

NAME: DENDETI JAGADEESH

BATCH NO: 127

MAIL.ID: jagadeeshdendeti2001@gmail.com

DATE: 06-08-2024

TRAINER: Mr. Madhukar

PROJECT – 1

Project:



The above architecture is the “Architecting 3 Tier Architecture on AWS”.

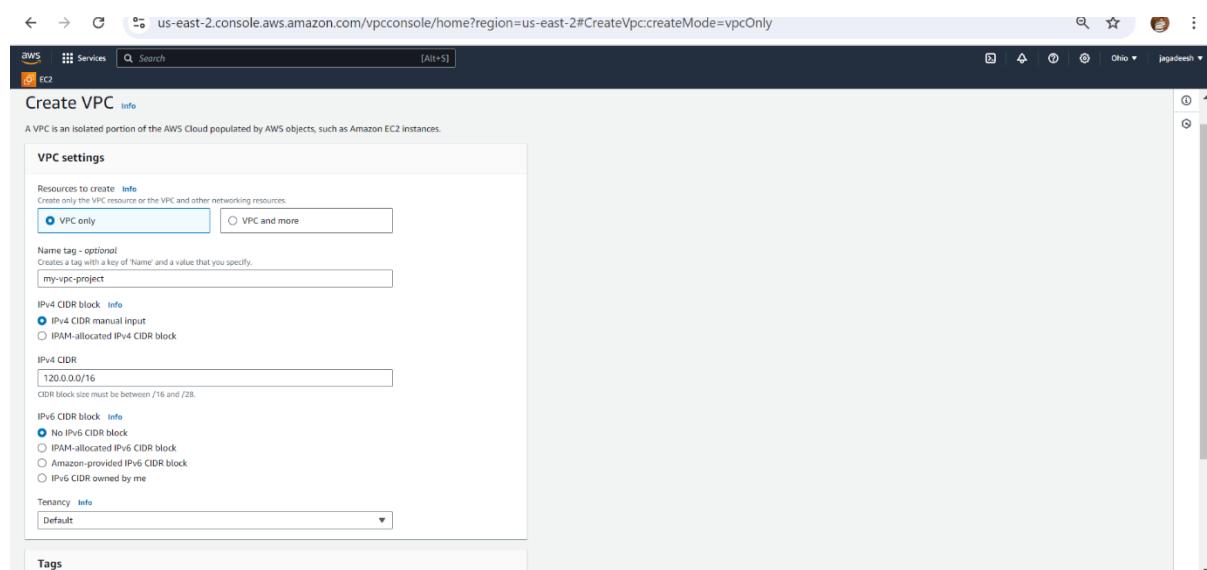
- First tier of our architecture is a web tier. It consists of 2 public subnets in separate availability zones, and an Auto scaling group with launch template and security group.
- The second tier is an Application tier. This tier will consist of 2 Private, an Autoscaling group with launch a template and same security group used in above web tier.
- The third tier is a Database tier. This tier will have an RDS(relational database service) in 2 Private subnets and an same security group used in above both the tiers.

Creating the above architecture we have to follow the following steps:

1. Create VPC, Subnets – 6, Internet gate way – 1, Route tables – 2, Nat gate way – 1.
2. Launch an EC2 instance.
3. Create Load Balancer
4. Create an AMI (image).
5. Create Autoscaling group, Create launch template.
6. Create Subnet group.
7. Create Database (RDS).
8. Establish connection.

Step: 1

Create VPC and its components :



- Go to VPC dashboard click on create VPC.
- Click on VPC only and name tag as my-vpc-project1
- Give IPV4 CIDR (classless inter domain routing) as 120.0.0.0/16
- Click on VPC, it is created

The screenshot shows the AWS VPC dashboard with the URL us-east-2.console.aws.amazon.com/vpcconsole/home?region=us-east-2#vpcs. The dashboard lists two VPCs:

Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR	DHCP option set	Main route table	Main gateway
mydefault1	vpc-0bbb34665dfa11662	Available	172.31.0.0/16	-	dopt-05f1adff871731978c	rtb-0eaafab9f7d5cf01	ad-C
my-vpc-project	vpc-0046e8de11c05f3dd	Available	120.0.0.0/16	-	dopt-05f1adff871731978c	-	-

- Create 6 subnets (2-public, 4-private).
- Create first subnet.
- Click on subnet, click on create subnet, select our VPC (my-project-vpc).
- Give name tag as availability zones

The screenshot shows the 'Create subnet' wizard in the AWS VPC console. The steps are:

- VPC**: Selects the VPC ID: `vpc-092c3300e0df97746 (my-vpc-project1)`.
- Associated VPC CIDRs**: Shows the IPv4 CIDR: `120.0.0.0/16`.
- Subnet settings**: Subnet 1 of 1. Subnet name: `my-subnet-01`.

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs
< > ^ v

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="ny-sub-private"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs
< > ^ v

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-sub-private2"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.

IPv4 subnet CIDR block
 256 IPs
< > ^ v

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-sub-public2"/>

[Add new tag](#)
You can add 49 more tags.
[Remove](#)

[Add new subnet](#)

[Cancel](#) [Create subnet](#)

aws Services Search [Alt+S] EC2

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1b

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
120.0.0.0/16

IPv4 subnet CIDR block
120.0.5.0/16 65,536 IPs
< > ^

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-sub-private3"/>

Add new tag You can add 49 more tags.
Remove

Add new subnet

Cancel **Create subnet**

aws Services Search [Alt+S] EC2

The name can be up to 256 characters long.

Availability Zone [Info](#)
Choose the zone in which your subnet will reside, or let Amazon choose one for you.
US East (N. Virginia) / us-east-1b

IPv4 VPC CIDR block [Info](#)
Choose the VPC's IPv4 CIDR block for the subnet. The subnet's IPv4 CIDR must lie within this block.
120.0.0.0/16

IPv4 subnet CIDR block
120.0.4.0/24 256 IPs
< > ^

Tags - optional

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="my-sub-public2"/>

Add new tag You can add 49 more tags.
Remove

Add new subnet

Cancel **Create subnet**

These are the subnets we created.

← → ⌂ us-east-1.console.aws.amazon.com/vpcconsole/home?region=us-east-1#subnets:

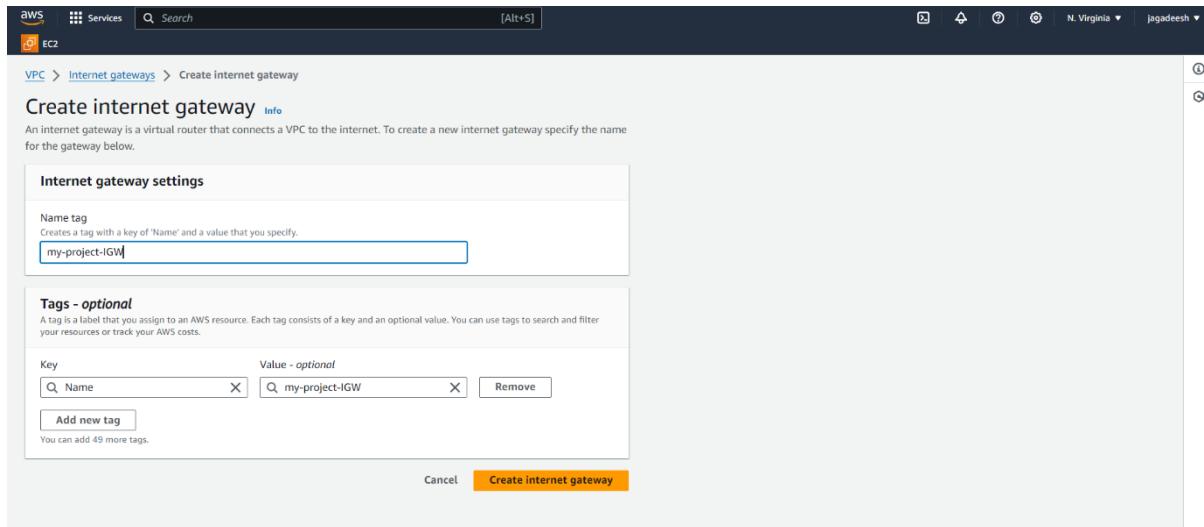
aws Services Search [Alt+S] EC2

VPC dashboard Subnets (12) [Info](#)

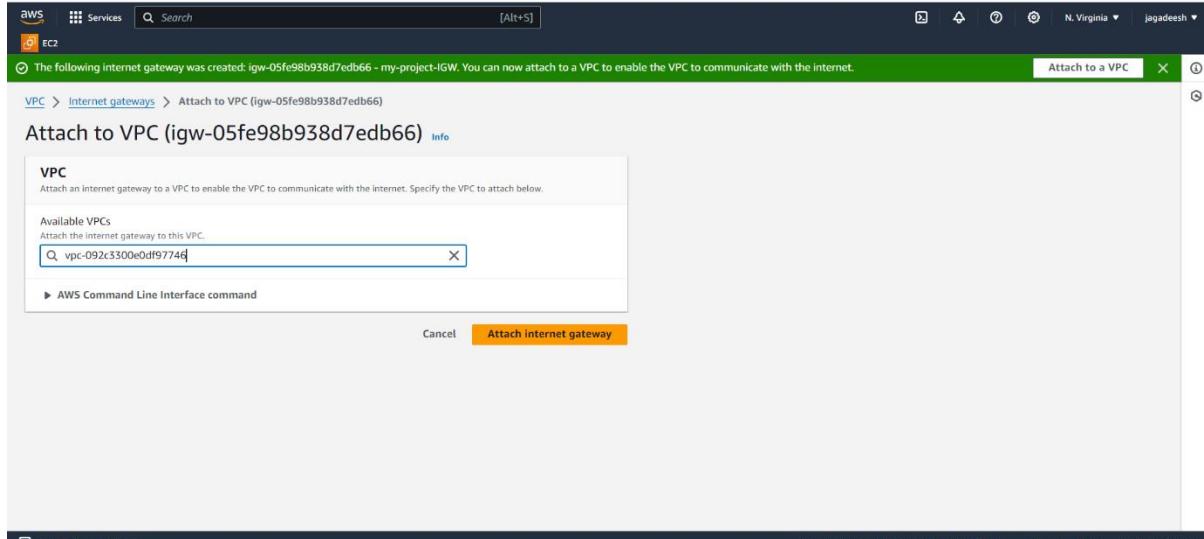
Name	Subnet ID	State	VPC	IPv4 CIDR	IPv6 CIDR
ny-sub-private1	subnet-00d5dd7813c603d64	Available	vpc-092c3300e0df97746 my...	120.0.2.0/24	-
my-sub-private2	subnet-09e873202fd465160	Available	vpc-092c3300e0df97746 my...	120.0.3.0/24	-
my-sub-public1	subnet-07ed75b66d1f6faab1	Available	vpc-092c3300e0df97746 my...	120.0.1.0/24	-
my-sub-private3	subnet-06d5f6077b5dcd8f4	Available	vpc-092c3300e0df97746 my...	120.0.5.0/24	-
-	subnet-09b33e4a94fecb698	Available	vpc-0c083825cf7feee28	172.31.80.0/20	-
-	subnet-086496a708c900d3f	Available	vpc-0c083825cf7feee28	172.31.32.0/20	-
my-sub-private4	subnet-078774e39c9596fb0	Available	vpc-092c3300e0df97746 my...	120.0.6.0/24	-
-	subnet-062e78f6c7ea78858	Available	vpc-0c083825cf7feee28	172.31.48.0/20	-
-	subnet-0e245bf8712da1934	Available	vpc-0c083825cf7feee28	172.31.16.0/20	-
my-sub-public2	subnet-078be4f59c8150e3	Available	vpc-092c3300e0df97746 my...	120.0.4.0/24	-

Subnets: subnet-09b33e4a94fecb698, subnet-086496a708c900d3f, subnet-0ed1463cf949206a, subnet-072f0011274866389, subnet-0e245bf8712da1934, subnet-062e78f6c7ea78858

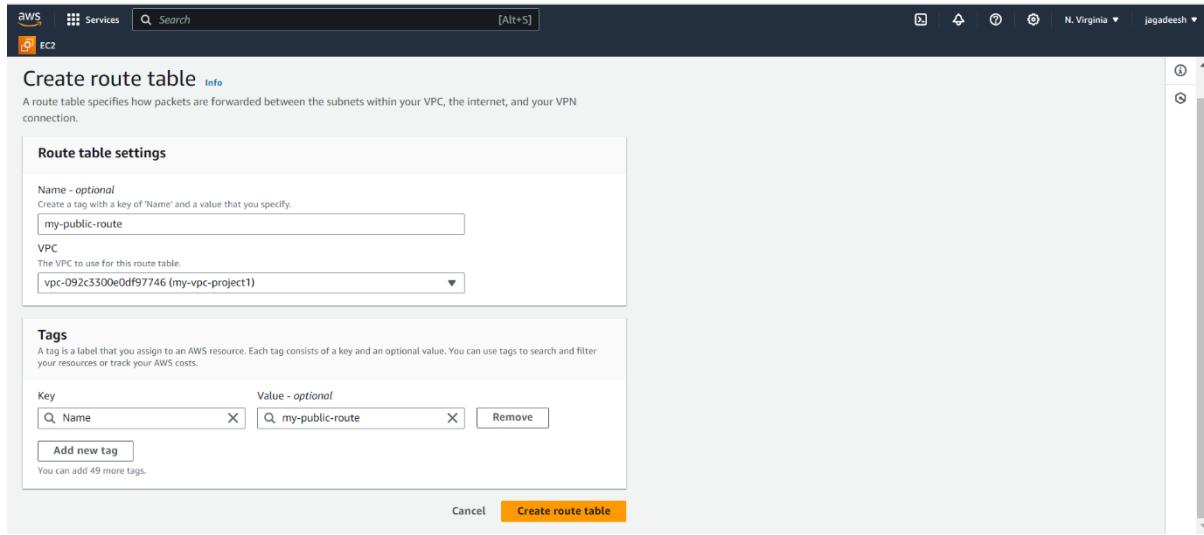
- Create internet gateway name my-project-IGW



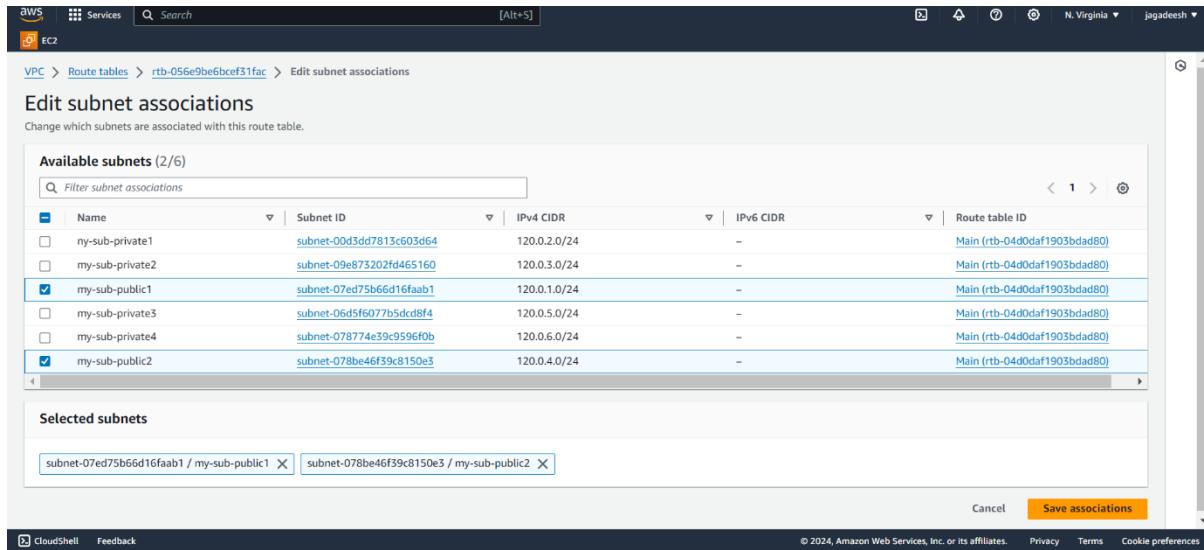
- This igw is attached to VPC.
- Go to actions in internet gate way and click on attach to VPC.
- Select our VPC (my-vpc-project1). Click on attach internet gateway



- Create route table, give name as my-public-route
- Select our VPC (my-vpc-project), create it.



- Click on route table id, open it.
- Go down click on edit subnet association.
- Select both public subnet and click on save association



- G0 actions and click on edit routes .
- Click on add routes give all traffic (0.0.0.0/0) and select our internet gateway, save changes .

Destination

120.0.0.0/16	Target	Status	Propagated
	local	Active	No
<input type="text" value="0.0.0.0/0"/> X	<input type="text" value="local"/> X		
	Internet Gateway	-	No
	<input type="text" value="igw-05fe98b938d7edb66"/> X		

Add route Remove Preview Save changes

- Create private route table name as my-private-route
- Select our VPC, (my-vpc-project1), create it.
- Click on route table id, open it.
- Go down click on edit subnet association.
- Select all private subnet and click on save association

Create route table [Info](#)

A route table specifies how packets are forwarded between the subnets within your VPC, the internet, and your VPN connection.

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 - <i>optional</i>
<input type="text" value="Name"/> X	<input type="text" value="my-private-route"/> X

Add new tag
You can add 49 more tags.

Cancel Create route table

Available subnets (4/6)

Name	Subnet ID	IPv4 CIDR	IPv6 CIDR	Route table ID
my-sub-private1	subnet-00d3dd7813c603d64	120.0.2.0/24	-	Main (rtb-04d0daf1903bdad80)
my-sub-private2	subnet-09e873202fd465160	120.0.3.0/24	-	Main (rtb-04d0daf1903bdad80)
my-sub-public1	subnet-07ed75b66d16faab1	120.0.1.0/24	-	rtb-056e9be6bccef31fac / my-public-ro.
my-sub-private3	subnet-06d5f6077b5dcdf8f4	120.0.5.0/24	-	Main (rtb-04d0daf1903bdad80)
my-sub-private4	subnet-078774e39c9596f0b	120.0.6.0/24	-	Main (rtb-04d0daf1903bdad80)
my-sub-public2	subnet-078be46f39c8150e3	120.0.4.0/24	-	rtb-056e9be6bccef31fac / my-public-ro.

Selected subnets

- subnet-00d3dd7813c603d64 / my-sub-private1
- subnet-09e873202fd465160 / my-sub-private2
- subnet-078774e39c9596f0b / my-sub-private4
- subnet-06d5f6077b5dcdf8f4 / my-sub-private5

Cancel **Save associations**

- Create NAT gateway, give name as my-nat.
- Select public subnet(my-sub-public1).
- Select connectivity type as IPV4.
- Click on allocate Elastic IP.

Create NAT gateway info

A highly available, managed Network Address Translation (NAT) service that instances in private subnets can use to connect to services in other VPCs, on-premises networks, or the internet.

NAT gateway settings

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

The name can be up to 256 characters long.

Subnet
Select a subnet in which to create the NAT gateway.

Connectivity type
Select a connectivity type for the NAT gateway.
 Public
 Private

Elastic IP allocation ID Info
Assign an Elastic IP address to the NAT gateway.

Additional settings Info

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

- Now go to private route and click on actions.
- Click on edit routes and add route.
- Give all traffic (0.0.0.0/0) and select NAT gateway

The screenshot shows the 'Edit routes' page in the AWS VPC console. It lists two routes:

- A route from **Destination** `120.0.0.0/16` to **Target** `local` with **Status** `Active`.
- A route from **Destination** `0.0.0.0/0` to **Target** `Internet Gateway` with **Status** `Active`. The target is further expanded to show the specific Internet Gateway ID `igw-05fe98b938d7edb66`.

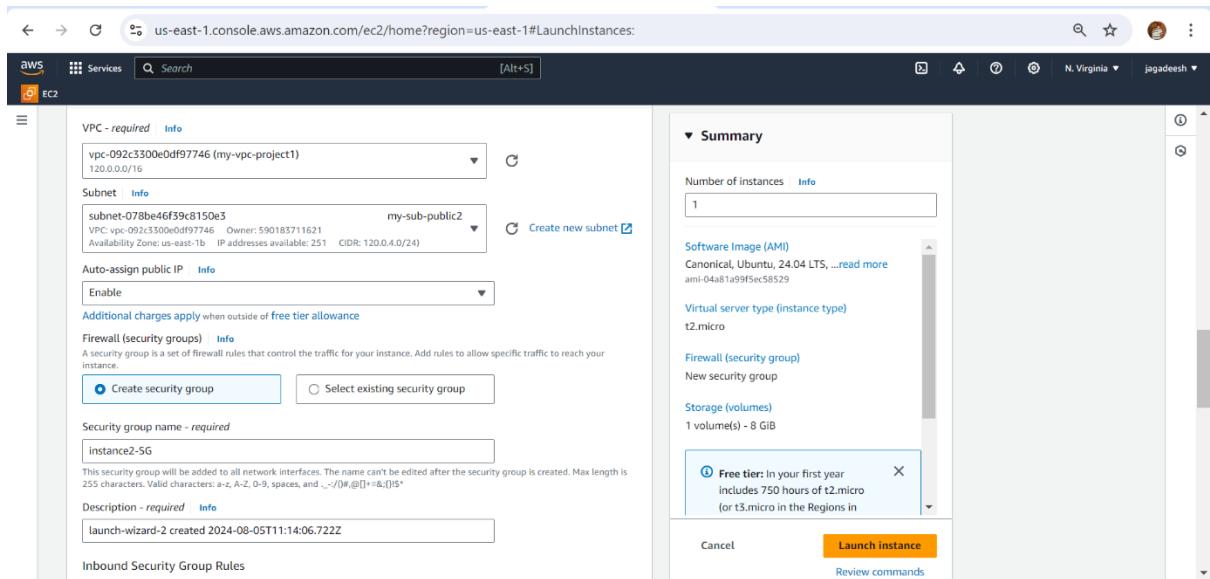
Buttons at the bottom include `Add route`, `Cancel`, `Preview`, and `Save changes`.

Step2: Launch an EC2 instance.

- Go to EC2 dashboard click on launch instance.
- Name as and select ami as ubuntu
- Instance type as t2.micro and key pair as project.
- Click on edit network settings, select our VPC and public subnet.
- Auto assign IP enable and create a security group as project-sg.
- Launch the instance.

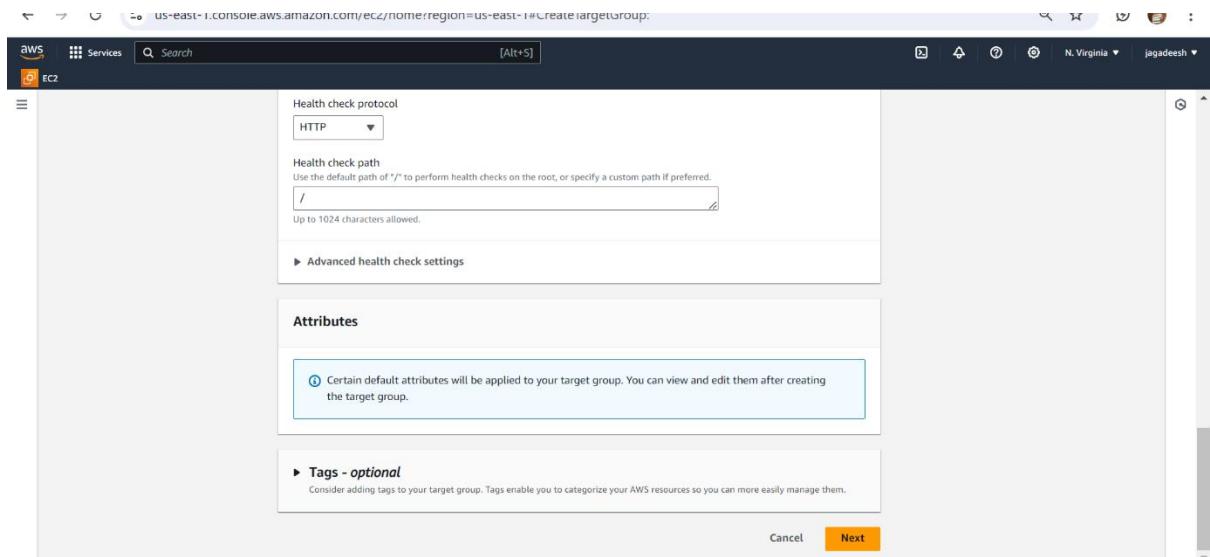
The screenshot shows the 'Launch an instance' wizard in the AWS EC2 console. The configuration steps are as follows:

- Name and tags**: The name is set to `e.g. My Web Server`.
- Application and OS Images (Amazon Machine Image)**: An AMI is selected: `Amazon Linux 2023 AMI 2023.5.2...read more`.
- Summary** section:
 - Number of instances**: 1
 - Software Image (AMI)**: `Amazon Linux 2023 AMI 2023.5.2...read more`
 - Virtual server type (instance type)**: `t2.micro`
 - Firewall (security group)**: New security group
 - Storage (volumes)**: 1 volume(s) - 8 GiB
 - Free tier**: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in)
- Launch instance** button is visible at the bottom right.



Step 3:

- Now we create 2 LoadBalancer
- One is public instances and another one is private instances
- Then create target groups and attached the instance and check the
- Instance ssh -i copied and paste on the server
- Then next apt update -y apt install -y cd /var/www/html
- Systemctl restart apache2
- We take another instance like same procedure



AWS Services Search [Alt+S] N. Virginia jagadeesh

EC2 Ports for the selected instances
Ports for routing traffic to the selected instances.
80
1-65535 (separate multiple ports with commas)

Include as pending below

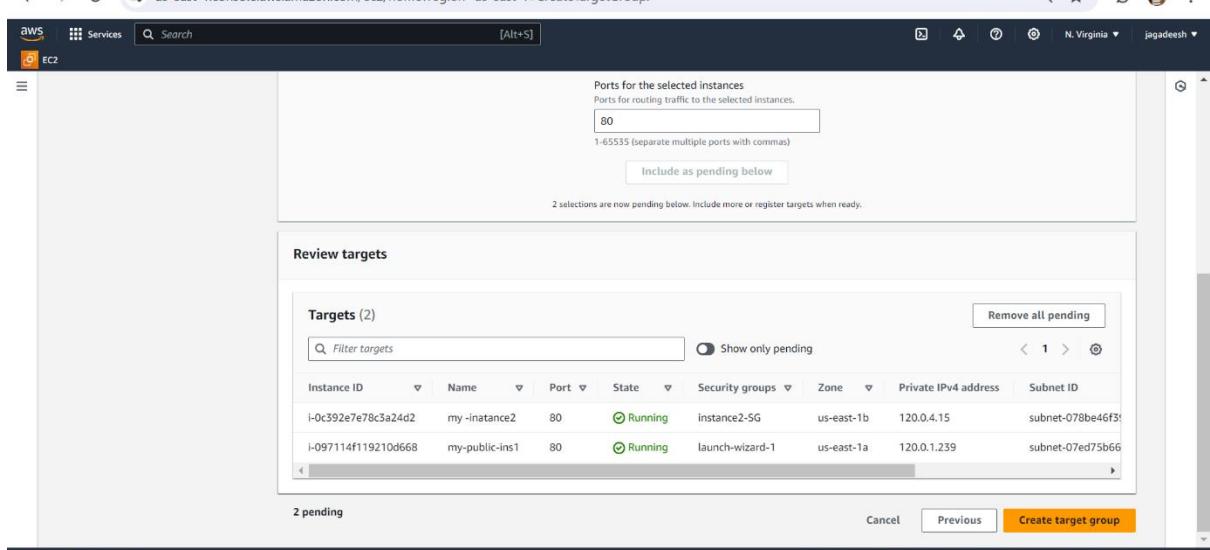
2 selections are now pending below. Include more or register targets when ready.

Review targets

Targets (2)

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID
i-0c392e7e78c3a24d2	my-inatance2	80	Running	instance2-5G	us-east-1b	120.0.4.15	subnet-078be46f3
i-097114f119210d668	my-public-ins1	80	Running	launch-wizard-1	us-east-1a	120.0.1.239	subnet-07ed75b66

2 pending Cancel Previous Create target group



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

EC2 Services Search [Alt+S] N. Virginia jagadeesh

EC2 Load balancers

Load balancers

Elastic Load Balancing scales your load balancer capacity automatically in response to changes in incoming traffic.

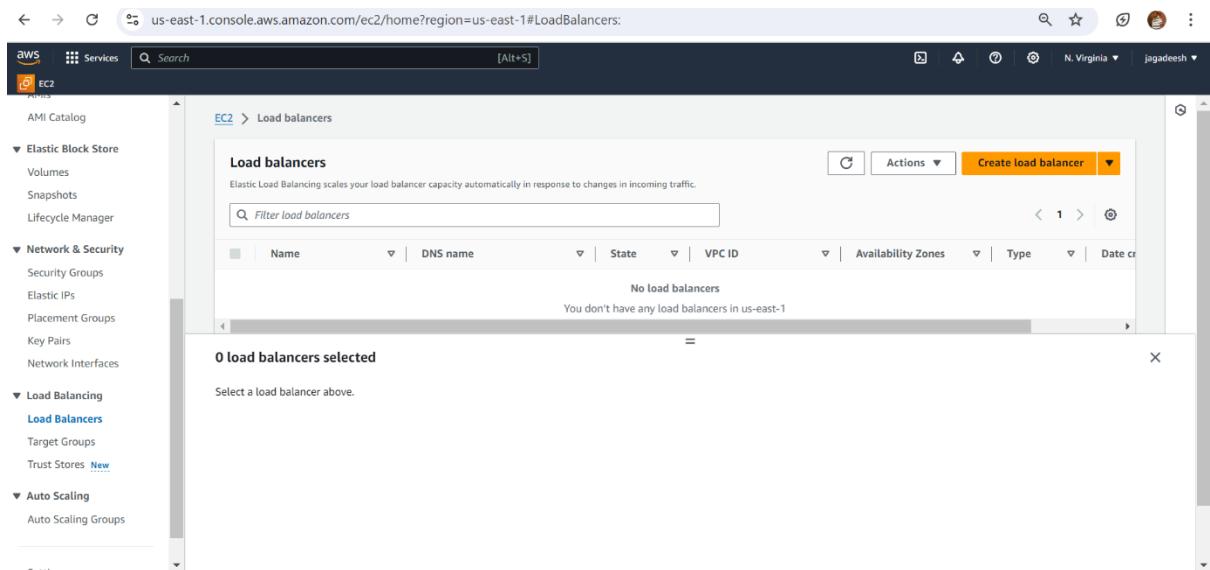
Name	DNS name	State	VPC ID	Availability Zones	Type	Date created
No load balancers You don't have any load balancers in us-east-1						

0 load balancers selected

Select a load balancer above.

Load Balancers Target Groups Trust Stores New

Auto Scaling Auto Scaling Groups



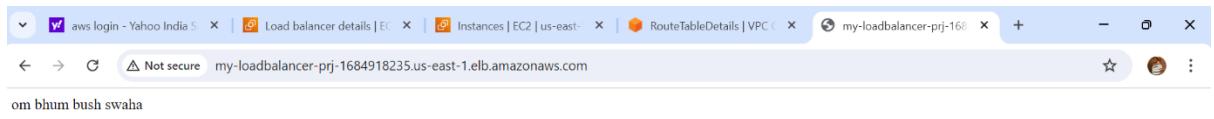
The screenshot shows the AWS CloudFront Create ALB Wizard - Step 3: Listener Configuration. The top navigation bar includes the AWS logo, Services, a search bar, and account information for N. Virginia and jagadeesh. The main content area has two sections:

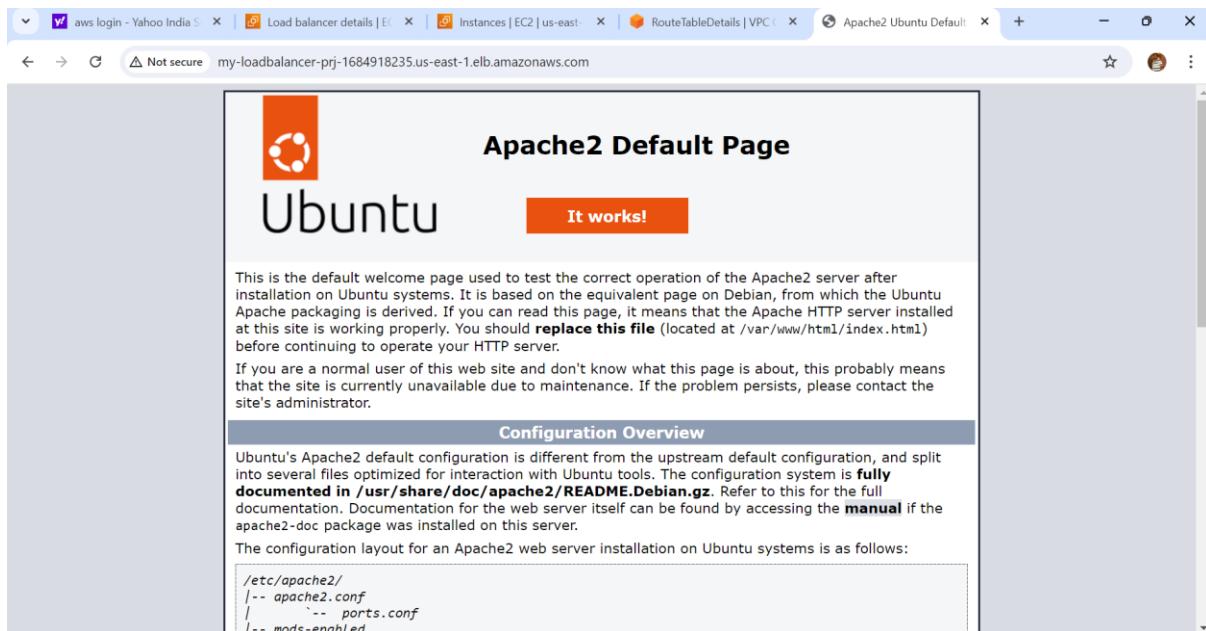
- Security groups**: A dropdown menu titled "Select up to 5 security groups" contains one item: "instance2-SG".
- Listeners and routing**: A table for "Listener HTTP:80" shows a single rule: "Forward to my-project-TG" (Target type: Instance, IPv4). The "Default action" is set to "HTTP".

At the bottom, there are links for CloudShell, Feedback, and a footer with copyright information for 2024, Amazon Web Services, Inc. or its affiliates.

```
root@ip-120-0-22-8:~# cd /var/www/html
root@ip-120-0-22-8:/var/www/html# ls
index.html
root@ip-120-0-22-8:/var/www/html# rm index.html
root@ip-120-0-22-8:/var/www/html#
```

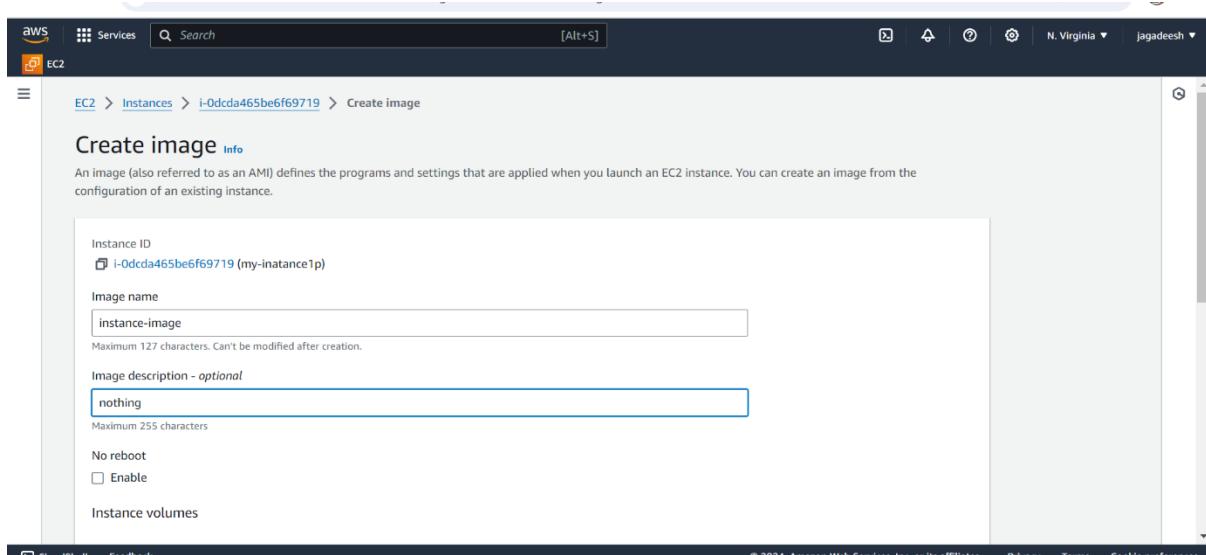
```
root@ip-120-0-22-8:~/var/www/html
root@ip-120-0-22-8:/var/www/html# ls
index.html
root@ip-120-0-22-8:/var/www/html# rm index.html
root@ip-120-0-22-8:/var/www/html# vi index.html
root@ip-120-0-22-8:/var/www/html# systemctl restart apache2
```

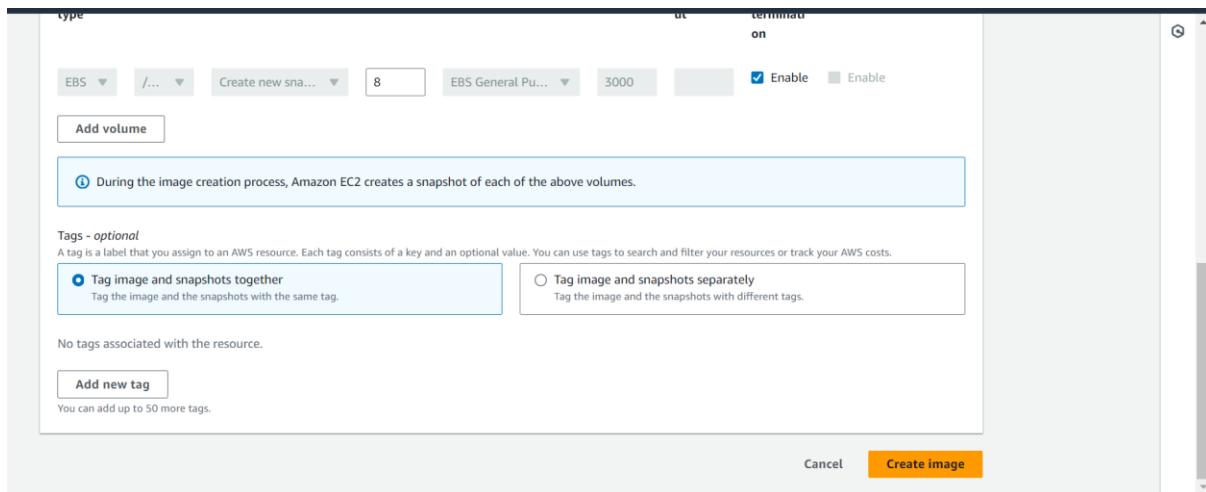




Step 4: Create an AMI (image)

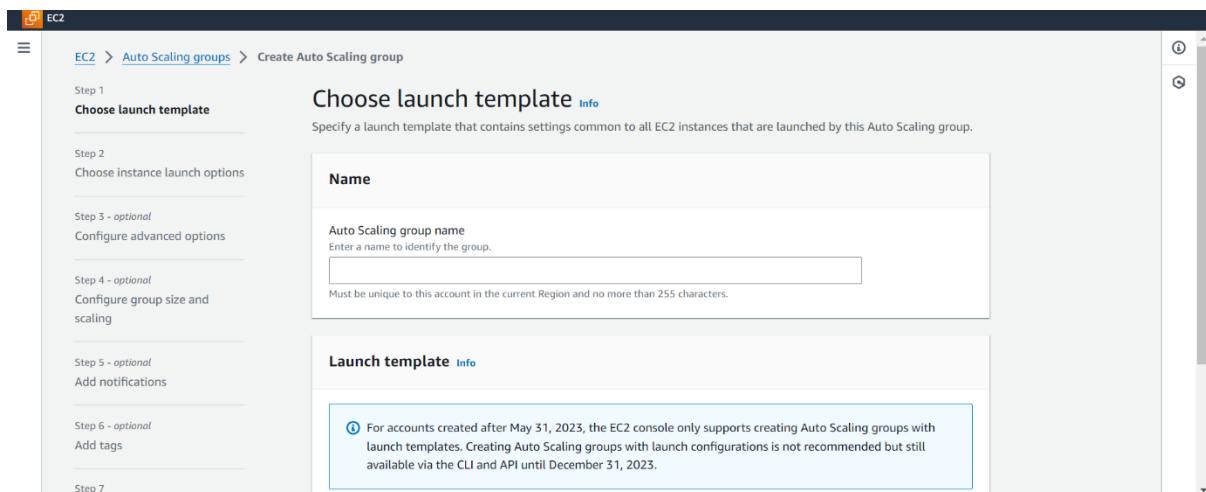
- After running the instance, click on actions.
- Click on image and templates and click on create image.
- Give image name as my-image.
- Wait until the image is available.





Step 5: Create Autoscaling group.

- For creating autoscaling group we need to create an launch template.
- After available of image. Click on create a launch template.
- Template name as my-public-template, description as nothing.
- Select AMI's as share with me, select my-image.
- Instance type as t2.micro and key pair as project.
- Select existing security group (my-instance-SG) which is used to launch an EC2 instance
- Now click on create launch template.



- Open autoscaling group.
- Click on create autoscaling group.
- Give name as public-AS
- Choose the created launch template (my-public-tm) and click on next.
- Select our VPC (my-project-vpc), and both public subnets.
- Click on next and click on no load balancer
- Give desired capacity as 2 in sizing desired capacity min – 2 and max – 3 and click on next.
- Click on add notification, give name as my-topic and give email.
- Click on next and click on create auto scaling group.

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

aws Services Search [Alt+S]

EC2

Version
Default (1) C

Create a launch template version [\[\]](#)

Description	Launch template my-public-lt lt-0d3545aa5337076469	Instance type t2.micro
AMI ID	Security groups	Request Spot Instances No
ami-04a81a99f5ec58529	-	Security group IDs sg-0501a4088c3a931f2
Additional details		
Storage (volumes)	Date created Tue Aug 06 2024 16:06:15 GMT+0530 (India Standard Time)	

Cancel **Next**

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

aws Services Search [Alt+S]

EC2

Step 5 - optional
[Add notifications](#)

Step 6 - optional
[Add tags](#)

Step 7
[Review](#)

Instance type
t2.micro

Network [info](#)

us-east-1a | subnet-0ba2345f0b0d1093 (my-sub-private2)
120.0.44.0/24

us-east-1a | subnet-0c90608f221d0163b (my-ssub-private1)
120.0.33.0/24

us-east-1a | subnet-0e296972ec0dd6df4 (my-sub-public1)
120.0.22.0/24

us-east-1b | subnet-07a97adff0bfddc681 (my-sub-public2)
120.0.55.0/24

us-east-1b | subnet-056a20764590d3f73 (my-sub-private3)
120.0.66.0/24

us-east-1b | subnet-05534c438dfe05ca2 (my-sub-private4)
120.0.77.0/24

Select Availability Zones and subnets C

us-east-1a | subnet-0e296972ec0dd6df4 (my-sub-public1)
120.0.22.0/24

us-east-1b | subnet-07a97adff0bfddc681 (my-sub-public2)
120.0.55.0/24

Screenshot of the AWS EC2 Auto Scaling wizard Step 3 - optional: Configure advanced options. The "Load balancing" section is shown, with the "Attach to an existing load balancer" option selected. The "Scaling" section below it is also visible.

Configure advanced options - optional

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

Load balancing

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

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

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

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

Attach to an existing load balancer

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

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

Choose from Classic Load Balancers

Screenshot of the AWS EC2 Auto Scaling wizard Step 4 - optional: Configure group size and scaling. The "Desired capacity type" and "Scaling" sections are shown, with the "No scaling policies" option selected.

Auto Scaling group updated successfully

Configure advanced options

Desired capacity type

Choose the unit of measurement for the desired capacity value. vCPUs and Memory(GiB) are only supported for mixed instances groups configured with a set of instance attributes.

Units (number of instances)

Desired capacity

Specify your group size.

2

Scaling

You can resize your Auto Scaling group manually or automatically to meet changes in demand.

Scaling limits

Set limits on how much your desired capacity can be increased or decreased.

Min desired capacity Max desired capacity

2 3

Equal or less than desired capacity Equal or greater than desired capacity

Automatic scaling - optional

Choose whether to use a target tracking policy

You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

No scaling policies

Target tracking scaling policy

Screenshot of the AWS EC2 Auto Scaling wizard Step 5 - optional: Add notifications. The "Add notification" button is highlighted, and the "Next" button is the final step in the wizard.

Add notifications - optional

Send notifications to SNS topics whenever Amazon EC2 Auto Scaling launches or terminates the EC2 instances in your Auto Scaling group.

Add notification

Cancel Skip to review Previous **Next**

- In similar way create another launch template name as private-tm and create auto scaling group name as MY-PRIVATE-AS
- In it select create VPC and give two private subnets

Now create private auto scaling

Auto Scaling group updated successfully

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1 Choose launch template

Step 2 Choose instance launch options

Step 3 - optional Configure advanced options

Step 4 - optional Configure group size and scaling

Step 5 - optional Add notifications

Step 6 - optional Add tags

Step 7 Review

Choose launch template

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group.

Name

Auto Scaling group name
Enter a name to identify the group.
MY-PRIVATE-AS

Must be unique to this account in the current Region and no more than 255 characters.

Launch template

Info

For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.

Launch template
Choose a launch template that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.
private-AS

Auto Scaling group updated successfully

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 6 - optional Add tags

Step 7 Review

Network

Info

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

VPC

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

vpc-01da79ecba341f827 (my-vpc-project1)

120.0.0/16

Create a VPC

Availability Zones and subnets

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

Select Availability Zones and subnets

us-east-1a | subnet-0c90608f221d0163b (my-sub-private1)
120.0.33.0/24

us-east-1b | subnet-056a20764590d3f73 (my-sub-private3)
120.0.66.0/24

Create a subnet

Cancel Skip to review Previous Next

Auto Scaling group updated successfully

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1 Choose launch template

Step 2 Choose instance launch options

Step 3 - optional Configure advanced options

Step 4 - optional Configure group size and scaling

Step 5 - optional Add notifications

Step 6 - optional Add tags

Step 7 Review

Configure advanced options - optional

Info

Integrate your Auto Scaling group with other services to distribute network traffic across multiple servers using a load balancer or to establish service-to-service communications using VPC Lattice. You can also set options that give you more control over health check replacements and monitoring.

Load balancing

Info

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

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

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

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

Attach to a new load balancer

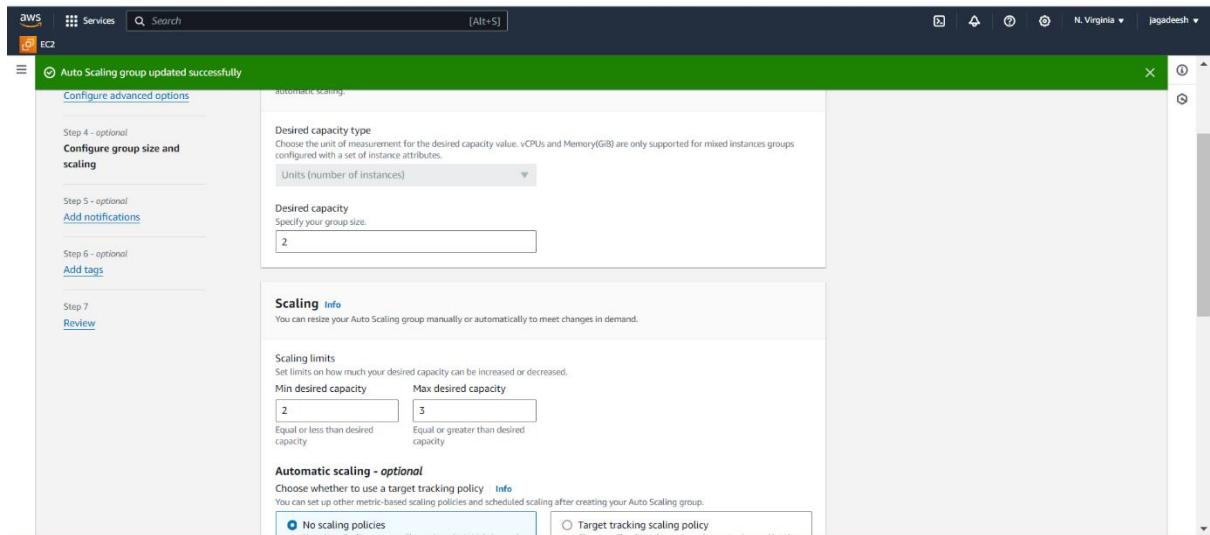
Info

Define a new load balancer to create attachment to this Auto Scaling group.

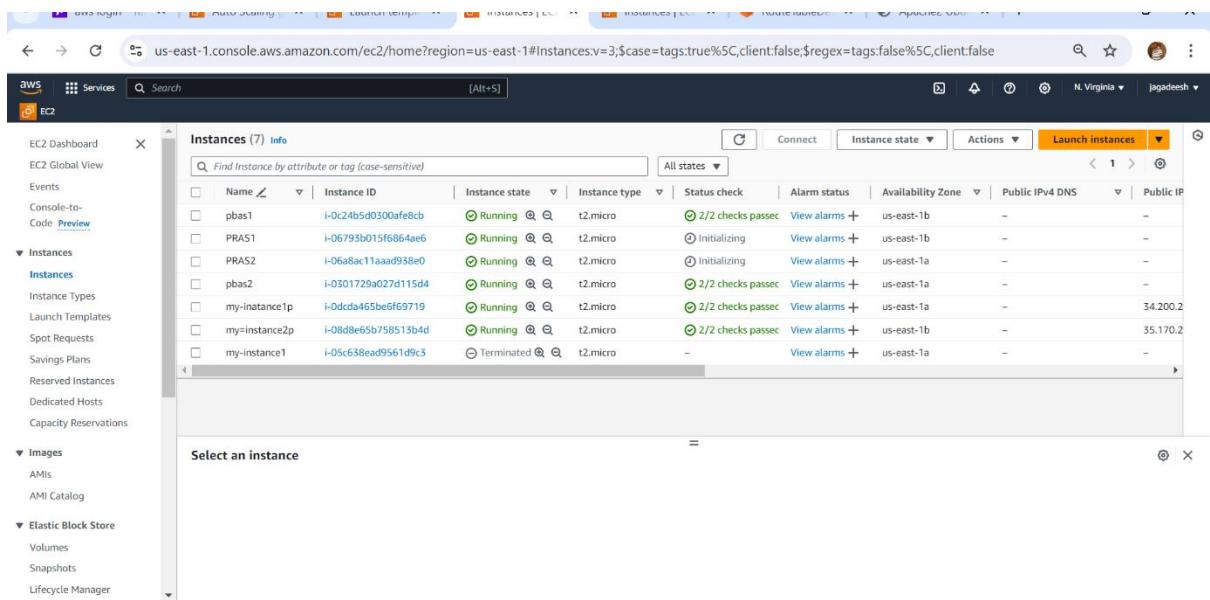
Load balancer type

Info

Choose from the load balancer types offered below. Type selection cannot be changed after the load balancer is created. If you need a different type of load balancer than those offered here, visit the Load Balancing console.



- After creating autoscaling group we can get the four extra servers from both public and private auto scaling group.



Step 6: Create subnet group

- Give name as my-subnet-grp and description nothing.
- Select created VPC.
- Give availability zones and select private subnets from each zone.
- Create the DB subnet group.

The screenshot shows the AWS RDS Subnet groups page. The left sidebar has links for Dashboard, Databases, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups (selected), Parameter groups, Option groups, Custom engine versions, Zero-ETL integrations, Events, and Event subscriptions. The main content area shows a table titled "Subnet groups (0)". It has columns for Name, Description, Status, and VPC. A message at the top says "No db subnet groups" and "You don't have any db subnet groups." A "Create DB subnet group" button is located at the bottom right of the table area.

The screenshot shows the "Create DB subnet group" wizard, Step 1: Subnet selection. It asks to choose Availability Zones and Subnets. Under "Availability Zones", "us-east-1b" and "us-east-1a" are selected. Under "Subnets", "subnet-0ba23435f0b0d1093 (120.0.44.0/24)" and "subnet-05534c458dfe05ca2 (120.0.77.0/24)" are selected. A note says "For Multi-AZ DB clusters, you must select 3 subnets in 3 different Availability Zones." The "Subnets selected (2)" section shows the two chosen subnets with their details. At the bottom are "Cancel" and "Create" buttons.

Screenshot of the AWS EC2 Subnet Selection dialog.

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone ▾
us-east-1b X us-east-1a X

Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets ▾
subnet-0ba23435f0b0d1093 (120.0.44.0/24) X
subnet-05534c438dfe05ca2 (120.0.77.0/24) X

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

Subnets selected (2)

Availability zone	Subnet ID	CIDR block
us-east-1a	subnet-0ba23435f0b0d1093	120.0.44.0/24
us-east-1b	subnet-05534c438dfe05ca2	120.0.77.0/24

Cancel Create

us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#create-db-subnet-group:

Screenshot of the AWS RDS Add Subnets dialog.

Add subnets

Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone ▾
us-east-1b X us-east-1a X

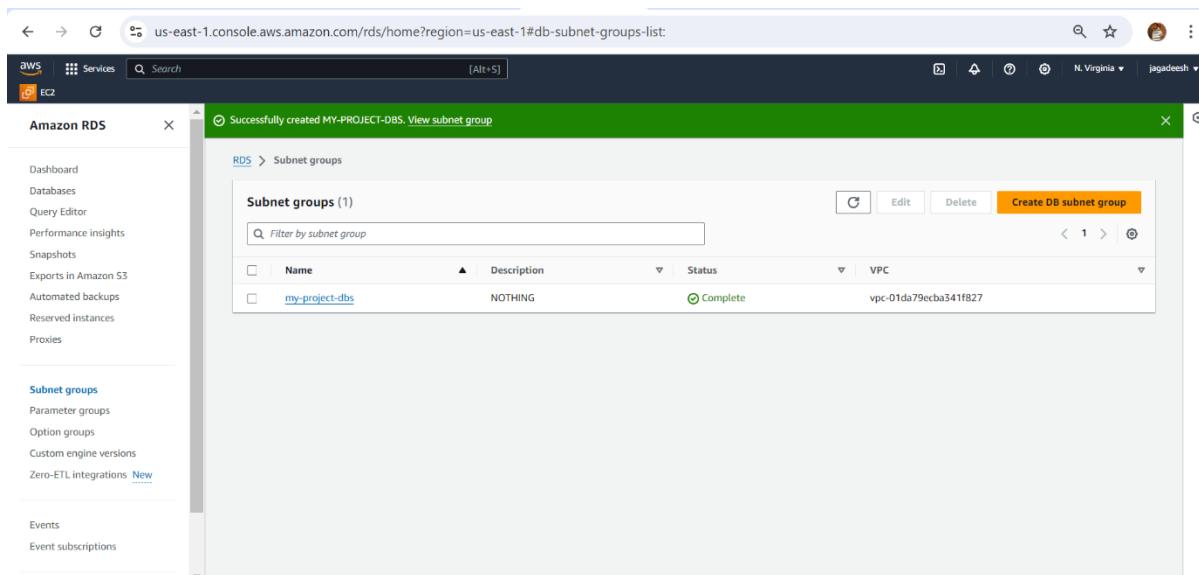
Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets ▾

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

Subnets selected (0)

Availability zone	Subnet ID	CIDR block
No subnets added to this group		

Cancel Create



Step 7: Create Database(RDS)

- Click on create database, select standard create, select engine type as MySQL.
- Select templates as production and select multi-AZ DB cluster.
- Select on self-managed, give password and confirm the password.
- Select memory optimized class.
- In connectivity, click on Don't connect to the EC2 compute resource and select created vpc (my-vpc-project1).
- Select subnet group (my-project-dbs) and give public access as yes.
- Choose existing security group (instance1-SG).
- Go to VPC dashboard, click on VPC, click on actions, go to edit VPC settings and click on the enable DNS hostnames.
- Create the database.

Standard create
You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create
Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

Aurora (MySQL Compatible)



Aurora (PostgreSQL Compatible)



MySQL



MariaDB



PostgreSQL



Oracle

ORACLE®

Enter a name for your DB cluster. The name must be unique across all DB clusters owned by your AWS account in the current AWS Region.
database-1

The DB cluster identifier is case-insensitive, but is stored as all lowercase (as in "mydbcluster"). 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 cluster.

admin

1 to 16 alphanumeric characters. The first character must be a letter.

Credentials management

You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - *most secure*
RDS generates a password for you and manages it throughout its lifecycle using AWS Secret Manager.

Self managed
Create your own password or have RDS create a password that you manage.

Auto generate password

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

Master password [Info](#)

Minimum constraints: At least 8 printable ASCII characters. Can't contain any of the following symbols: / \ * @

Confirm master password [Info](#)

MySQL

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

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

7 Subnets, 3 Availability Zones

Only VPCs with a corresponding DB subnet group are listed.

 After a database is created, you can't change its VPC.

DB subnet group

Choose the DB subnet group. The DB subnet group defines which subnets and IP ranges the DB cluster can use in the VPC that you selected.

my-project-dbs

3 Subnets, 3 Availability Zones

Public access

Yes

RDS assigns a public IP address to the cluster. Amazon EC2 instances and other resources outside of the VPC can connect to your cluster. Resources inside the VPC can also connect to the cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

No

RDS doesn't assign a public IP address to the cluster. Only Amazon EC2 instances and other resources inside the VPC can connect to your cluster. Choose one or more VPC security groups that specify which resources can connect to the cluster.

VPC security group (firewall)

Choose one or more VPC security groups to allow access to your database. Make sure that the security group rules allow the appropriate incoming traffic.

Choose existing

Choose existing VPC security groups

Create new

Create new VPC security group

Existing VPC security groups

Choose one or more options

my-instance1-SG 

► Additional configuration

Enhanced Monitoring

► Additional configuration

Database options, encryption turned on, backup turned on, backtrack turned off, maintenance, CloudWatch Logs, delete protection turned on.

Estimated Monthly costs

DB instance	427.05 USD
Storage	150.00 USD
Provisioned IOPS	900.00 USD
Total	1477.05 USD

This billing estimate is based on on-demand usage as described in [Amazon RDS Pricing](#). Estimate does not include costs for backup storage, IOs (if applicable), or data transfer.

Estimate your monthly costs for the DB Instance using the [AWS Simple Monthly Calculator](#).

 You are responsible for ensuring that you have all of the necessary rights for any third-party products or services that you use with AWS services.

Cancel

Create database

Amazon RDS

Databases (4)

DB identifier	Status	Role	Engine	Region & AZ	Size	Recommendation
database-1	Available	Multi-AZ DB cluster	MySQL Community	ap-south-1	3 instances	
database-1-instance-1	Available	Writer instance	MySQL Community	ap-south-1b	db.r6gd.large	
database-1-instance-2	Available	Reader instance	MySQL Community	ap-south-1c	db.r6gd.large	
database-1-instance-3	Available	Reader instance	MySQL Community	ap-south-1a	db.r6gd.large	

EC2

Successfully created database database-127

Set up EC2 connection

Select EC2 instance

Database: database-127

EC2 instance: i-0ddca465be6f69719 my-instance1p us-east-1a

Create EC2 instance

Cancel Continue

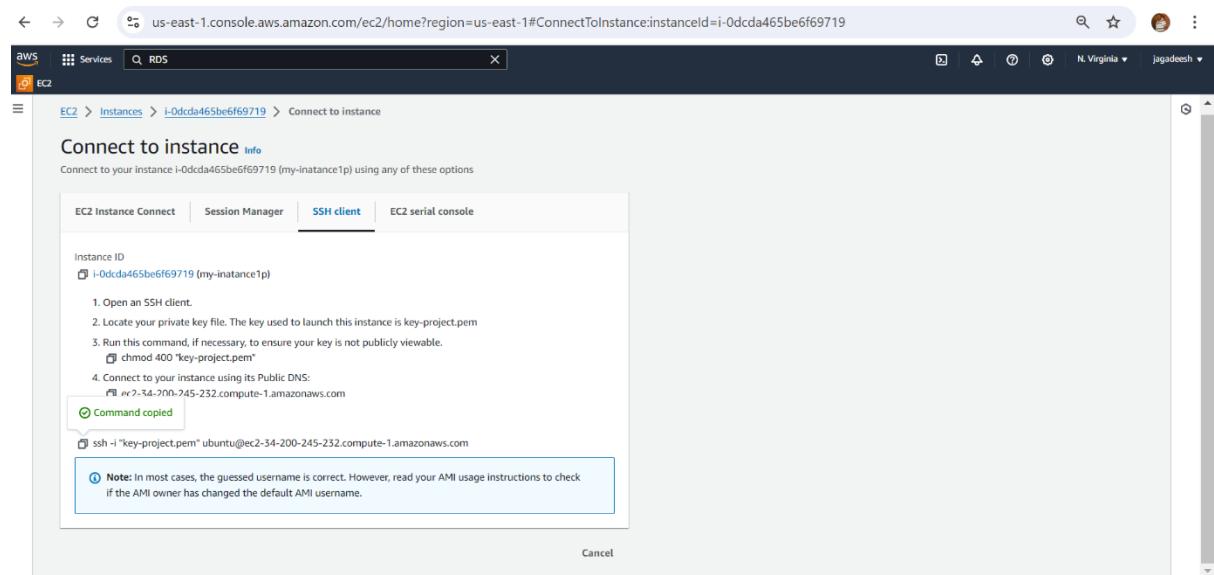


Step8: Establish connection

- Go to EC2 instance, click on instance1-pub, open it click on connect and again click on connect.

Give commands as:

1. sudo -i (convert from normal user to root user).
2. apt update -y
3. sudo apt install mysql-server (to install mysql)



```
oot@ip-120-0-22-8:~# apt update -y
get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
get:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Fetched 126 kB in 1s (222 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
2 packages can be upgraded. Run 'apt list --upgradable' to see them.
oot@ip-120-0-22-8:~# apt install mysql-server
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcgifast-perl libcgipm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7t64 libfcgi-bin
  libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl
  libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64 libtimedate-perl
  liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common
  mysql-server-8.0 mysql-server-core-8.0
Suggested packages:
  libdata-dump-perl libipc-sharedcache-perl libio-compress-brotli-perl libbusiness-isbn-perl libregexp-ipv6-perl
  libwww-perl mailx tinyca
The following NEW packages will be installed:
  libcgifast-perl libcgipm-perl libclone-perl libencode-locale-perl libevent-pthreads-2.1-7t64 libfcgi-bin
  libfcgi-perl libfcgi0t64 libhtml-parser-perl libhtml-tagset-perl libhtml-template-perl libhttp-date-perl
  libhttp-message-perl libio-html-perl liblwp-mediatypes-perl libmecab2 libprotobuf-lite32t64 libtimedate-perl
  liburi-perl mecab-ipadic mecab-ipadic-utf8 mecab-utils mysql-client-8.0 mysql-client-core-8.0 mysql-common
  mysql-server mysql-server-8.0 mysql-server-core-8.0
Upgraded, 28 newly installed, 0 to remove and 42 not upgraded.
Need to get 29.6 MB of archives.
After this operation, 242 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

```
root@ip-120-0-22-8:~#
root@ip-120-0-22-8:~# mysql -h database-127.cluster-clog0g8e0oe8.us-east-1.rds.amazonaws.com -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 84
Server version: 8.0.35 Source distribution

Copyright (c) 2000, 2024, 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> |
```

Use commands like:

```
CREATE TABLE Persons (
    ID int NOT NULL
    LastName varchar(255) NOT NULL,
    FirstName varchar(255),
    Age int,
    PRIMARY KEY (ID)
);
```

- we can insert data into that table using this command
- `INSERT INTO table_name (column1, column2, column3, ...)`
`VALUES (value1, value2, value3, ...);`
- `show tables;` (to show tables in that DB)
- `Select * from Table name`

```
+-----+
| Database          |
+-----+
| information_schema |
| mysql             |
| performance_schema |
| sys               |
+-----+
4 rows in set (0.00 sec)

mysql> create database jagadeesh;
Query OK, 1 row affected (0.04 sec)

mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use jagadeesh;
Database changed
mysql> CREATE TABLE Persons (
    ->     ID int NOT NULL,
    ->     LastName varchar(255) NOT NULL,
    ->     FirstName varchar(255),
    ->     Age int,
    ->     PRIMARY KEY (ID)
    -> );
Query OK, 0 rows affected (0.01 sec)

mysql> INSERT INTO Persons (ID, LastName, FirstName, Age)
-> VALUE ('102', 'JAGADEESH', 'Thirumalesh', '9');
```

```

| 102 | JAGADEESH | Thirumalesh |    9 |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO Persons (ID, LastName, FirstName, Age) VALUE ('102', 'APPARAO', 'Aruna', '8');
ERROR 1062 (23000): Duplicate entry '102' for key 'Persons.PRIMARY'
mysql> select * from Persons;
+----+-----+-----+-----+
| ID | LastName | FirstName | Age  |
+----+-----+-----+-----+
| 102 | JAGADEESH | Thirumalesh |    9 |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> INSERT INTO Persons (ID, LastName, FirstName, Age) VALUE ('102', 'APPARAO', 'Aruna', '8');
ERROR 1062 (23000): Duplicate entry '102' for key 'Persons.PRIMARY'
mysql> INSERT INTO Persons (ID, LastName, FirstName, Age) VALUE ('111', 'APPARAO', 'Aruna', '8');
Query OK, 1 row affected (0.00 sec)

mysql> select * from Persons;
+----+-----+-----+-----+
| ID | LastName | FirstName | Age  |
+----+-----+-----+-----+
| 102 | JAGADEESH | Thirumalesh |    9 |
| 111 | APPARAO   | Aruna     |    8 |
+----+-----+-----+-----+
2 rows in set (0.00 sec)

```

```

root@ip-120-0-22-8: ~
-> 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 tables' at line 2
mysql> select * from Persons
-> 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 tables' at line 2
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| jagadeesh     |
| mysql         |
| performance_schema |
| sys           |
+-----+
5 rows in set (0.00 sec)

mysql> select * from Persons;
+----+-----+-----+-----+
| ID | LastName | FirstName | Age  |
+----+-----+-----+-----+
| 102 | JAGADEESH | Thirumalesh |    9 |
+----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

```