 'application_se...' with Instance ID 'i-031b0147a7734faab', Instance state 'Running', Instance type 't2.micro', Status check 'Initializing', Alarm status 'No alarms', and Availability Zone 'us-east-1a'; and 'Bastion_host' with Instance ID 'i-0e90a251daab69d2d', Instance state 'Running', Instance type 't2.micro', Status check '2/2 checks passed', Alarm status 'No alarms', and Availability Zone 'us-east-1a'. At the top right of the table view are buttons for Connect, Actions, and Launch instances.

92) Click create load balancer.

The screenshot shows the AWS CloudFormation interface for managing load balancers. In the top navigation bar, there's a link to 'New EC2 Experience' with a feedback invitation. The main content area is titled 'Load balancers' with a sub-instruction: 'Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.' Below this is a search bar and a table header with columns for Name, DNS name, State, VPC ID, and Availability Zones. A prominent orange button labeled 'Create load balancer' is highlighted with a red box. The table below shows '0 load balancers selected' and a placeholder message: 'Select a load balancer above.'

93) Choose application load balancer.

This screenshot displays the 'Load balancer types' section of the AWS documentation. It compares three types of load balancers:

- Application Load Balancer**: Represented by a diagram showing a user agent connecting to an ALB (Application Load Balancer) via HTTP and HTTPS. The ALB then routes traffic to three targets (Lambda, API Gateway, and EC2 instances). A red box highlights this section.
- Network Load Balancer**: Represented by a diagram showing a user agent connecting to an NLB (Network Load Balancer) via TCP, UDP, or TLS. The NLB then routes traffic to three targets (ALB, Lambda, and EC2 instances).
- Gateway Load Balancer**: Represented by a diagram showing a user agent connecting to a GWLB (Gateway Load Balancer) via TCP, UDP, or TLS. The GWLB then routes traffic to three targets (AWS WAF, AWS Lambda, and AWS CloudFront).

Below each diagram is a descriptive text block:

- Application Load Balancer**: 'Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic.'
- Network Load Balancer**: 'Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, and support for multiple protocols.'
- Gateway Load Balancer**: 'Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE.'

94) Give name to load balancer → choose internet facing → ipv4.

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

Scheme [Info](#)
 [internet-facing](#) An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)
 Internal An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)
Select the type of IP addresses that your subnets use.
 [IPv4](#) Recommended for internal load balancers.
 [Dualstack](#) Includes IPv4 and IPv6 addresses.

95) Choose the vpc we created.

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 you've created the load balancer. If you change the VPC for your targets, then your targets might stop receiving traffic. [Create new VPC](#)

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.

96) select both availability zone with public subnet 1 and 2.

The screenshot shows the AWS CloudFormation console with two EC2 instances listed:

- us-east-1a (use1-az4)**: Subnet: subnet-09f5835a168a83382, Security group: admin_public1
- us-east-1b (use1-az6)**: Subnet: subnet-08d80e5bb57fa15ac, Security group: admin_public2

97) select the security group we created → click create target group.

The screenshot shows the AWS CloudFormation console with a security group selected:

Security groups
Select up to 5 security groups
admin_security sg-0ebf9e03b34c832a4 X
VPC: vpc-084f8a5472556f25d

Listeners and routing
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: HTTP Port: 80 Default action: Select a target group
Forward to: Select a target group (highlighted by a red box)
Create target group (highlighted by a red box)

98) choose instance in target group.

Step 1
Specify group details

Step 2
Register targets

Specify group details

Basic configuration

Choose a target type

Instances

- Supports load balancing to instances within a specific VPC.
- 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.

IP addresses

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

99) Give name to target group → choose our vpc → click next.

admin_target

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

Protocol Port

HTTP : 80

1-65535

VPC

Select the VPC with the instances that you want to include in the target group.

project_vpc

vpc-084f8a5472536f25d

IPv4: 20.20.0.0/16

Protocol version

HTTP1

Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

HTTP2

Send requests to targets using HTTP/2. Supported when the request protocol is HTTP/2 or gRPC, but gRPC-specific features are not available.

gRPC

Send requests to targets using gRPC. Supported when the request protocol is gRPC.

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

100) register the application server to the target group.

The screenshot shows the 'Register targets' step of a CloudFormation stack creation. A table lists available instances:

Instance ID	Name	Status	Security groups
i-031b0147a7734faab	application_server	Running	admin_security
i-0e90a251daab69d2d	Bastion_host	Running	admin_security

A red box highlights the first instance, 'application_server'. Below the table, it says '1 selected'. A port range '80' is entered in a field labeled 'Ports for the selected instances'. A button 'Include as pending below' is visible.

101) The target group has been created successfully.

The screenshot shows the EC2 Target groups page. A table displays one target group:

Name	ARN	Port	Protocol	Target type
admintarget	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/admintarget/54321	80	HTTP	Instance

A red box highlights the 'admintarget' row. Below the table, it says '0 target groups selected'.

102) Go back to the load balancer → refresh and add the target group we created.

Listeners and routing Info
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 1-65535	Forward to	admintarget Target type: Instance, IPv4

Create target group Info

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

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

103) Once the target group is added → click create load balancer.

Basic configuration Edit
adminelb

- Internet-facing
- IPv4

Security groups Edit
• admin_security
sg-0ebf9e03b34c832a4 Info

Network mapping Edit
VPC vpc-084f8a5472536f25d Info
project_vpc

- us-east-1
 - subnet-09f5835a168a83382 Info
 - admin_public1
- us-east-1b
 - subnet-08d80e5bb57fa15ac Info
 - admin_public2

Listeners and routing Edit
• HTTP:80 defaults to
admintarget Info

Add-on services Edit
None

Tags Edit
None

Attributes

Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.

Cancel Create load balancer

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

104) Go to instances.

The screenshot shows the AWS CloudFormation Instances page. There are two instances listed:

- application_se...** (i-031b0147a7734faab): Running, t2.micro, 2/2 checks passed, No alarms, us-east-1a
- Bastion_host** (i-0e90a251daab69d2d): Running, t2.micro, 2/2 checks passed, No alarms, us-east-1a

A modal window titled "Select an instance" is open, showing the same two instances. The "Connect" button in the modal is highlighted with a red box.

105) Select the public instance(bastion_host) → click connect.

The screenshot shows the AWS CloudFormation Instances page with the Bastion_host instance selected. The "Connect" button is highlighted with a red box. Below the table, the instance details are shown:

Instance: i-0e90a251daab69d2d (Bastion_host)

Details			Security	Networking	Storage	Status checks	Monitoring	Tags
Instance summary <div style="display: flex; justify-content: space-between;"> <div>Instance ID i-0e90a251daab69d2d (Bastion_host)</div> <div>Public IPv4 address 54.86.183.206 open address</div> <div>Private IPv4 addresses 20.20.1.116</div> </div> <div>IPv6 address -</div> <div>Hostname type Ubuntu Server 16.04 LTS</div> <div>Instance state Running</div> <div>Private IP DNS name (IPv4 only) 54.86.183.206</div> <div>Public IPv4 DNS -</div>								

106) open windows power shell and connect the public instance(bastion_host)using command

Directory → cd downloads

Connect command → ssh -I "keypairname.pem" ubuntu@public ip of instance

Update command → sudo apt-get update -y

Install command → sudo apt-get install mysql-client -y

New keypairfile command → nano keyfile.pem

```
ubuntu@ip-20-20-3-215: ~
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\dhanesh> cd .\Downloads\←
PS C:\Users\dhanesh\Downloads> ssh -i "keypair.pem" ubuntu@54.86.183.206 ←
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

 * Documentation: https://help.ubuntu.com
 Welcom to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Tue May 23 17:03:25 UTC 2023

 System load: 0.0 Processes: 103
 Usage of /: 23.9% of 7.57GB Users logged in: 1
 Memory usage: 28% IPv4 address for eth0: 20.20.1.116
 Swap usage: 0%

 * Ubuntu Pro delivers the most comprehensive open source security and
 compliance features.

 https://ubuntu.com/aws/pro

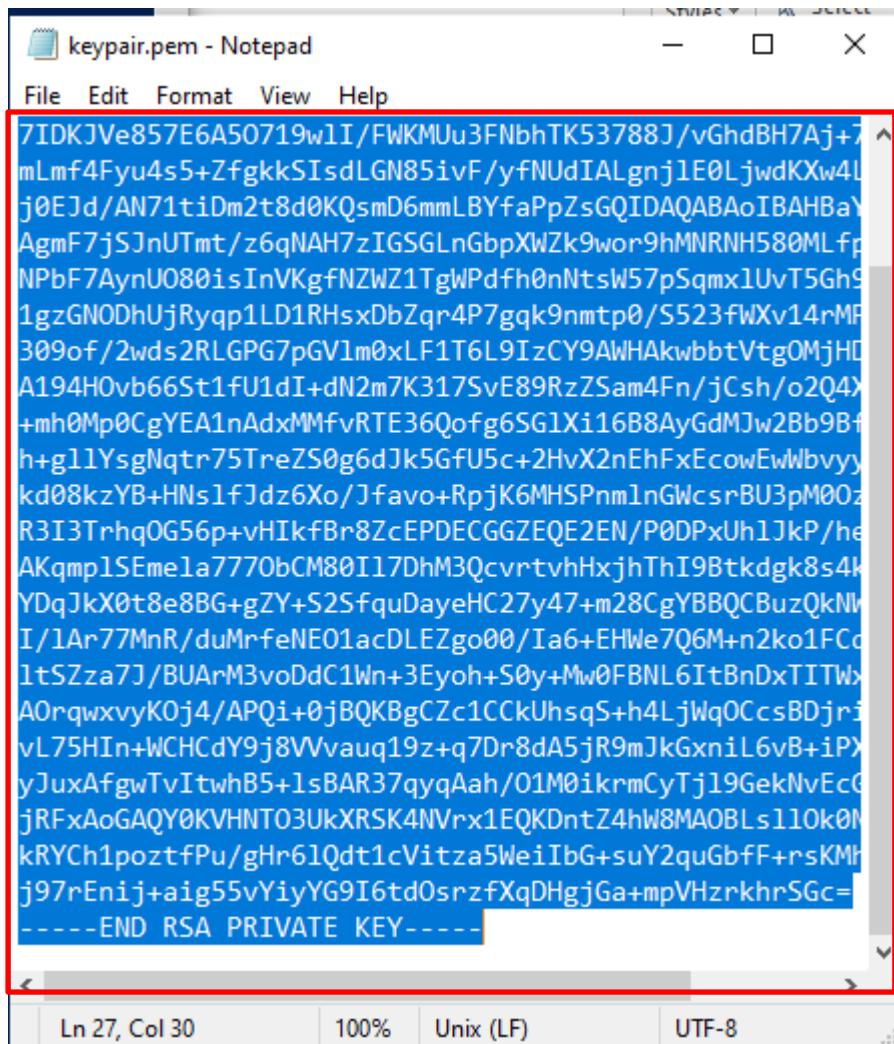
Expanded Security Maintenance for Applications is not enabled.

7 updates can be applied immediately.
6 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Tue May 23 16:58:20 2023 from 223.182.247.95
ubuntu@ip-20-20-1-116:~$ sudo apt-get install mysql-client -y ←
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mysql-client is already the newest version (8.0.33-0ubuntu0.22.04.2).
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
ubuntu@ip-20-20-1-116:~$ sudo su ←
root@ip-20-20-1-116:/home/ubuntu# ssh -i "keypair.pem" ubuntu@20.20.3.215
Warning: Identity file keypair.pem not accessible: No such file or directory.
```

107) open the keypair file in our file manager or in downloads using notepad → copy the text inside the keypair.



The screenshot shows a Windows Notepad window titled "keypair.pem - Notepad". The window contains a large amount of base64-encoded RSA private key data. A red box highlights the entire content of the text area. At the bottom of the text area, there is a line of dashes followed by the text "-----END RSA PRIVATE KEY-----". The status bar at the bottom of the Notepad window displays "Ln 27, Col 30", "100%", "Unix (LF)", and "UTF-8".

```
7IDKJVe857E6A50719w1I/FWKMUu3FNbhTK53788J/vGhdBH7Aj+7  
mLmf4Fyu4s5+ZfgkkSIsdLGN85ivF/yfNUdIALgnj1E0LjwdKXw4L  
j0EJd/AN71tiDm2t8d0KQsmD6mmLBYfaPpZsGQIDAQABoIBAHBaY  
AgmF7jSJnUTmt/z6qNAH7zIGSGLnGbpXWZk9wor9hMNRNH580MLfp  
NPbF7AynU080isInVkgfNZWZ1TgWPdfh0nNtsW57pSqmxB1UvT5Gh9  
1gzGNODhUjRyqp1LD1RHsxDbZqr4P7gqk9nmtP0/S523fWXv14rMF  
309of/2wds2RLGPG7pGV1m0xLF1T6L9IzCY9AWHAkwbbtVtg0MjHD  
A194HOvb66St1FU1dI+dN2m7K317SvE89RzZSam4Fn/jCsh/o2Q4X  
+mh0Mp0CgYEAnAdxMMFvRTE36Qofg6SG1Xi16B8AyGdMJw2Bb9Bf  
h+g11YsgNqtr75TreZS0g6dJk5GfU5c+2HvX2nEhFxEcwEwWbvy  
kd08kzYB+HNs1fJdz6Xo/Jfavo+RpjK6MHSPnmlnGWcsrBU3pM0Oz  
R3I3TrhqOG56p+vHIkfBr8ZcEPDECGGZEQE2EN/P0DPxUh1JkP/he  
AKqmp1SEmela7770bCM80I17DhM3QcvrtvhHxjhThI9Btkdgk8s4k  
YDqJkX0t8e8BG+gZY+S2SfquDayeHC27y47+m28CgYBBQCBuzQkNw  
I/1Ar77MnR/duMr-feNE01acDLEZgo0/Ia6+EHWe7Q6M+n2ko1FCd  
1tSZza7J/BUArM3voDdC1Wn+3Eyoh+S0y+Mw0FBNL6ItBnDxTITWx  
A0rqwxvyK0j4/APQi+0jBQKBgCZc1CCkUhsqS+h4LjWqOCcsBDjri  
vL75HIn+WCHCdY9j8Wvauq19z+q7Dr8dA5jR9mJkGxniL6vB+iPX  
yJuxAfgwTvItwhB5+lsBAR37qyqAah/01M0ikrmCyTj19GekNvEc  
jRFxAoGAQY0KVHNT03UkXRSK4NVrx1EQKDntZ4hW8MA0BLs1l0k0N  
kRYCh1poztfPu/gHr61Qdt1cVitz5WeiIbG+suY2quGbFF+rskMh  
j97rEnij+aig55vYiyYG9I6td0srzfXqDHgjGa+mpVHzrkhrSGc=  
-----END RSA PRIVATE KEY-----
```

108) paste the text in new keyfile → "click **ctrl+o** → **enter** → **ctrl+x**" in keyboard.

```
ubuntu@ip-20-20-3-215: ~
GNU nano 6.2                                     keyfile.pem *
-----BEGIN RSA PRIVATE KEY-----
MIIEogIBAAKCAQEAEhsrB4ECHCQ4exvp9nDDBWcZup+4HBMmroW70XRicHdJGFgpI
EbFL+6H0IPamX192LtsWJdg1rM/gdJZc+gUL3SF49hObgBKzxjnvwFmiCrjoUhC9
uiVyohFPRUcXbUsAzbLAWryIzMzh+V2aABEfM6k3S1HB80cAdBRoD+oJuDgRR1m
7IDKJve857E6A50719w1I/FWKMu3FnbhTK53788j/vghdBH7Aj+7fnbyOdbTPQz
mLmf4Fyu4s5+2fgkkSiSdLGN85ivF/yfNUdIALgnj1e0LjwdKXw4LGcPedY4L7qD
j0EJd/AN71tiDm2t8d0KQsmD6mmLBYfaPpZsGQIDAQABoIBAHBaY14sCyK2HvT
AgmF7jsJnuUTmt/z6qNAH7zIGSGlnGbpXwZk9wor9hMNRNHS80MLfpUdoMocoQvyL
NPbF7AynUO80isInVkgfNZW1TgWPdfh0nNtsW57pSqmxiUvT5Gh9qvssBTf1npH
1gzGNODhUjRyqp1LD1RHsxDbZqr4P7gqk9nmtp0/S523fNxV14rMPvQXMKn/kjs0
309of/2wds2RLGP7pGV1m0xLF1T6.9IzCY9AhHAkwbbtVtg0MjhDyyW2eUifsgJ
A194HOvba66StifU1dI+dN2m7K3175vE89rzZSam4Fn/jCsh/o2Q4XlgHyPaw1MEV
+mh0Mp0CgYEAIaDxMMfVRE360qfg6SG1X1i6B8AyGdMjw2B9BfbvUzpFJNaf0
h+g11YsgNqtr75TrzS0g6dJk5GfU5c+2HvX2nEnFxEcwEWbvyoddytT9Nk79
kd08kzYB+Hns1fJdz6Xo/Jfavo+Rpjk6MHPmmlngWcsrBU3pM00zPcCgYEaoOr0
R313TrhqQG56p+vHTkfBr8ZcEPDECgZEQE2EN/P0DPxUh1jkP/hexjhQs1D3GG4
AKomp1SEmeLa7770bCM80I17DH3QcvrtvhHxjhThI98tkdgk8s4k8hbvVvGbPyJ
YDqJkx0t8e8BG+gZY+S2sfquDayeHC27y47+m28CgYBBQCBuZQkNWhKGExeXhauF
I/1Ar77MnR/duMrfeNE01acDLEZgo0/Ia6+EHWe7Q6M+n2ko1FCd1b3Jmqq3vzq
l+szza7J/BUArM3voDdC1Wn+3Eyoh+S0y+Mw0FBNL6ItBnDxTITWx8zyEf655tmt
AOrqwxvyKoj4/APQi+0jBQKBgcZc1CCkUhsqS+h4LjWqOCcsBDjriTdgrttwqRUu
vl75HIn+wCHCdY9j8VVvaug19z+q7Dr8dA5jR9mJkGxniL6vB+iPxg7U9e9xthhA
yJuxAfgwTvItwhB5+1sBAR37qyqAh/O1M0ikrmCytj19GeKnVeEcGKYXH1LTeIR
jRFxAoGAQY0KVHNT03UKXRSK4NVrx1EQKDntZ4hW8MAOBls110k0NMuso5DQ7asg
kRYCh1pozfPu/gHr6l0dt1cVitzasWeibG+suY2quGbf+rskMhuLiiXXLaY0Y
j97rEnij+aig55vYiyYG9I6tdOsrfXqDHgjGa+mpVHzrkhrSGc=
-----END RSA PRIVATE KEY-----
```

^G Help ^O Write Out ^W Where Ts ^K Cut ^T Execute ^C Location ^M-U Undo ^M-A Set Mark

109) getting back to home of Ubuntu command → sudo su

Accessing new keypair command → chmod 400 keyfile.pem

Connecting private instance command → ssh ubuntu@private ip address of private instance(application_server is my private instance paste its ip) -i keyfile.pem

Ssh [ubuntu@20.20.3.215](ssh://ubuntu@20.20.3.215) -i keyfile.pem

```
ubuntu@ip-20-20-3-215:~  
Reading state information... Done  
mysql-client is already the newest version (8.0.33-0ubuntu0.22.04.2).  
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.  
ubuntu@ip-20-20-1-116:~$ sudo su  
root@ip-20-20-1-116:/home/ubuntu# ssh -i "keypair.pem" ubuntu@20.20.3.215  
Warning: Identity file keypair.pem not accessible: No such file or directory.  
The authenticity of host '20.20.3.215 (20.20.3.215)' can't be established.  
ED25519 key fingerprint is SHA256:95i+piKLZkNV9odhyhZCGE2ufBnzRYgR4wsRHii51k.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '20.20.3.215' (ED25519) to the list of known hosts.  
ubuntu@20.20.3.215: Permission denied (publickey).  
root@ip-20-20-1-116:/home/ubuntu# ssh ubuntu@20.20.3.215 keypair.pem  
ubuntu@20.20.3.215: Permission denied (publickey).  
root@ip-20-20-1-116:/home/ubuntu# ssh ubuntu@20.20.3.215 -i keypair.pem  
Warning: Identity file keypair.pem not accessible: No such file or directory.  
ubuntu@20.20.3.215: Permission denied (publickey).  
root@ip-20-20-1-116:/home/ubuntu# chmod 400 keypair.pem  
chmod: cannot access 'keypair.pem': No such file or directory  
root@ip-20-20-1-116:/home/ubuntu# nano keyfile.pem  
root@ip-20-20-1-116:/home/ubuntu# ls  
keyfile.pem  
root@ip-20-20-1-116:/home/ubuntu# chmod 400 keyfile.pem  
root@ip-20-20-1-116:/home/ubuntu# ssh ubuntu@20.20.3.215 -i keyfile.pem  
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)  
  
* Documentation: https://help.ubuntu.com  
* Management: https://landscape.canonical.com  
* Support: https://ubuntu.com/advantage  
  
System information as of Tue May 23 17:12:49 UTC 2023  
  
System load: 0.0 Processes: 94  
Usage of /: 20.6% of 7.57GB Users logged in: 0  
Memory usage: 23% IPv4 address for eth0: 20.20.3.215  
Swap usage: 0%  
  
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update
```

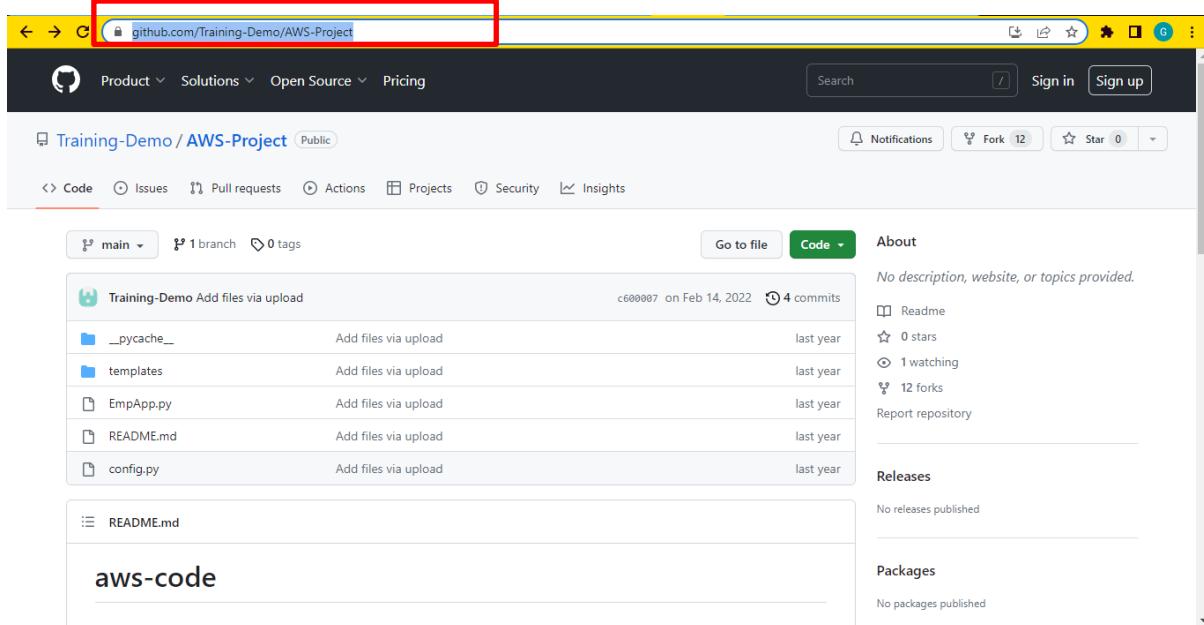
110) update private instance → sudo apt-get update -y

Install mysql → sudo apt-get install mysql-client -y

```
ubuntu@ip-20-20-3-215:~$ sudo apt-get update -y  
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease  
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]  
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]  
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]  
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]  
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]  
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]  
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]  
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]  
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]  
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [609 kB]  
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [172 kB]  
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [14.6 kB]  
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [260 kB]  
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [38.7 kB]  
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [906 kB]  
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [186 kB]  
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 c-n-f Metadata [18.9 kB]  
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [35.3 kB]  
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse Translation-en [8452 B]  
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 c-n-f Metadata [468 B]  
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 Packages [40.9 kB]  
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main Translation-en [10.2 kB]  
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/main amd64 c-n-f Metadata [388 B]  
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/restricted amd64 c-n-f Metadata [116 B]  
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [23.3 kB]  
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe Translation-en [15.0 kB]  
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/universe amd64 c-n-f Metadata [548 B]  
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports/multiverse amd64 c-n-f Metadata [116 B]  
Get:30 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [394 kB]  
Get:31 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [113 kB]
```

```
getchd 24.0 MB in 4s (5544 KB/s)
Reading package lists... Done
Ubuntu@ip-20-20-3-215:~$ sudo apt-get install mysql-client -y
```

111) Now go the git code → copy the url.



The screenshot shows a GitHub repository page. The URL 'github.com/Training-Demo/AWS-Project' is highlighted with a red box in the browser's address bar. The repository name 'Training-Demo / AWS-Project' is displayed above the code area. The code tab is selected, showing a list of files: __pycache__, templates, EmpApp.py, README.md, and config.py, all added via upload last year. The repository has 4 commits and 12 forks. The 'About' section notes 'No description, website, or topics provided.' and includes links for Readme, Stars (0), Forks (12), and Watchers (1). The 'Releases' and 'Packages' sections show no releases or packages published.

112) git clone command → git clone paste the url

git clone <https://github.com/Training-Demo/AWS-Project>

```
ubuntu@ip-20-20-3-215:~$ git clone https://github.com/Training-Demo/AWS-Project
Cloning into 'AWS-Project'...
remote: Enumerating objects: 19, done.
remote: Counting objects: 100% (19/19), done.
remote: Compressing objects: 100% (16/16), done.
remote: Total 19 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (19/19), 7.22 KiB | 924.00 KiB/s, done.
```

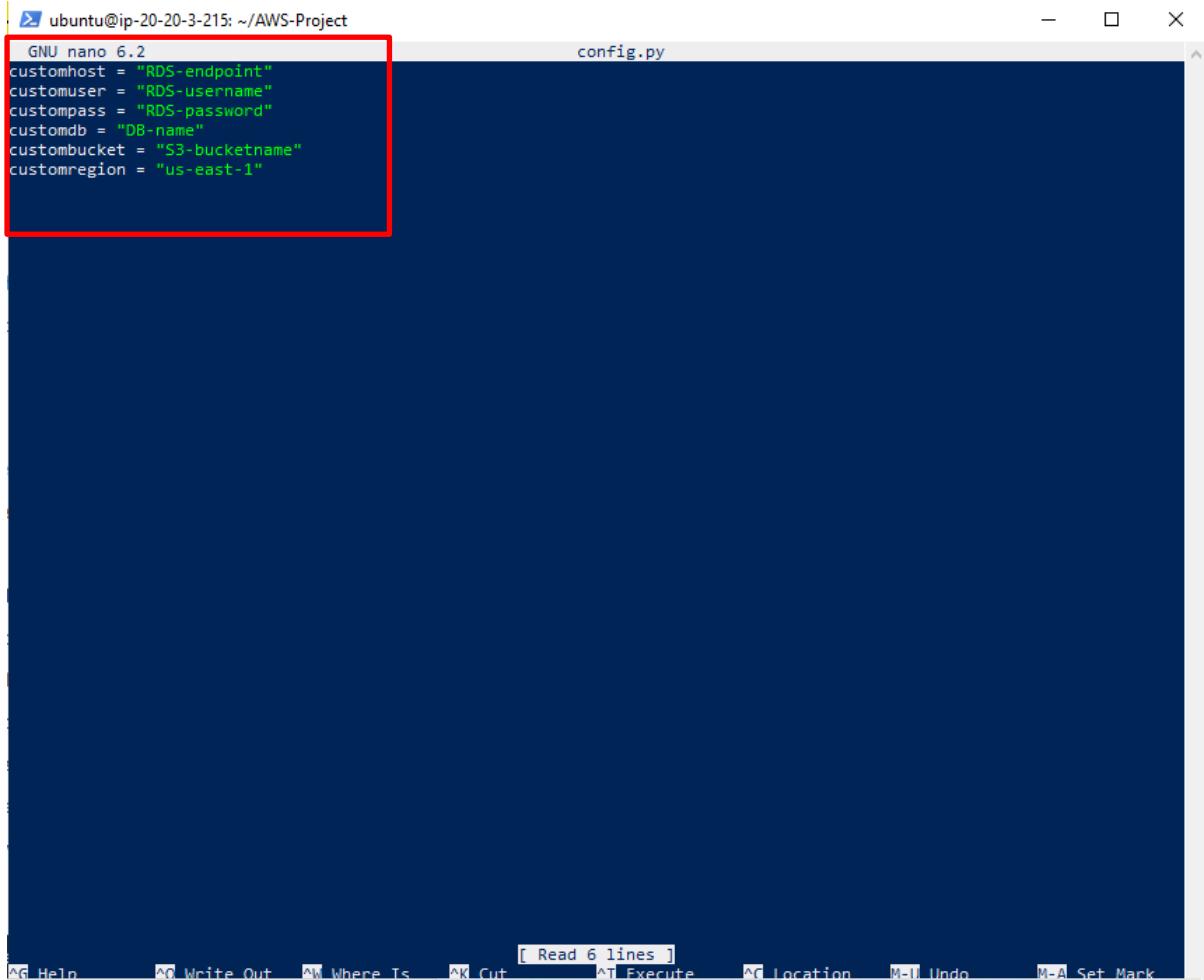
ls

cd AWS-Project

nano config.py

```
ubuntu@ip-20-20-3-215:~$ ls
AWS-Project
ubuntu@ip-20-20-3-215:~$ cd AWS-Project/ ←
ubuntu@ip-20-20-3-215:~/AWS-Project$ ls
EmpApp.py README.md __pycache__ config.py templates
ubuntu@ip-20-20-3-215:~/AWS-Project$ nano config.py ←
```

113) Here we need to paste the s3,db,dbpassword,dbusername,rds endpoint.



```
GNU nano 6.2                                     config.py
customhost = "RDS-endpoint"
customuser = "RDS-username"
custompass = "RDS-password"
customdb = "DB-name"
custombucket = "S3-bucketname"
customregion = "us-east-1"
```

114) open rds copy endpoint paste it in customhost.

The screenshot shows the AWS CloudFormation interface for an Amazon RDS instance. The left sidebar is titled 'Amazon RDS' and lists various options like Dashboard, Databases, Query Editor, etc. The main content area is titled 'Connectivity & security' and contains three tabs: Monitoring, Logs & events, Configuration, Maintenance & backups, and Tags. The 'Configuration' tab is selected. It displays the following details:

Endpoint & port	Networking	Security
Endpoint admininsdb.ciybt5wjksbs.us-east-1.rds.amazonaws.com	Availability Zone us-east-1b VPC project_vpc (vpc-084f8a5472536f25d) Subnet group default-vpc-084f8a5472536f25d Subnets subnet-0073d7946686422c2 subnet-09f5835a168a83382 subnet-08d80e5bb57fa15ac	VPC security groups admin_security (sg-0ebf9e03b34c832a4) Active Publicly accessible No Certificate authority Info rds-ca-2019 Certificate authority date August 22, 2024, 22:38 (UTC+05:30)

115) be remember that we should need to give the database name in customdb(admin_rds). Then click **ctrl+o → enter → ctrl+x**.

The screenshot shows a terminal window on an Ubuntu system (version 20.04) with the command 'nano config.py' running. The file contains the following Python code:

```

GNU nano 6.2
customhost = "admininsdb.ciybt5wjksbs.us-east-1.rds.amazonaws.com"
customuser = "admin"
custompass = "admin123"
customdb = "admin_rds"
custombucket = "adminprojt"
customregion = "us-east-1"

```

116) then paste the below command → you can also get the command in github → open README.md.

36 lines (23 sloc) | 812 Bytes

For Ubuntu use:

```

sudo apt-get update

----->For Sql-client

sudo apt-get install mysql-client

----->For python and related frameworks

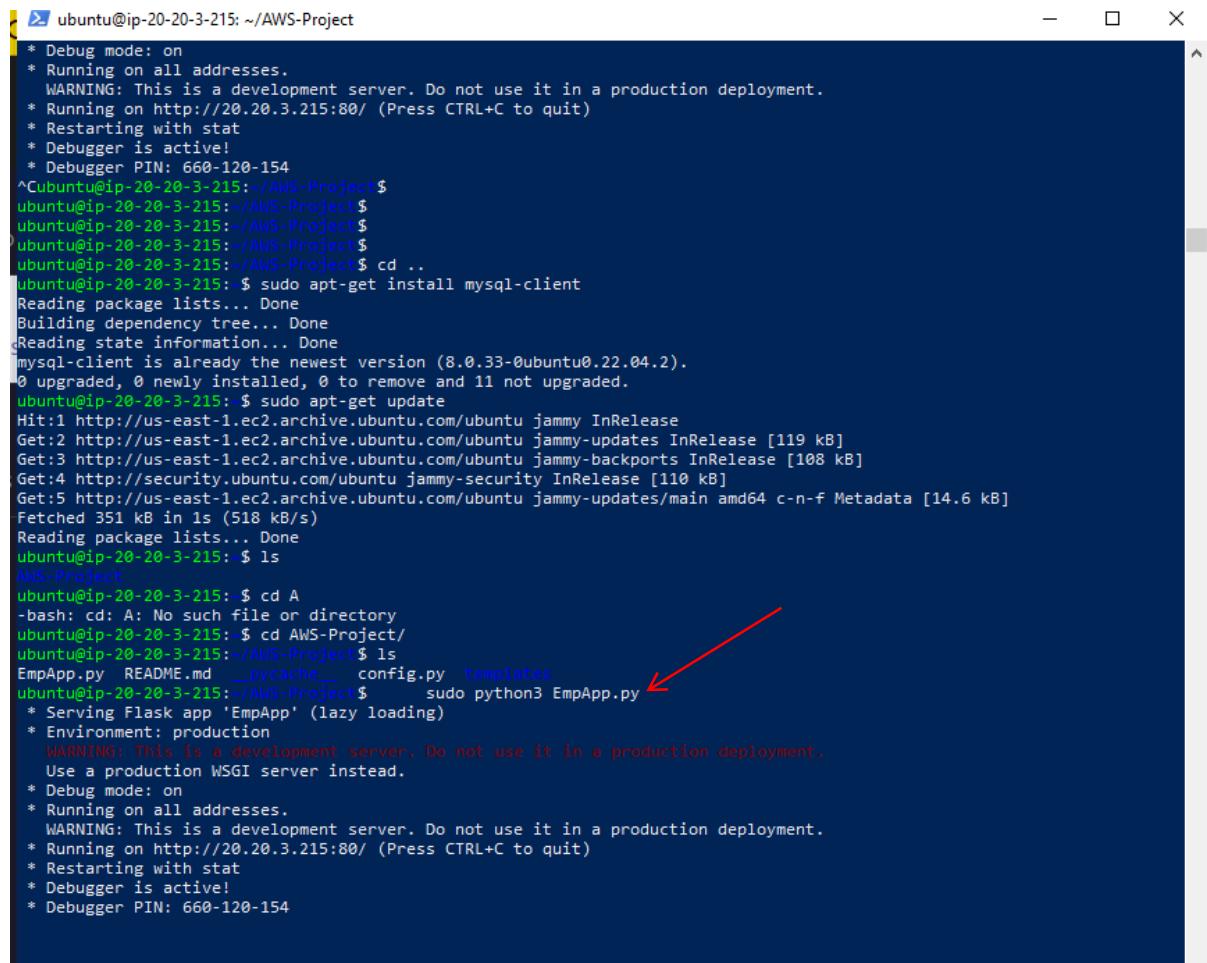
sudo apt-get install python3 -y
sudo apt-get install python3-flask -y
sudo apt-get install python3-pymysql -y
sudo apt-get install python3-boto3 -y

----->For running application

sudo python3 EmpApp.py

```

117) Now run the application using sudo python3 EmpApp.py command.



The screenshot shows a terminal window on an Ubuntu system. The user has navigated to the directory ~/AWS-Project. They have already installed MySQL client via apt-get. Now, they are executing the command `sudo python3 EmpApp.py`. A red arrow points from the text "117)" in the previous slide to this terminal command.

```

ubuntu@ip-20-20-3-215: ~/AWS-Project
* Debug mode: on
* Running on all addresses.
WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://20.20.3.215:80/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 660-120-154
^Cubuntu@ip-20-20-3-215:~/AWS-Project$ 
ubuntu@ip-20-20-3-215:~/AWS-Project$ 
ubuntu@ip-20-20-3-215:~/AWS-Project$ 
ubuntu@ip-20-20-3-215:~/AWS-Project$ cd ..
ubuntu@ip-20-20-3-215: $ sudo apt-get install mysql-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
mysql-client is already the newest version (8.0.33-0ubuntu0.22.04.2).
0 upgraded, 0 newly installed, 0 to remove and 11 not upgraded.
ubuntu@ip-20-20-3-215: $ sudo apt-get update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [108 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 c-n-f Metadata [14.6 kB]
Fetched 351 kB in 1s (518 kB/s)
Reading package lists... Done
ubuntu@ip-20-20-3-215: $ ls
AWS-Project
ubuntu@ip-20-20-3-215: $ cd A
-bash: cd: A: No such file or directory
ubuntu@ip-20-20-3-215: $ cd AWS-Project/
ubuntu@ip-20-20-3-215:~/AWS-Project$ ls
EmpApp.py README.md __pycache__ config.py templates
ubuntu@ip-20-20-3-215:~/AWS-Project$ sudo python3 EmpApp.py
* Serving Flask app 'EmpApp' (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Running on all addresses.
WARNING: This is a development server. Do not use it in a production deployment.
* Running on http://20.20.3.215:80/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 660-120-154

```

118) Once the application start running → go to load balancer → copy the DNS name and paste it in the browser.

EC2 old experience is going away soon.
This EC2 console experience is being deprecated and will soon be replaced with the new experience. The new experience will be made as the default over the next few months. All new features and improvements will only be available in the new experience. However, you can still switch between the old and new experiences by using the New EC2 Experience toggle on the top left of the navigation pane.

Name	DNS name	State	VPC ID	Availability Zones	Type
adminloaddbc	adminloaddbc-1075133596.u...	Active	vpc-01625a14d9642ab24	us-east-1b, us-east-1a	application

BASIC CONFIGURATION

Name	adminloaddbc
ARN	arn:aws:elasticloadbalancing:us-east-1:258506467594:loadbalancer/app/adminloaddbc/2f0bd12ded39d81f
DNS name	adminloaddbc-1075133596.us-east-1.elb.amazonaws.com
State	Active

119) the hosted page will get open.

Not secure | adminloaddbc-1075133596.us-east-1.elb.amazonaws.com

Employee Database

GET EMPLOYEE INFORMATION

Employee ID:

First Name:

Last Name:

Primary Skills:

Location:

Image: Choose File aboutus.html

UPDATE DATABASE

120) Now go to rds copy the endpoint.

121) get back to power shell → type

cd ..

mysql -h paste rds endpoint -u dbuser name -p db password

(eg.mysql -h database-1admin.ciybts5wjkbs.us-east-1.rds.amazonaws.com -u admin-p)

```
ubuntu@ip-20-20-3-22: $ cd AWS-Project/
ubuntu@ip-20-20-3-22:~/AWS-Project$ ls
EmpApp.py README.md __pycache__ config.py templates
ubuntu@ip-20-20-3-22:~/AWS-Project$ sudo python3 EmpApp.py
* Serving Flask app 'EmpApp' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
Traceback (most recent call last):
  File "/home/ubuntu/AWS-Project/EmpApp.py", line 161, in <module>
    app.run(host='0.0.0.0',port=80,debug=True)
  File "/usr/lib/python3/dist-packages/flask/app.py", line 922, in run
    run_simple(t.cast(str, host), port, self, **options)
  File "/usr/lib/python3/dist-packages/werkzeug/serving.py", line 984, in run_simple
    s.bind(server_address)
OSError: [Errno 98] Address already in use
ubuntu@ip-20-20-3-22:~/AWS-Project$ cd ..
ubuntu@ip-20-20-3-22: $ mysql -h database-1admin.ciybts5wjkbs.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 56
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

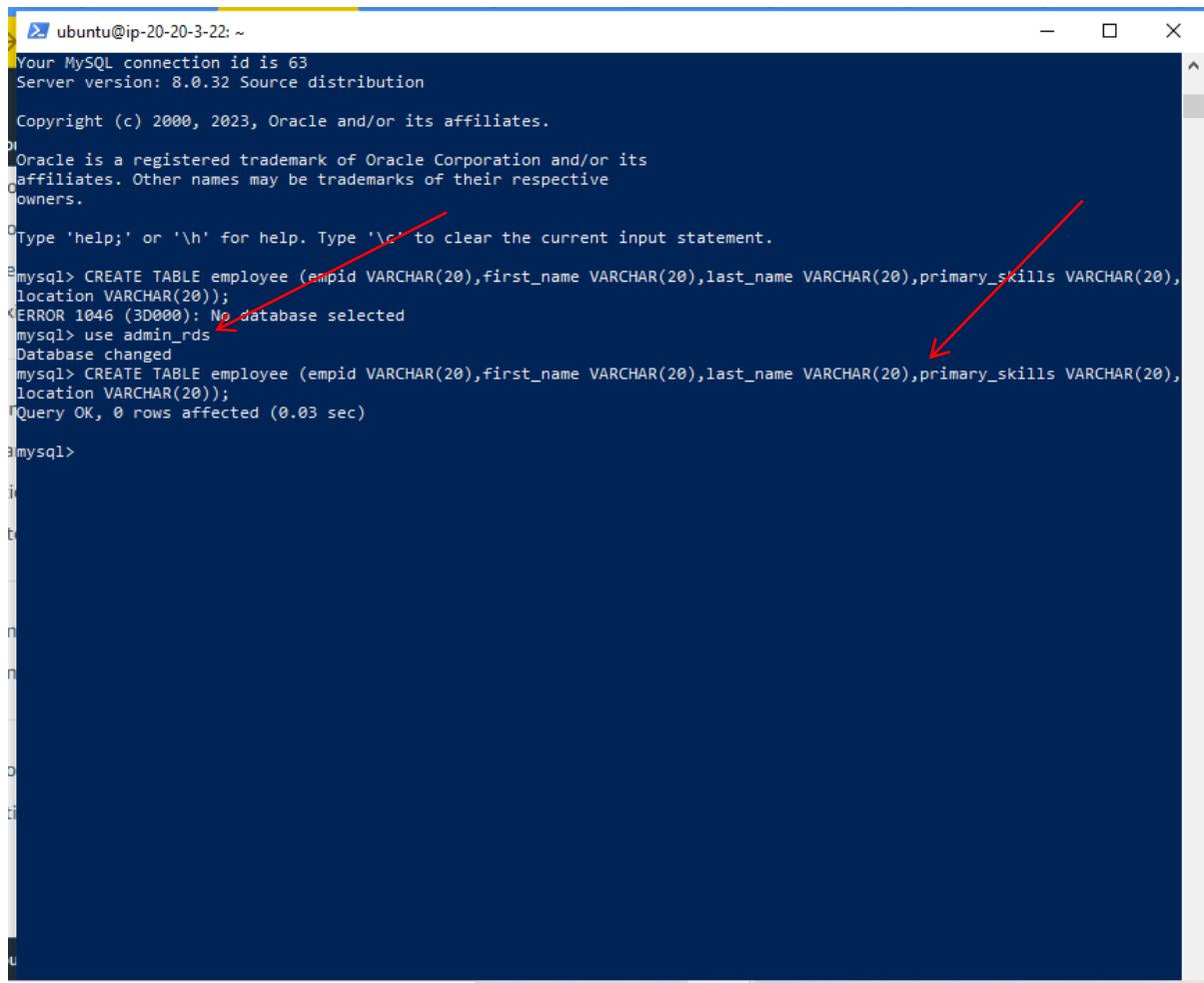
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

Type

use dbname(eg.use admin_rds)

```
CREATE TABLE employee (empid VARCHAR(20),first_name  
VARCHAR(20),last_name VARCHAR(20),primary_skills  
VARCHAR(20),location VARCHAR(20));
```



```
ubuntu@ip-20-20-3-22: ~  
Your MySQL connection id is 63  
Server version: 8.0.32 Source distribution  
  
Copyright (c) 2000, 2023, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> CREATE TABLE employee (empid VARCHAR(20),first_name VARCHAR(20),last_name VARCHAR(20),primary_skills VARCHAR(20),  
location VARCHAR(20));  
ERROR 1046 (3D000): No database selected  
mysql> use admin_rds  
Database changed  
mysql> CREATE TABLE employee (empid VARCHAR(20),first_name VARCHAR(20),last_name VARCHAR(20),primary_skills VARCHAR(20),  
location VARCHAR(20));  
Query OK, 0 rows affected (0.03 sec)  
  
mysql>
```

122) the table has been created successfully.

Type

show tables;

the table will appear.

```
ubuntu@ip-20-20-3-22: ~
Your MySQL connection id is 63
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

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

mysql> CREATE TABLE employee (empid VARCHAR(20),first_name VARCHAR(20),last_name VARCHAR(20),primary_skills VARCHAR(20),
location VARCHAR(20));
ERROR 1046 (3D000): No database selected
mysql> use admin_rds
Database changed
mysql> CREATE TABLE employee (empid VARCHAR(20),first_name VARCHAR(20),last_name VARCHAR(20),primary_skills VARCHAR(20),
location VARCHAR(20));
Query OK, 0 rows affected (0.03 sec)

mysql> show databases
-> show databases;
-> show tables;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version
for the right syntax to use near 'show databases':
show tables' at line 2
mysql> show tables;

| Tables_in_admin_rds |
+-----+
| employee      |
+-----+
1 row in set (0.00 sec)

mysql> show tables;
+-----+
| Tables_in_admin_rds |
+-----+
| employee      |
+-----+
1 row in set (0.01 sec)

mysql>
```

123) Now search for IAM in aws.

The screenshot shows the AWS Management Console search results for the term 'iam'. The left sidebar includes links for CloudFormation, Services (9), Documentation (47,207), Tutorials (2), Events (13), Marketplace (476), Features (19), Blogs (1,554), Knowledge Articles (16), and Resource Access Manager. The main search results page displays 'Services' and 'Features' sections. The 'Services' section lists IAM, IAM Identity Center (successor to AWS Single Sign-On), Resource Access Manager, and Serverless Application Repository. The 'Features' section lists Groups, default-vpc-, and Certificate authority. A red box highlights the 'IAM' service card, which is described as 'Manage access to AWS resources'.

124) click roles.

The screenshot shows the AWS IAM dashboard. On the left, there's a sidebar with navigation links like 'Identity and Access Management (IAM)', 'Dashboard', 'Access management', 'Access reports', and 'Tools'. A red box highlights the 'Roles' link under 'Access management'. The main content area is titled 'IAM dashboard' and contains 'Security recommendations'. It lists three items: 'Add MFA for root user' (warning), 'Root user has no active access keys' (info), and 'Update your access permissions for AWS Billing, Cost Management, and Account consoles' (info). To the right, there's an 'AWS Account' section with details like 'Account ID: 258506467594', 'Sign-in URL: https://258506467594.signin.amazonaws.com/console', and a 'View affected policies' button. There's also a 'Quick Links' section for 'My security credentials'.

125) choose aws service → use case as ec2

The screenshot shows the 'Trusted entity type' configuration screen in CloudFormation. It's Step 2 of a process, with 'Add permissions' selected. Below it, 'Step 3' is 'Name, review, and create'. The 'Trusted entity type' section has four options: 'AWS service' (selected, highlighted with a red box), 'AWS account', 'Web identity', and 'SAML 2.0 federation'. Under 'Use case', 'EC2' is selected. A dropdown menu for 'Use cases for other AWS services' is open, showing 'Choose a service to view use case'. At the bottom, there are links for 'CloudShell', 'Feedback', 'Language', and copyright information.

126) search for s3 and add s3 full access.

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Add permissions [Info](#)

Permissions policies (Selected 1/848) [Info](#)

Choose one or more policies to attach to your new role.

Filter policies by property or policy name and press enter. 10 matches < 1 > [Clear filters](#)

Policy name	Type	Description
<input type="checkbox"/> s3crf_for_s3versbucket...	Custom...	
<input checked="" type="checkbox"/> AmazonS3FullAccess	AWS m...	Provides full access to all buckets via the AWS Management Console.
<input type="checkbox"/> AmazonS3ReadOnly...	AWS m...	Provides read only access to all buckets via the AWS Management Console.
<input type="checkbox"/> AmazonDMSRedsh...	AWS m...	Provides access to manage S3 settings for Redshift endpoints for DMS.
<input type="checkbox"/> QuickSightAccessF...	AWS m...	Policy used by QuickSight team to access customer data produced by S3 ...
<input type="checkbox"/> AmazonS3Outposts...	AWS m...	Provides full access to Amazon S3 on Outposts via the AWS Management ...

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

127) search for dynamo db and add dynamodb full access.

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Add permissions [Info](#)

Permissions policies (Selected 2/848) [Info](#)

Choose one or more policies to attach to your new role.

Filter policies by property or policy name and press enter. 4 matches < 1 > [Clear filters](#)

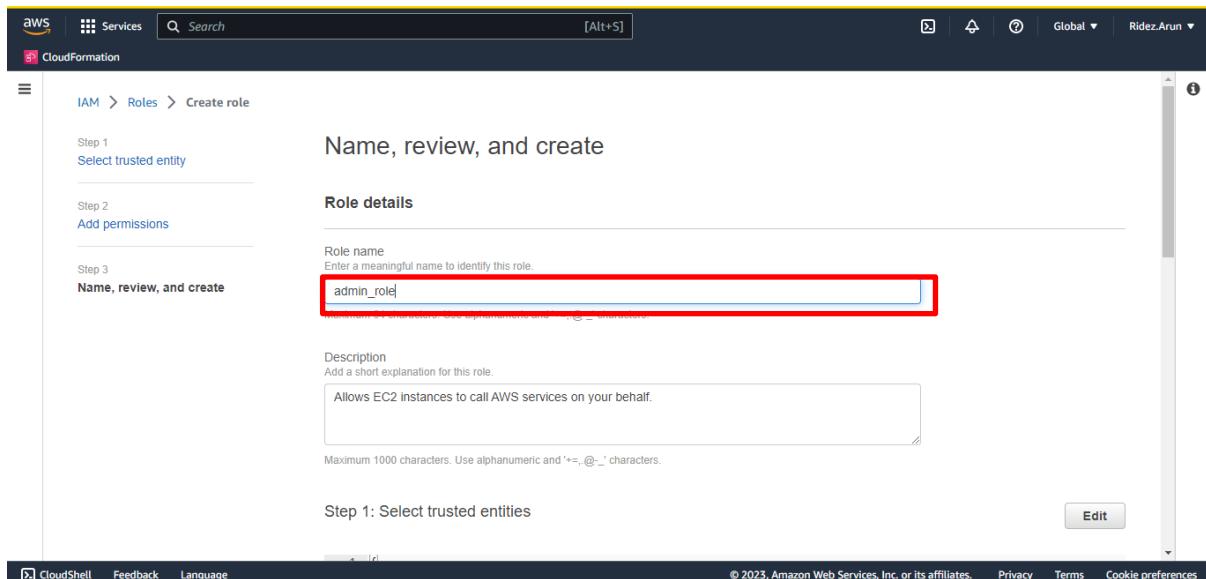
Policy name	Type	Description
<input checked="" type="checkbox"/> AmazonDynamoDB...	AWS m...	Provides full access to Amazon DynamoDB via the AWS Management Co...
<input type="checkbox"/> AmazonDynamoDB...	AWS m...	Provides read only access to Amazon DynamoDB via the AWS Managem...
<input type="checkbox"/> AWSLambdaInvoca...	AWS m...	Provides read access to DynamoDB Streams.
<input type="checkbox"/> AWSLambdaDyna...	AWS m...	Provides list and read access to DynamoDB streams and write permission...

Set permissions boundary - optional [Info](#)

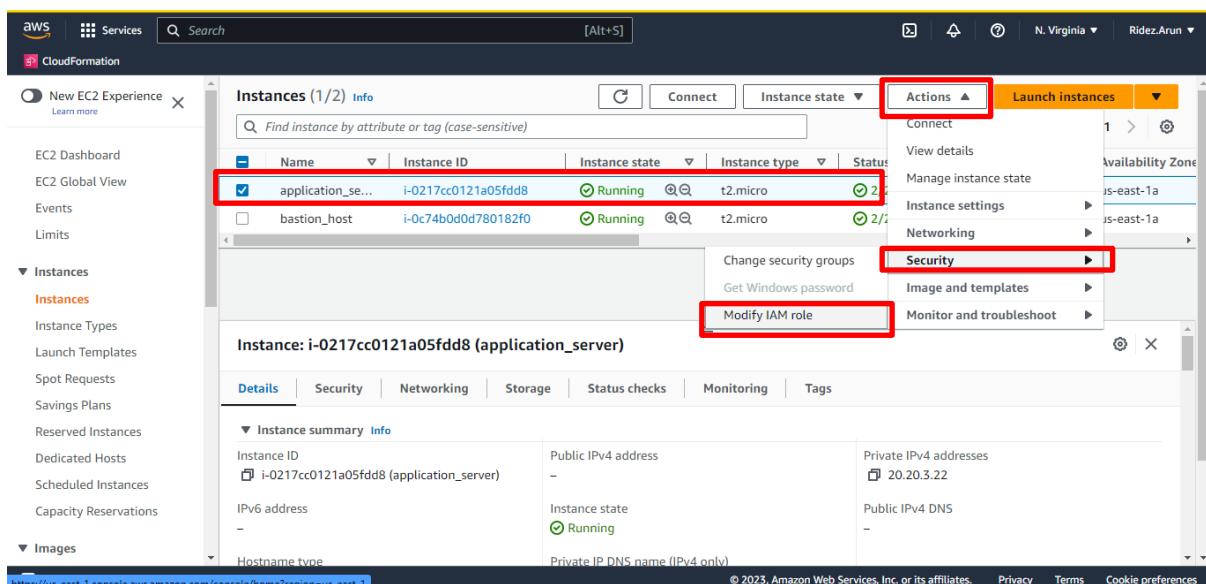
Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission.

CloudShell Feedback Language © 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

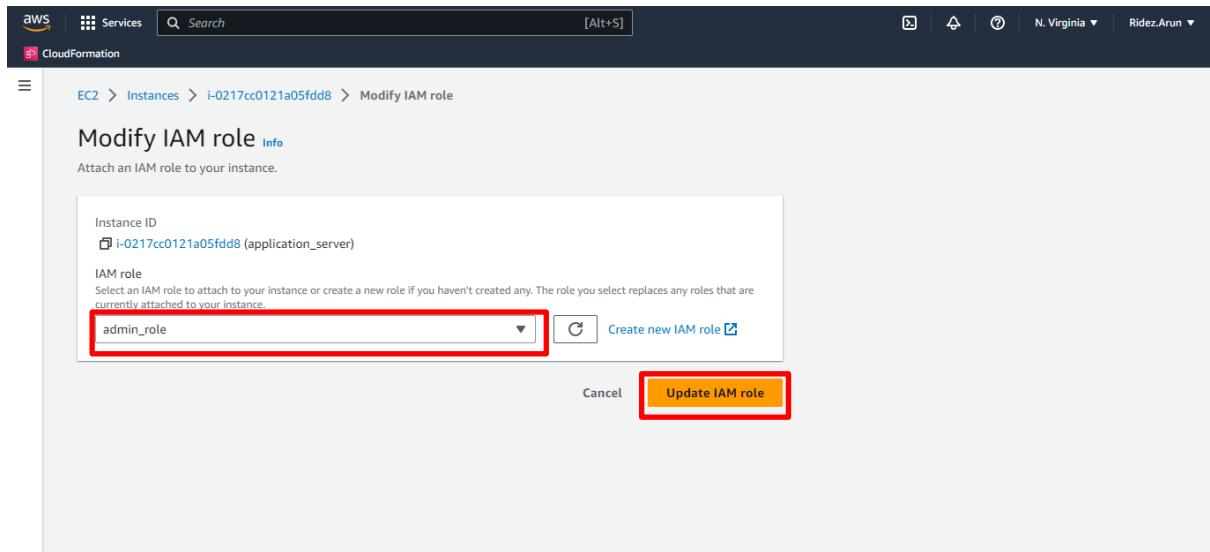
128) Give name to role.



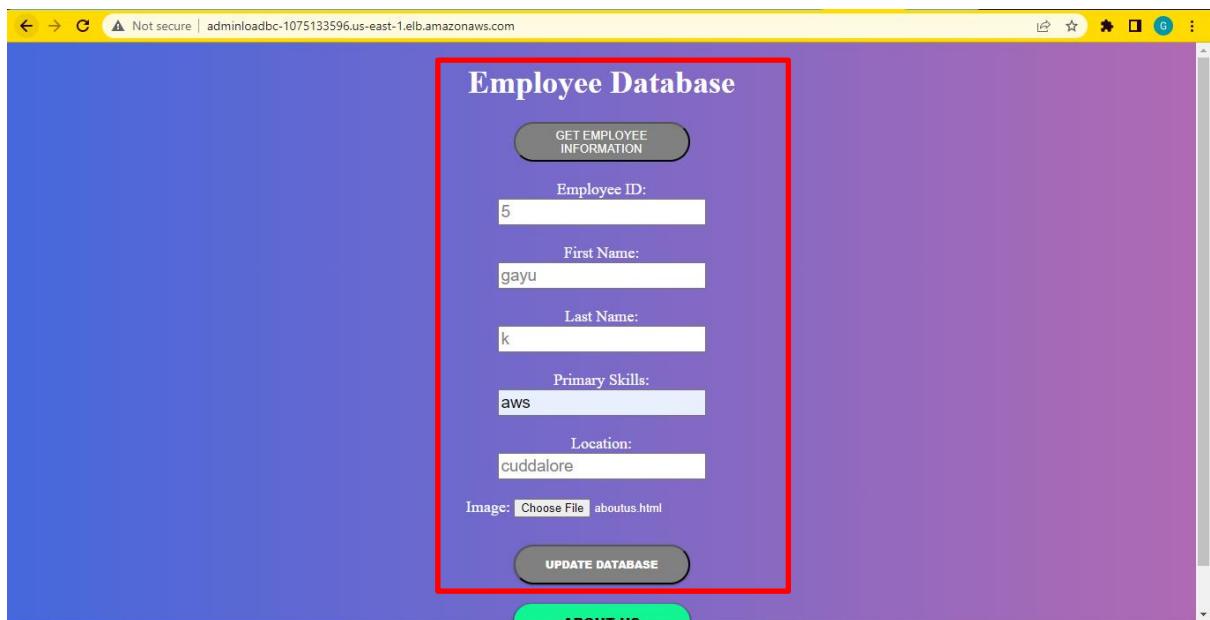
129) Now go to instance → select private instance(application_server) → actions → security → modify IAM role.



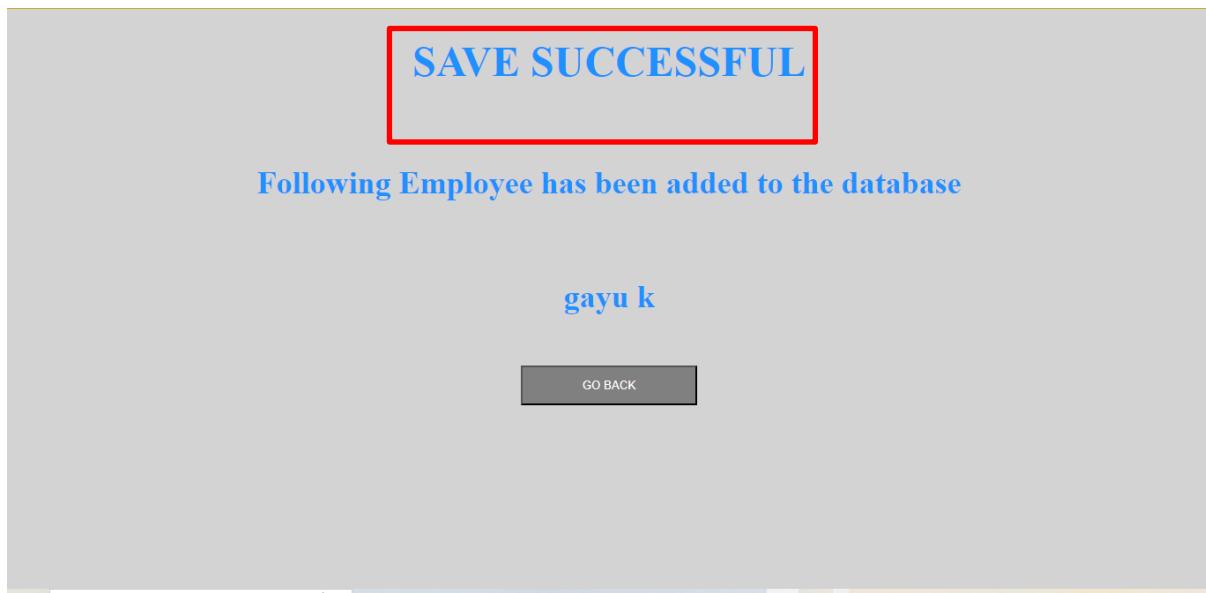
130) choose the created role → click update iam role.



131) Now go to the hosted page → give credentials → click update database.



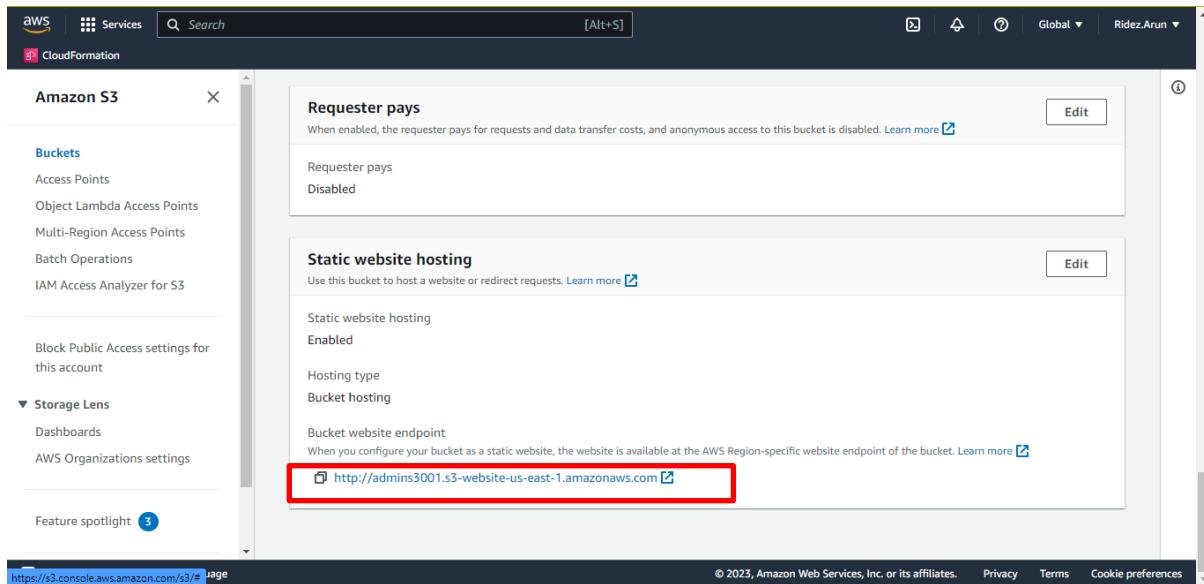
132) the data get added to database successfully.



133) Now go to s3→go to properties.

A screenshot of the Amazon S3 Buckets list page. The left sidebar shows navigation options like Buckets, Storage Lens, and Feature spotlight. The main area displays an account snapshot and a table of buckets. The first bucket, "admins3001", is highlighted with a red box. Its details are shown in a modal overlay: Name: admins3001, AWS Region: US East (N. Virginia) us-east-1, Access: Public, Creation date: May 24, 2023, 10:13:02 (UTC+05:30). The second bucket, "cf-templates-a6wehwzeypn1-us-east-1", is also listed with its details.

134) static website hosting→copy the endpoint.



135)open new powershell → connect to private and public instance with same above process.

Proceed this alone

Cd downloads

ssh -i "keypairname.pem" ubuntu@public ip of instance

ls

Ssh ubuntu@20.20.3.215 -i keyfile.pem

ls

cd AWS-Project

ls

cd templates

ls

nano AddEmp.html

the nano window will get open.

The screenshot shows a terminal window with the following content:

```
ubuntu@ip-20-20-3-22: ~/AWS-Project/templates
ubuntu@ip-20-20-1-90: $ ls
publickey.pem
ubuntu@ip-20-20-1-90: $ ssh ubuntu@20.20.3.22 -i pubkey.pem
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.19.0-1025-aws x86_64)

 * Documentation: https://help.ubuntu.com
 * Management: https://landscape.canonical.com
 * Support: https://ubuntu.com/advantage

 System information as of Wed May 24 08:44:47 UTC 2023

 System load: 0.080078125 Processes: 113
 Usage of /: 24.9% of 7.57GB Users logged in: 1
 Memory usage: 37% IPv4 address for eth0: 20.20.3.22
 Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

1 update can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Wed May 24 08:08:42 2023 from 20.20.1.90
ubuntu@ip-20-20-3-22: $ ls
AWS-Project
ubuntu@ip-20-20-3-22: $ cd AWS-Project/
ubuntu@ip-20-20-3-22:~/AWS-Project$ ls
EmpApp.py README.md __pycache__ config.py templates
ubuntu@ip-20-20-3-22:~/AWS-Project$ cd templates/
ubuntu@ip-20-20-3-22:~/AWS-Project/templates$ ls
AddEmp.html AddEmpOutput.html GetEmp.html GetEmpOutput.html _DS_Store
ubuntu@ip-20-20-3-22:~/AWS-Project/templates$ nano AddEmp.html
```

Red arrows point to the following lines in the terminal output:

- An arrow points to the line "publickey.pem".
- Two arrows point to the line "1 update can be applied immediately. To see these additional updates run: apt list --upgradable".
- Three arrows point to the line "Enable ESM Apps to receive additional future security updates. See https://ubuntu.com/esm or run: sudo pro status".
- Two arrows point to the line "ubuntu@ip-20-20-3-22: \$ ls".
- One arrow points to the line "AddEmp.html AddEmpOutput.html GetEmp.html GetEmpOutput.html _DS_Store".

136) scroll down the code you will see < a href/about.....>remove about and paste the s3 endpoint there,.then press ctrl+o,enter,ctrl+x.

```
GNU nano 6.2                                     AddEmp.html
    text-align: center;
    text-transform: uppercase;
    color: #FFF;
    box-shadow: 0 0 20px #eee;
    border-radius: 20px;
    width: 200px;
    box-shadow: 0 1px 3px rgba(0,0,0,1), 0 1px 2px rgba(0,0,0,0.5);
    display: inline-block;
    border-radius: 20px;
}


```

```
</style>
<center>
    <font color="white" size="4" style="font-family: Gadget">
        <h1 style="color: White">Employee Database</h1>
    </font>
    <body bgcolor="lightgrey">
        <form action="/addemp" autocomplete="on" method = "POST" enctype="multipart/form-data">
            <button type="submit" formaction="/getemp" style="background: grey; height: 45px; width: 200px; border-radius: 20px; color: black; font-size: 14pt; margin-bottom: 10px;"></button>
            Employee ID:<br> <input style="height:24px;font-size:14pt; color:grey;" type="number" name="emp_id" required="required" />
            First Name:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="first_name" required="required" />
            Last Name:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="last_name" required="required" />
            Primary Skills:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="primary_skills" required="required" />
            Location:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="location" required="required" />
            Image: <input type="file" name="emp_image_file" style="height:2nt-size:14pt,color:5px;">
            <button class="gradient-button" type="submit" style="background: grey; height: 45px; width: 200px; border-radius: 20px; color: black; font-size: 14pt; margin-top: 10px;"></button>
        </form>
        <a href="/about" class="gradient-b" type="submit" style="background: grey; height: 45px; width: 200px; color: black; size: 14pt; border-radius: 20px; margin-top: 10px;"></a>
    </body>
</center>
</html>
```

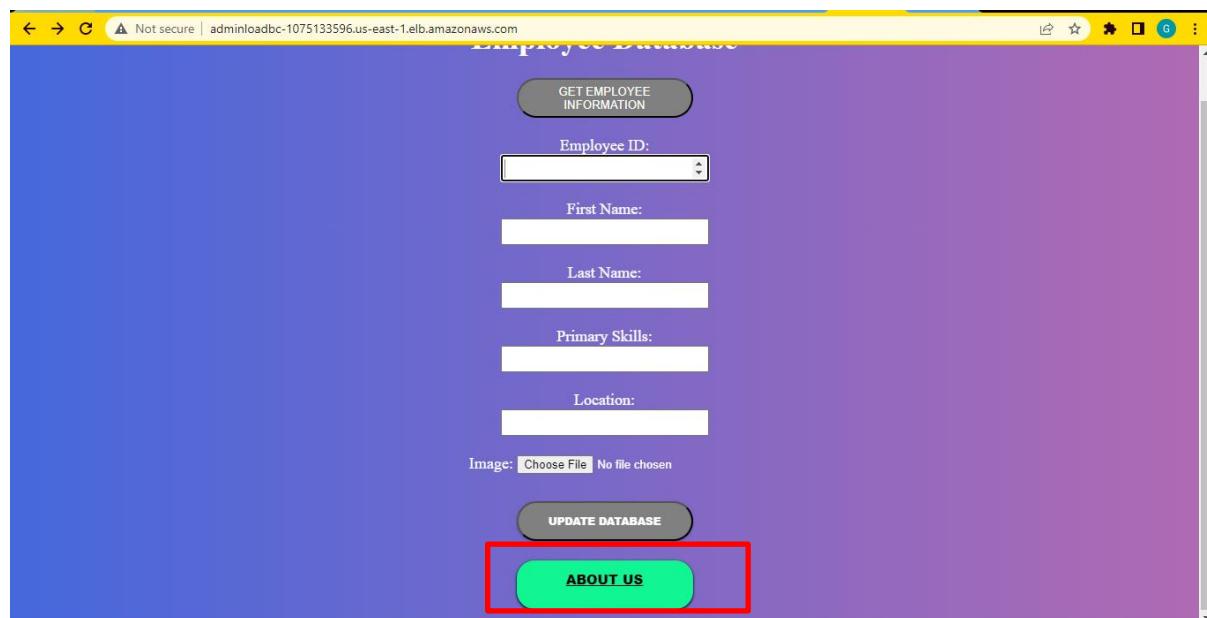
⌘ H Help ⌘ W Write Out ⌘ M Where Is ⌘ X Cut ⌘ E Execute ⌘ C Location ⌘ U Undo ⌘ A Set Mark

```
ubuntu@ip-20-20-3-22: ~/AWS-Project/templates          AddEmp.html *
```

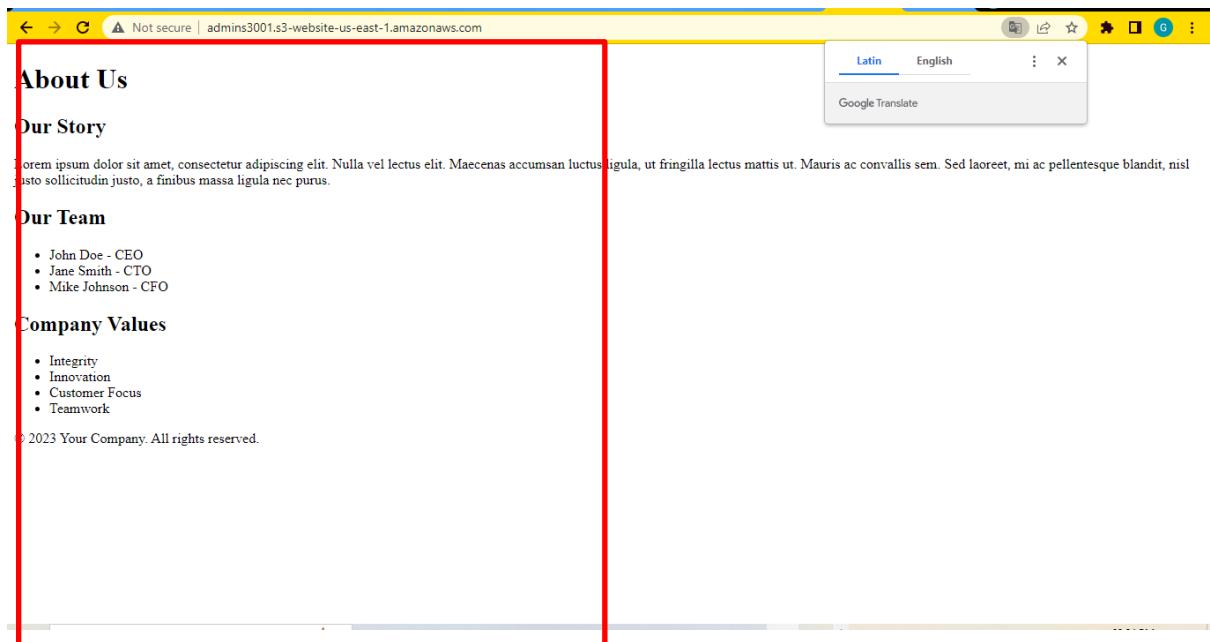
```
GNU nano 6.2
    text-align: center;
    text-transform: uppercase;
    color: #FFF;
    box-shadow: 0 0 20px #eee;
    border-radius: 20px;
    width: 200px;
    box-shadow: 0 1px 3px rgba(0,0,0,1), 0 1px 2px rgba(0,0,0,0.5);
    display: inline-block;
    border-radius: 20px;
}

</style>
<center>
    <font color="white" size="4" style="font-family: Gadget">
        <h1 style="color: White">Employee Database</h1>
        <body bgcolor="lightgrey">
            <form action="/adtemp" autocomplete="on" method = "POST" enctype="multipart/form-data">
                <button type="submit" formaction="/getemp" style="background: grey; height: 45px; width: 150px; border-radius: 20px; margin-bottom: 10px;">Get Employee</button>
                Employee ID:<br> <input style="height:24px;font-size:14pt; color:grey;" type="number" name="emp_id" />
                First Name:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="emp_fn" />
                Last Name:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="emp_ln" />
                Primary Skills:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="emp_ps" />
                Location:<br> <input style="height:24px;font-size:14pt; color:grey;" type="text" name="emp_loc" />
                Image: <input type="file" name="emp_image_file" style="height:2nt-size:14pt;color:5px;" />
                <button class="gradient-button" type="submit" style="background: grey; height: 45px; width: 150px; border-radius: 20px; margin-top: 10px;">Submit</button>
            </form>
            <a href="http://admin3061.s3-website-us-east-1.amazonaws.com" class="gradient-R" type="submit" style="background: grey; height: 45px; width: 150px; border-radius: 20px; margin-top: 10px;">View All Employees</a>
        </body>
    </font>
</center>
</html>
```

137) now go to the hosted page → click about us.



138) our aboutus.html page will appear.



Not secure | admins3001.s3-website-us-east-1.amazonaws.com

About Us

Our Story

• lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla vel lectus elit. Maecenas accumsan luctus ligula, ut fringilla lectus mattis ut. Mauris ac convallis sem. Sed laoreet, mi ac pellentesque blandit, nisl justo sollicitudin justo, a finibus massa ligula nec purus.

Our Team

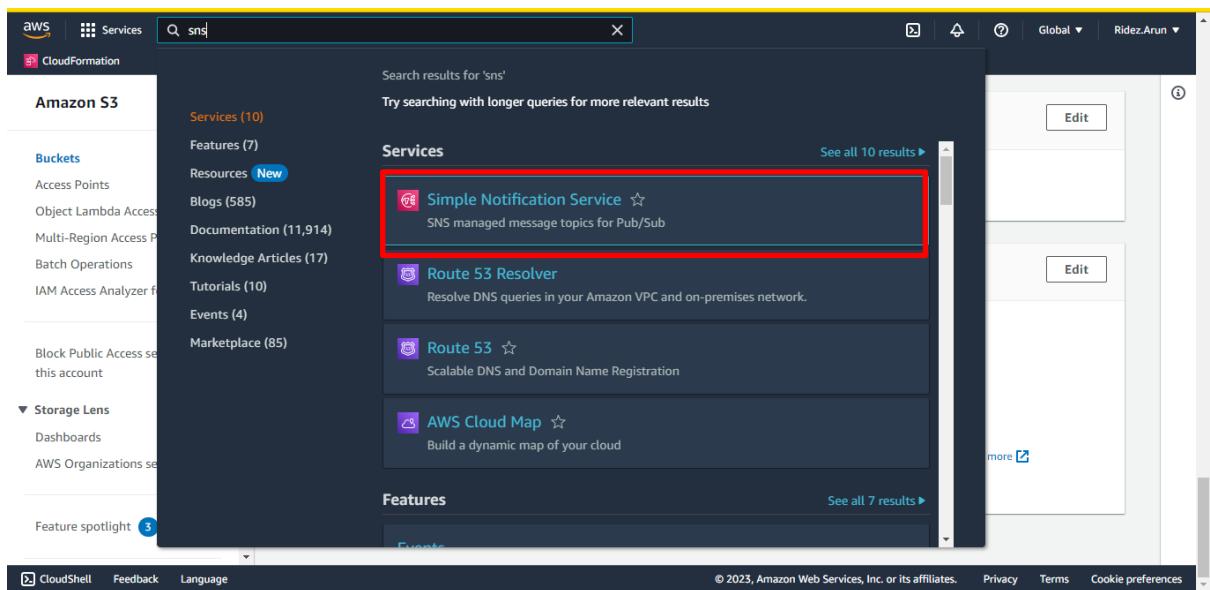
- John Doe - CEO
- Jane Smith - CTO
- Mike Johnson - CFO

Company Values

- Integrity
- Innovation
- Customer Focus
- Teamwork

© 2023 Your Company. All rights reserved.

139) search for sns in aws.



aws | Services | sns | CloudFormation

Amazon S3

Buckets

Access Points

Object Lambda Access

Multi-Region Access Points

Batch Operations

IAM Access Analyzer

Block Public Access settings for this account

Storage Lens

Dashboards

AWS Organizations settings

Feature spotlight

Services (10)

Features (7)

Resources New

Blogs (585)

Documentation (11,914)

Knowledge Articles (17)

Tutorials (10)

Events (4)

Marketplace (85)

Search results for 'sns'

Try searching with longer queries for more relevant results

Services

See all 10 results ▾

Simple Notification Service ☆
SNS managed message topics for Pub/Sub

Route 53 Resolver
Resolve DNS queries in your Amazon VPC and on-premises network.

Route 53 ☆
Scalable DNS and Domain Name Registration

AWS Cloud Map ☆
Build a dynamic map of your cloud

Features

See all 7 results ▾

CloudShell Feedback Language

© 2023, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

140) click next step.

Amazon Simple Notification Service

Pub/sub messaging for microservices and serverless applications.

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

Benefits and features

Create topic

Topic name: MyTopic

Next step

Pricing

Amazon SNS has no upfront costs. You pay based on the

141) choose standard → give name.

Create topic

Details

Type: Standard

Name: admin_sns

142) choose basic → select everyone in publishers.

The screenshot shows the AWS CloudFormation console with the 'CloudFormation' service selected. In the main area, there's a section titled 'Choose method' with two options: 'Basic' (selected) and 'Advanced'. Below this, there are two dropdown menus: 'Publishers' (set to 'Everyone') and 'Subscribers' (set to 'Everyone'). To the right, there's a 'JSON preview' window displaying the generated AWS SNS policy:

```
{ "Version": "2008-10-17", "Id": "__default_policy_ID", "Statement": [ { "Sid": "__default_statement_ID", "Effect": "Allow", "Principal": { "AWS": "*" }, "Action": [ "SNS:Publish", "SNS:RemovePermission", "SNS:SetTopicAttributes", "SNS:DeleteTopic", 
```

143) click create topic.

The screenshot shows the AWS CloudFormation console with the 'CloudFormation' service selected. The interface displays several optional configuration sections: 'Delivery policy (HTTP/S) - optional', 'Delivery status logging - optional', 'Tags - optional', and 'Active tracing - optional'. At the bottom right, there are 'Cancel' and 'Create topic' buttons, with the 'Create topic' button highlighted by a red box.

144) topic has been created.

The screenshot shows the AWS SNS Topics page. A green notification bar at the top right says "Topic admin_sns created successfully. You can create subscriptions and send messages to them from this topic." Below this, the "admin_sns" topic details are shown, including its Name (admin_sns), Display name (admin_sns), ARN (arn:aws:sns:us-east-1:258506467594:admin_sns), Topic owner (258506467594), and Type (Standard). There are tabs for Subscriptions, Access policy, Data protection policy, Delivery policy (HTTP/S), and Delivery status logging. The Subscriptions tab is selected.

145) now click subscription in left panel → click create subscription.

The screenshot shows the AWS SNS Subscriptions page. The "Subscriptions" link in the left sidebar is highlighted with a red box. On the main page, there is a table header for "Subscriptions (0)" with columns for ID, Endpoint, Status, Protocol, and Topic. Below the table, it says "No subscriptions found". At the bottom right of the table area, there is a "Create subscription" button, which is highlighted with a yellow box.

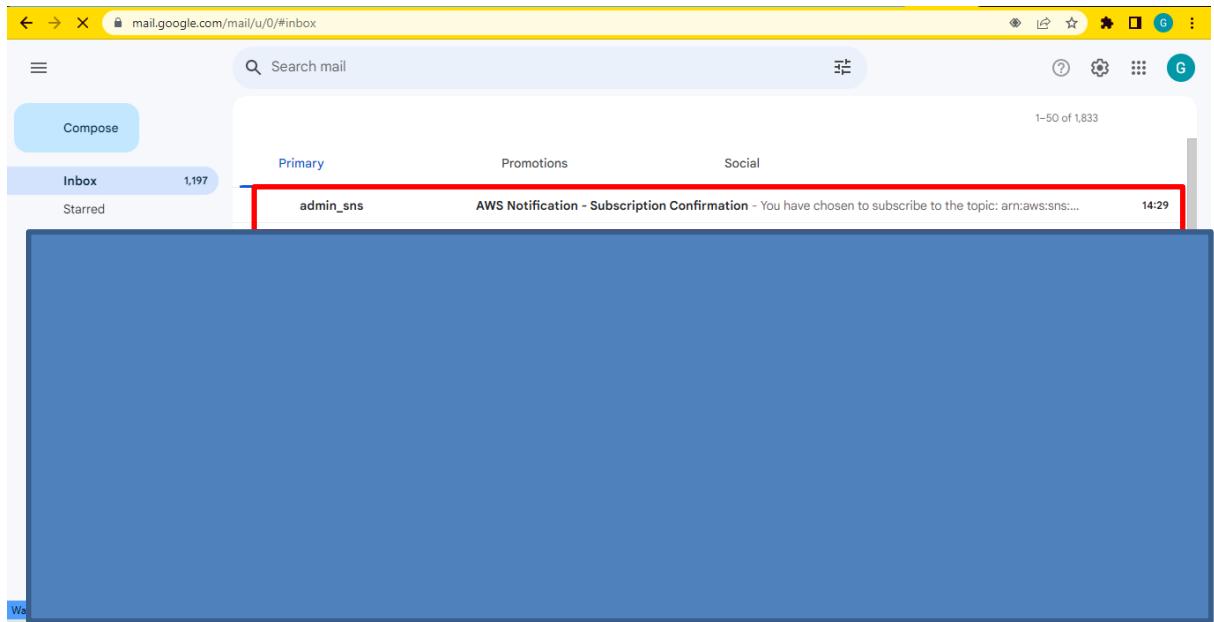
146) choose created sns → protocol as email → endpoint as our mail id.

The screenshot shows the 'Create subscription' step in the AWS CloudFormation console. The 'Topic ARN' field contains 'arn:aws:sns:us-east-1:258506467594:admin_sns'. The 'Protocol' dropdown is set to 'Email'. The 'Endpoint' field contains 'gayathrikinfotech@gmail.com'. A note at the bottom says 'After your subscription is created, you must confirm it.' The URL in the address bar is 'https://us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/1234567890123456789/subscriptions'.

147)click subscription.

The screenshot shows the 'Create subscription' step in the AWS CloudFormation console. The 'Protocol' dropdown is set to 'Email'. The 'Endpoint' field contains 'gayathrikinfotech@gmail.com'. A note at the bottom says 'After your subscription is created, you must confirm it.' Below this are sections for 'Subscription filter policy - optional' and 'Redrive policy (dead-letter queue) - optional'. The 'Create subscription' button is highlighted with a red box. The URL in the address bar is 'https://us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/1234567890123456789/subscriptions'.

148)check mail regarding subscription conformation.



149) it will be in pending until we conform.

A screenshot of the AWS SNS console. On the left, a sidebar shows options like Dashboard, Topics, Subscriptions, and Mobile. The Subscriptions option is selected. In the main area, it shows a list of subscriptions under a topic. One subscription is highlighted with a red box. The details for this subscription are shown in a modal window. The "Status" field is highlighted with a red box and contains the value "Pending confirmation". Other fields shown include ARN, Endpoint (an email address), Topic (admin sns), and Subscription Principal (an IAM ARN).

150) click confirm subscription.

The screenshot shows a Gmail inbox with 1,196 messages. A specific email from 'admin sns <no-reply@sns.amazonaws.com>' is selected, titled 'AWS Notification - Subscription Confirmation'. The message body contains a confirmation link: 'Confirm subscription' (which is highlighted with a red box). Below the link, a note says: 'Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to [sns-opt-out](#)'. At the bottom of the email view, there are 'Reply' and 'Forward' buttons. A modal dialog at the bottom asks 'Enable desktop notifications for Gmail.' with 'OK' and 'No, thanks' buttons.

151) the pending uproved.

The screenshot shows a web browser displaying an AWS SNS confirmation page. The URL is 'sns.us-east-1.amazonaws.com/confirmation.html?TopicArn=arn:aws:sns:us-east-1:258506467594:admin_sns&Token=2336412f37fb687f5d51e6e2425c464de0761df...'. The page title is 'Simple Notification Service'. The main content area is titled 'Subscription confirmed!' and contains the message: 'You have successfully subscribed.' Below this, it shows the 'Your subscription's id is:' followed by the ARN: 'arn:aws:sns:us-east-1:258506467594:admin_sns:fec9626f-c1e8-468e-aa7f-069b5959ec9bb'. At the bottom, there is a link: 'If it was not your intention to subscribe, [click here to unsubscribe](#)'. The entire content area is highlighted with a red box.



The screenshot shows the AWS SNS console with the path: Amazon SNS > Topics > admin sns > Subscription: fec9626f-c1e8-468e-aa7f-069b59ecd9bb. The 'Details' section is displayed, showing the ARN, Endpoint (gayathrikinfotech@gmail.com), Topic (admin sns), and Subscription Principal (arn:aws:iam::258506467594:root). A red box highlights the 'Status' field, which is set to 'Confirmed'. Another red box highlights the 'Protocol' field, which is set to 'EMAIL'.

152) Now go to s3→properties→event notifications→create.

The screenshot shows the AWS S3 console with the path: Amazon S3 > Buckets. The 'Event notifications (0)' section is highlighted with a red box. A red box also highlights the 'Create event notification' button. Below this, there is a table with columns: Name, Event types, Filters, Destination type, and Destination. The table currently displays 'No event notifications' and a note: 'Choose Create event notification to be notified when a specific event occurs.' A red box highlights the 'Create event notification' button again.

153) give name.

The screenshot shows the 'Create event notification' configuration page in the AWS CloudFormation console. The 'Event name' field is highlighted with a red box and contains the value 'admininevent'. A note below the field states: 'Event name can contain up to 255 characters.'

154)enable object creation,removal,acl,tagging.

The screenshot shows the 'Event types' section of the 'Create event notification' configuration page. The 'Object creation', 'Object removal', and 'Object restore' sections are highlighted with a large red box. Under 'Object creation', the 'All object create events' checkbox is checked. Under 'Object removal', the 'All object removal events' checkbox is checked. Under 'Object restore', the 'All restore object events' checkbox is checked.

The screenshot shows the AWS CloudFormation console with the 'CloudFormation' service selected. In the 'Event Types' section, the 'Object restore' section is highlighted with a red box. It contains the following event types:

- Delete marker created
s3:ObjectRemoved:DeleteMarkerCreated
- All restore object events
s3:ObjectRestore*
- Restore initiated
s3:ObjectRestore:Post
- Restore completed
s3:ObjectRestore:Completed
- Restored object expired
s3:ObjectRestore:Delete

Below this, there are sections for 'Object ACL' and 'Object tagging', each with its own set of event types.

155)choose sns as lambda function → select choose from your sns topics → save changes.

The screenshot shows the AWS CloudFormation console with the 'CloudFormation' service selected. In the 'Destination' section, the 'SNS topic' option is selected and highlighted with a red box. The 'Specify SNS topic' dropdown also has a red box around it, showing 'Choose from your SNS topics' selected. At the bottom right, the 'Save changes' button is highlighted with a red box.

156) now go to hosted page → give credentials → click update database.

Not secure | adminloaddbc-1075133596.us-east-1.elb.amazonaws.com

Employee Database

GET EMPLOYEE
INFORMATION

Employee ID:

First Name:

Last Name:

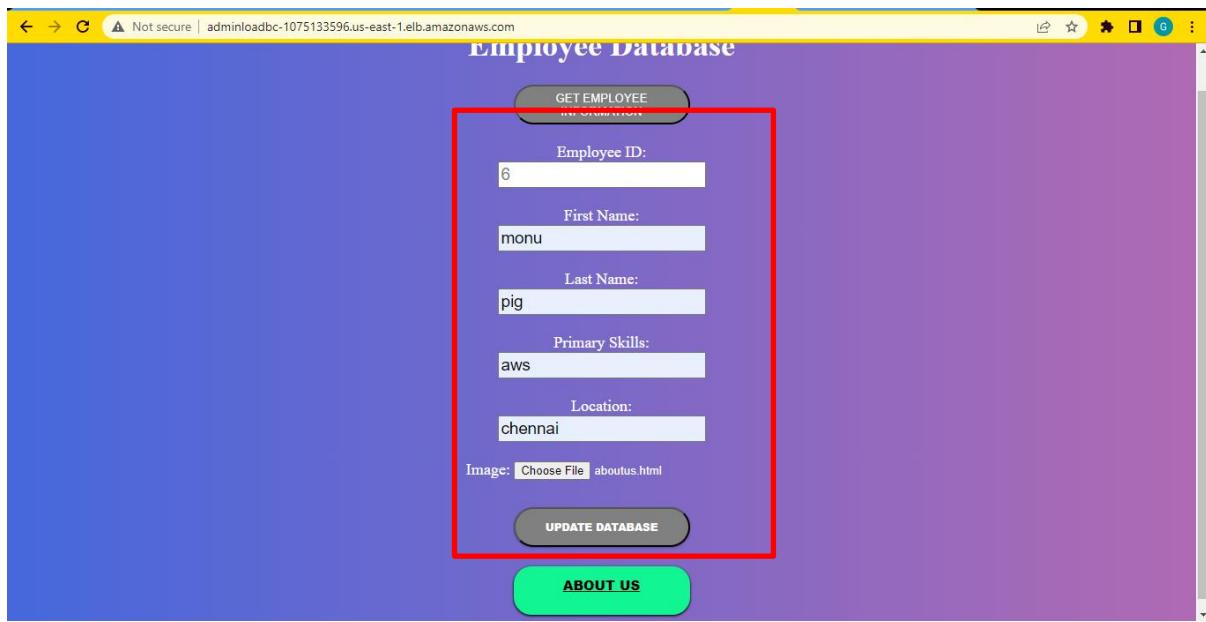
Primary Skills:

Location:

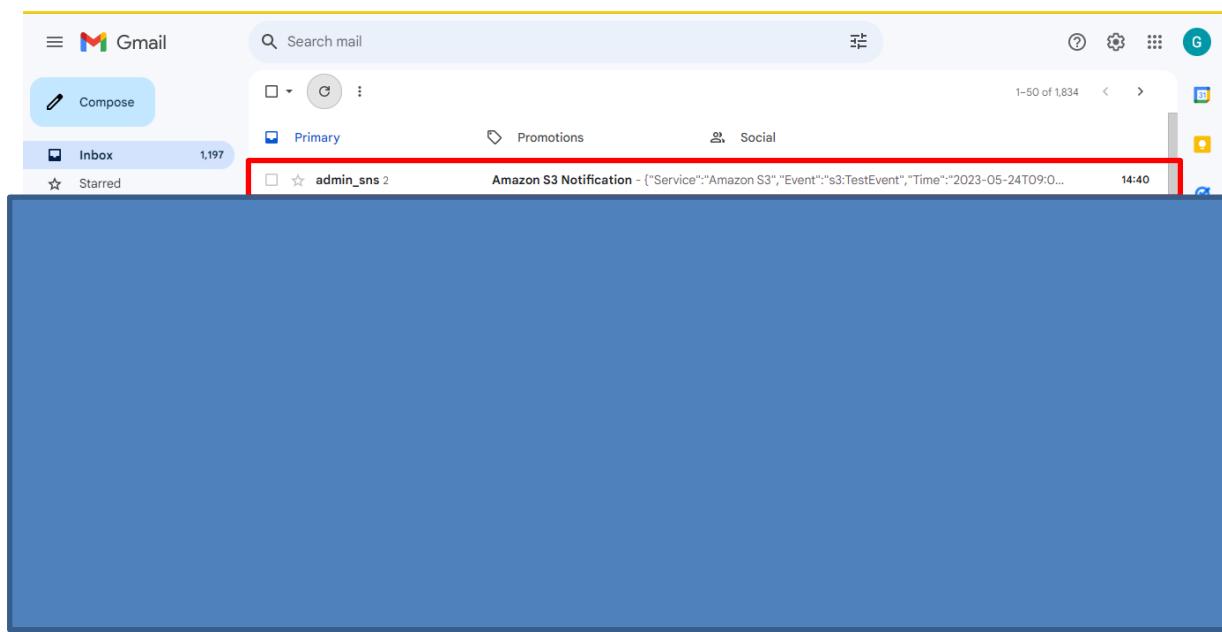
Image: aboutus.html

UPDATE DATABASE

ABOUT US



157) once to click update database → you will receive mail as notification.



The screenshot shows a Gmail inbox with a red box highlighting an email from "admin sns <no-reply@sns.amazonaws.com>" titled "Amazon S3 Notification". The email body contains a message about an S3 event and a unsubscribe link. Below it is another email from the same sender with a timestamp of 14:40 (5 minutes ago). The sidebar on the left shows the inbox has 1,196 messages. At the bottom, there's a prompt to enable desktop notifications for Gmail.

Compose

Inbox 1,196

Starred

Snoozed

Sent

Drafts

More

Labels

Amazon S3 Notification [Inbox](#)

admin sns <no-reply@sns.amazonaws.com>
to me ▾

14:38 (6 minutes ago) [Star](#) [Reply](#) [Forward](#) [Print](#) [Copy](#)

{"Service":"Amazon S3","Event":"s3:TestEvent","Time":"2023-05-24T09:08:44.686Z","Bucket":"admins3001","RequestId":"PCMEE2DVAX4AT4BM","HostId":"SA/bjJbP9CL5rJBiBzJPuRUFJKcEasPv+zVIBIkY4PdNb8TJ/c2NGANqbKFPjNGayH28TJMjMY="}

--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.us-east-1.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-east-1:258506467594:admin_sns:fc9626f-c1e8-468e-aa7f-069b595ecd9bb&Endpoint=gayathrikinfootech@gmail.com

Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at <https://aws.amazon.com/support>

14:40 (5 minutes ago) [Star](#) [Reply](#) [Forward](#) [Print](#) [Copy](#)

admin sns <no-reply@sns.amazonaws.com>
to me ▾

Records [{"eventVersion": "2.1", "eventSource": "aws:s3", "awsRegion": "us-east-1", "eventTime": "2023-05-24T09:10:08.596Z", "eventName": "ObjectCreated:Put", "userIdentity": {"principalId": "AWS AIDAIAATAZTOMD5FZTPKC"}, "requestParameters": {"sourceIPAddress": "2600:1f18:4e10:a801:7d5b:421f:d35d:fec0"}, "responseElements": {"x-amz-request-id": "ZC3KR6GM9AP1W27S", "x-amz-id-2": "ZCqoeOHy9aa7KTKnw1Tl8t8+jP0AFx9vSFaYy7NqskJA+InBew4phQKOEBy4J10n/plw5yyNEc", "s3": {"s3SchemaVersion": "1.0", "configurationId": "adminevent", "bucket": {"key": "AWSLogs/258506467594/elasticroadbalancing/us-east-1/_elasticroadbalancing/us-east-1_app.adminloadbc.2f0bd12ded39d8f12_20230524T0910Z.44.213.51.184_5z"}}, "versionId": "12X0D3B4DU6NWS"}], "arn": "arn:aws:s3:::admins3001"}, "object": {"Key": "AWSLogs/258506467594/elasticroadbalancing/us-east-1/_elasticroadbalancing/us-east-1_app.adminloadbc.2f0bd12ded39d8f12_20230524T0910Z.44.213.51.184_5z"}]

Enable desktop notifications for Gmail. [OK](#) [No, thanks](#)