

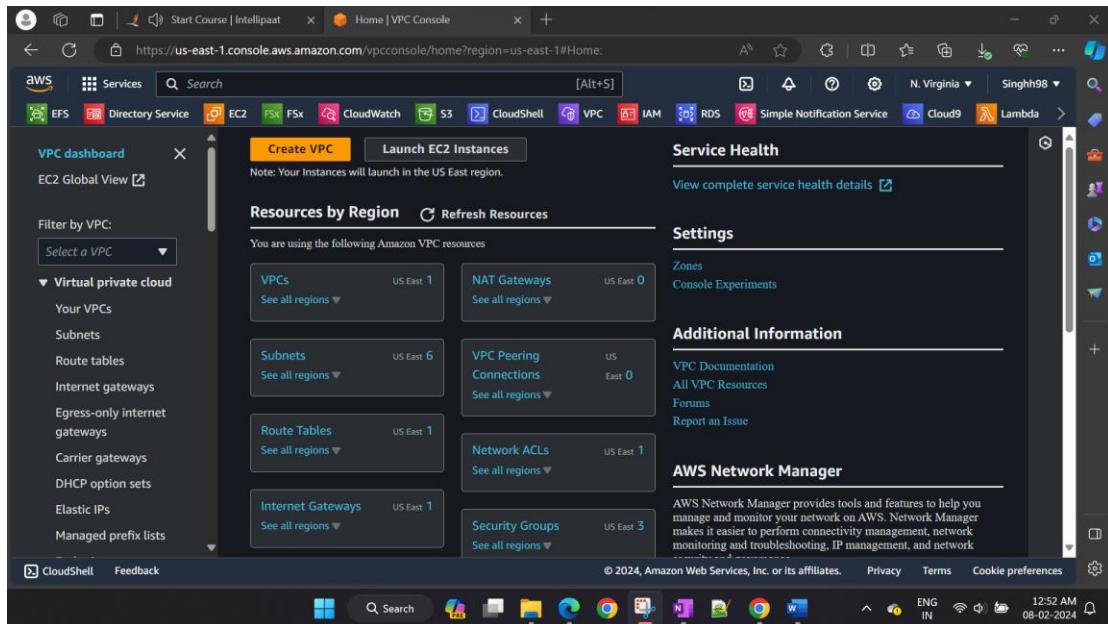
## **INDEX**

## **Page No**

Task – 1: Creation of VPC & verifying the component. ....	02
Task – 2: Creation of EC2 instance. ....	06
Task – 3: Creation of Subnet-Group in RDS. ....	09
Task – 4: Creation RDS. ....	12
Task – 5: Creation of Endpoint. ....	20
Task – 6: Creation of Target Group. ....	22
Task – 7: Creation of Load Balancer. ....	25
Task – 8: Creation of NAT Gateway. ....	27
Task – 9: Attaching NAT Gateway to the Private Subnet. ....	29
Task – 10: Addition of EC2 instance in the Target Group. ....	31
Task – 11: Installing Apache-2 web server in EC2 instance & running it .....	32
Task – 12: Downloading the PHP code in EC2 instance & running it. ....	36
Task – 13: Connecting to RDS database. ....	38
Task – 14: Changing Security Group rule of the MySQL RDS database. ....	40
Task – 15: Installing MySQL client for RDS database. ....	41
Task – 16: Connecting to MySQL RDS database. ....	42
Task – 17: Creation of intel mysql database. ....	42
Task – 18: Creation of table data in intel mysql database and entering the records. ....	44
Task – 19: Creation of Image of EC2 Instance. ....	45
Task – 20: Creation of Launch Template. ....	46
Task – 21: Creation of Auto-Scaling Group. ....	49
THE END .....	56

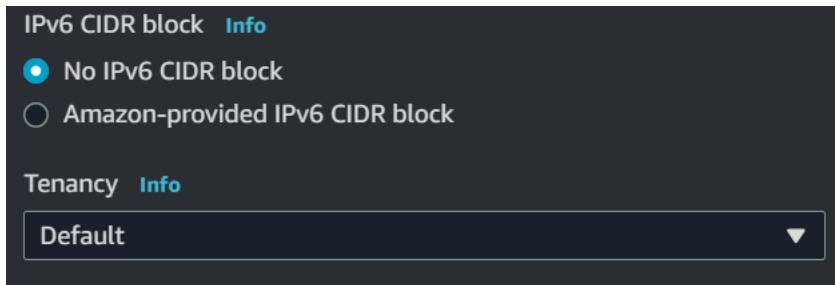
## Task – 1: Creation of VPC & verifying the component.

1. Login in to AWS management console by using our credentials.
2. Search the service **VPC**.
3. As we could see below after clicking on **VPC** service, there is VPC dashboard showing. And we will click on Create VPC.



4. Here we will select the VPC and more. We have given name to the VPC is **VPC-Project**. And the CIDR is **10.0.0.0/24**.

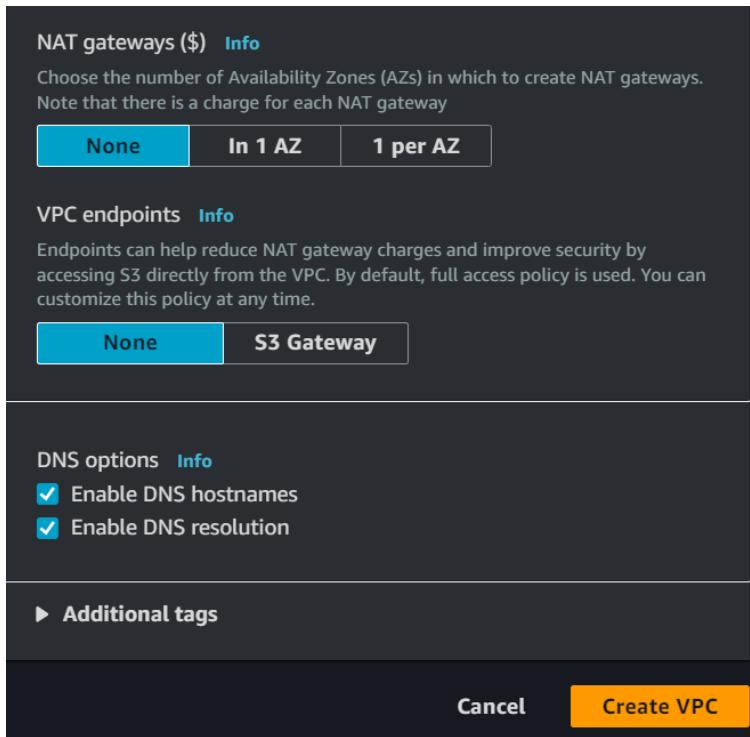
A screenshot of the 'Create VPC' wizard. The top navigation bar shows 'VPC > Your VPCs > Create VPC'. The main title is 'Create VPC' with an 'Info' link. A descriptive text states: 'A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as'. The 'VPC settings' section contains a 'Resources to create' field with two options: 'VPC only' (unchecked) and 'VPC and more' (checked). The 'Name tag auto-generation' section has a checked 'Auto-generate' checkbox and a text input field containing 'VPC-Project'. The 'IPv4 CIDR block' section shows '10.0.0.0/24' in a text input field with '256 IPs' next to it. A note at the bottom says 'CIDR block size must be between /16 and /28.'



5. We have taken the availability zones is 2, public subnet is 2, private subnet is 4.

The screenshot shows the 'Number of Availability Zones (AZs)' section with a dropdown menu set to 2. It also shows the 'Number of public subnets' section with a dropdown menu set to 2, and the 'Number of private subnets' section with a dropdown menu set to 4. Both sections include descriptive text and a 'Customize' link.

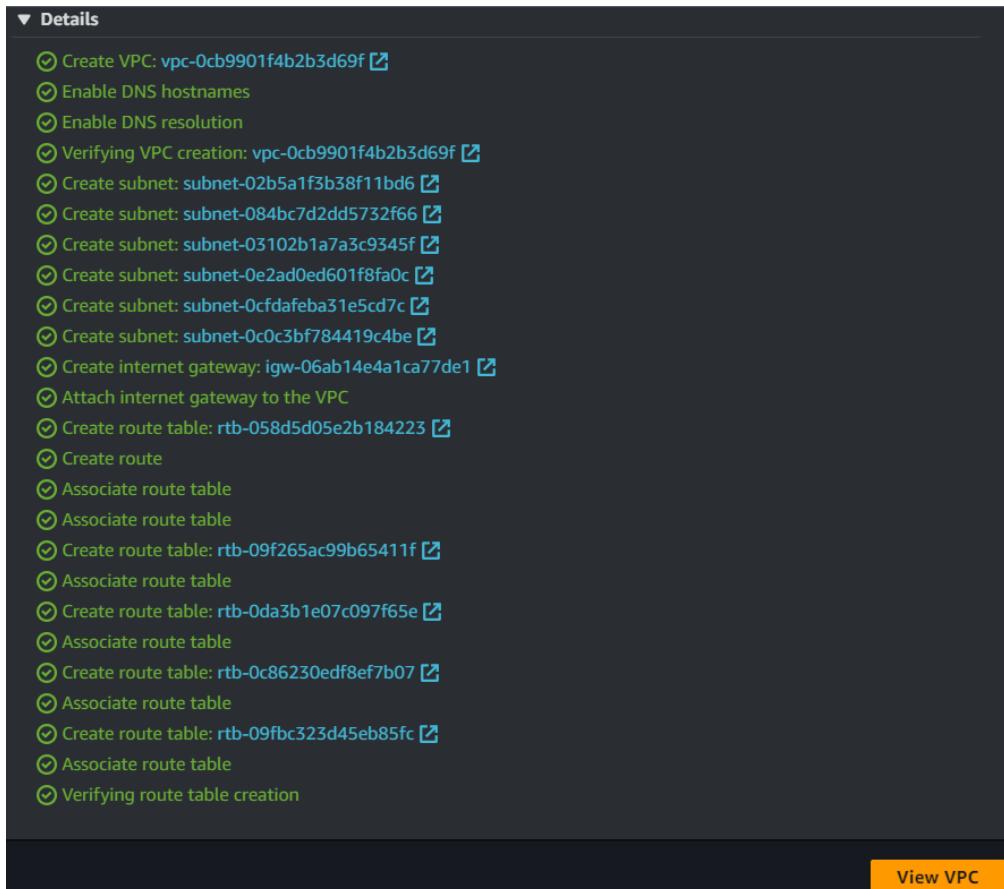
6. NAT Gateway & VPC endpoint will not be attaching as of now. And then we will click on Create VPC.



7. We could see the whole diagram of VPC which is created.



8. Now we could see the whole VPC components which is getting created.



## 9. Now we could see in VPC dashboard that the VPC has been created.

Your VPCs (2) <a href="#">Info</a>					
	Name	VPC ID	State	IPv4 CIDR	IPv6 CIDR
<input type="checkbox"/>	Default VPC	vpc-0cc0d19fd7c30f690	<span>Available</span>	172.31.0.0/16	-
<input type="checkbox"/>	VPC-Project-vpc	vpc-0cb9901f4b2b3d69f	<span>Available</span>	10.0.0.0/24	-

## 10.Two public & Four Private subnets has been created.

Subnets (12) <a href="#">Info</a>					
	Name	Subnet ID	State	VPC	IPv4 CIDR
<input type="checkbox"/>	VPC-Project-subnet-public2-us-east-1b	subnet-084bc7d2dd5732f66	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.1
<input type="checkbox"/>	VPC-Project-subnet-public1-us-east-1a	subnet-02b5a1f3b38f11bd6	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.0
<input type="checkbox"/>	VPC-Project-subnet-private4-us-east-1b	subnet-0c0c3bf784419c4be	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.1
<input type="checkbox"/>	VPC-Project-subnet-private3-us-east-1a	subnet-0cfdafeba31e5cd7c	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.1
<input type="checkbox"/>	VPC-Project-subnet-private2-us-east-1b	subnet-0e2ad0ed601f8fa0c	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.1
<input type="checkbox"/>	VPC-Project-subnet-private1-us-east-1a	subnet-03102b1a7a3c9345f	<span>Available</span>	vpc-0cb9901f4b2b3d69f   VPC...	10.0.0.1

## 11.Route tables has been created.

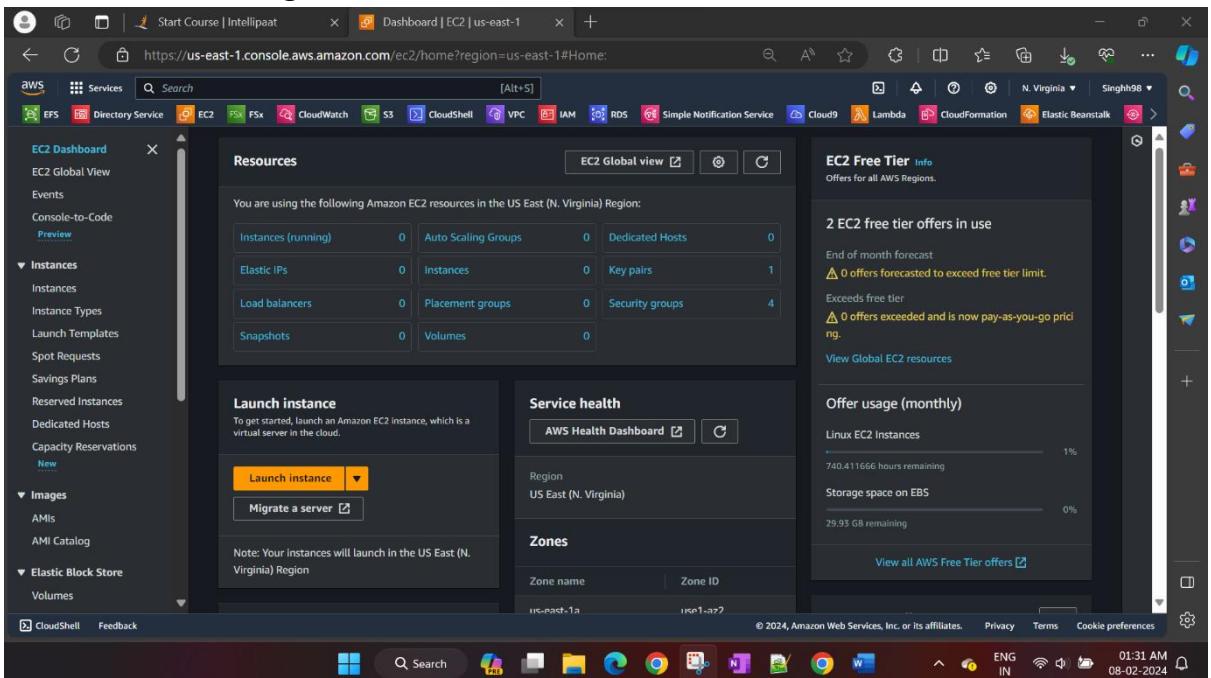
Route tables (6) <a href="#">Info</a>					
	Name	Route table ID	Explicit subnet associati...	Edge associations	Main
<input type="checkbox"/>	VPC-Project-rtb-public	rtb-058d5d05e2b184223	2 subnets	-	No
<input type="checkbox"/>	VPC-Project-rtb-private4-us-east-1b	rtb-09fbc323d45eb85fc	subnet-0c0c3bf784419c4...	-	No
<input type="checkbox"/>	VPC-Project-rtb-private3-us-east-1a	rtb-0c86230edf8ef7b07	subnet-0cfdafeba31e5cd...	-	No
<input type="checkbox"/>	VPC-Project-rtb-private2-us-east-1b	rtb-0da3b1e07c097f65e	subnet-0e2ad0ed601f8fa...	-	No
<input type="checkbox"/>	VPC-Project-rtb-private1-us-east-1a	rtb-09f265ac99b65411f	subnet-03102b1a7a3c93...	-	No
<input type="checkbox"/>	-	rtb-0296784925f256572	-	-	Yes

12. Internet-Gateway is attached with the public route table.

rtb-058d5d05e2b184223 / VPC-Project-rtb-public					
Details	Routes	Subnet associations	Edge associations	Route propagation	Tags
<strong>Routes (2)</strong>					
<input type="button"/> Filter routes				Both	Edit routes
Destination	Target	Status	Propagated		
0.0.0.0/0	igw-06ab14e4a1ca77de1	Active	No		
10.0.0.0/24	local	Active	No		

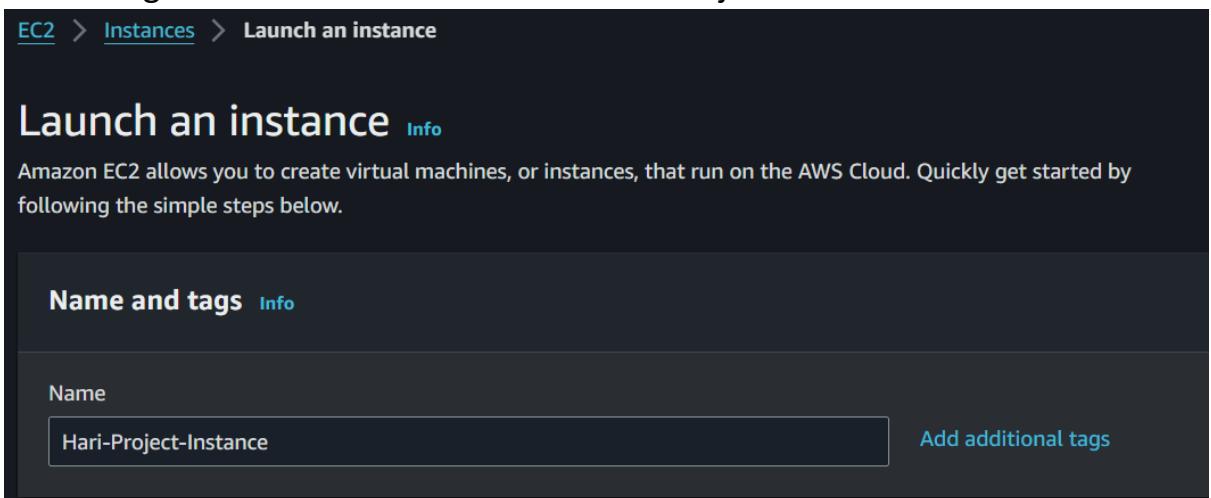
## Task – 2: Creation of EC2 instance.

1. Search the service **EC2** in AWS console.
2. As we could see below after clicking on **EC2** service, there is EC2 dashboard showing. And we will click on Launch Instance.



The screenshot shows the AWS EC2 Dashboard. On the left, there's a sidebar with various EC2-related options like Instances, Images, and Elastic Block Store. The main area displays resource counts (Instances running: 0, Auto Scaling Groups: 0, Dedicated Hosts: 0, etc.) and a prominent 'Launch instance' button. To the right, there's a 'Service health' section and a 'Free Tier' summary indicating 2 offers in use.

3. We have given name to the instance is Hari-Project-Instance.



The screenshot shows the 'Launch an instance' step in the EC2 wizard. It starts with a brief introduction about creating instances. Below that, the 'Name and tags' section is shown, where the instance name 'Hari-Project-Instance' is entered. There's also a link to 'Add additional tags'.

#### 4. We have taken **Ubuntu** as AMI.

The screenshot shows the AWS Management Console interface for selecting an Amazon Machine Image (AMI). At the top, there's a search bar with placeholder text "Search our full catalog including 1000s of application and OS images". Below it, a "Quick Start" section lists several operating systems: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux, each with its respective logo. To the right of this section is a "Browse more AMIs" button with the subtext "Including AMIs from AWS, Marketplace and the Community".

The main content area displays a specific AMI entry:

- Amazon Machine Image (AMI)**
- Ubuntu Server 22.04 LTS (HVM), SSD Volume Type**
- AMI ID: ami-0c7217cdde317cfec (64-bit (x86)) / ami-05d47d29a4c2d19e1 (64-bit (Arm))
- Virtualization: hvm ENA enabled: true Root device type: ebs
- Free tier eligible

Below this, detailed information is provided:

- Description**: Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2023-12-07
- Architecture**: 64-bit (x86)
- AMI ID**: ami-0c7217cdde317cfec
- Verified provider**

Under the "Instance type" section, the **t2.micro** instance type is selected, described as "Free tier eligible". It includes details about family, base pricing for various operating systems, and a note that additional costs apply for AMIs with pre-installed software. There are buttons for "All generations" and "Compare instance types".

In the "Key pair (login)" section, it is noted that a key pair is required for secure connection. A dropdown menu shows "Instance-key-pair" and a "Create new key pair" button.

5. In Network settings, we have taken the **VPC** which is created above. And we creating this instance in **private** subnet. we will create new security group as **Hari-Project-Instance-SG**.

▼ Network settings [Info](#)

VPC - required | [Info](#)

vpc-0cb9901f4b2b3d69f (VPC-Project-vpc)  
10.0.0.0/24

Subnet | [Info](#)

subnet-03102b1a7a3c9345f VPC-Project-subnet-private1-us-east-1a  
VPC: vpc-0cb9901f4b2b3d69f Owner: 101304436132 Availability Zone: us-east-1a  
IP addresses available: 11 CIDR: 10.0.0.128/28

[Create new subnet](#)

Auto-assign public IP | [Info](#)

Disable

Firewall (security groups) | [Info](#)

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group     Select existing security group

Security group name - required

Hari-Project-Instance-SG

This security group will be added to all network interfaces. The name can't be edited after the security group is created. Max length is 255 characters. Valid characters: a-z, A-Z, 0-9, spaces, and .-:/()\#=;&;{}!\$\*

Description - required | [Info](#)

Hari-Project-Instance-SG

## 6. Inbound rule we have taken SSH, HTTP from Anywhere.

Inbound Security Group Rules

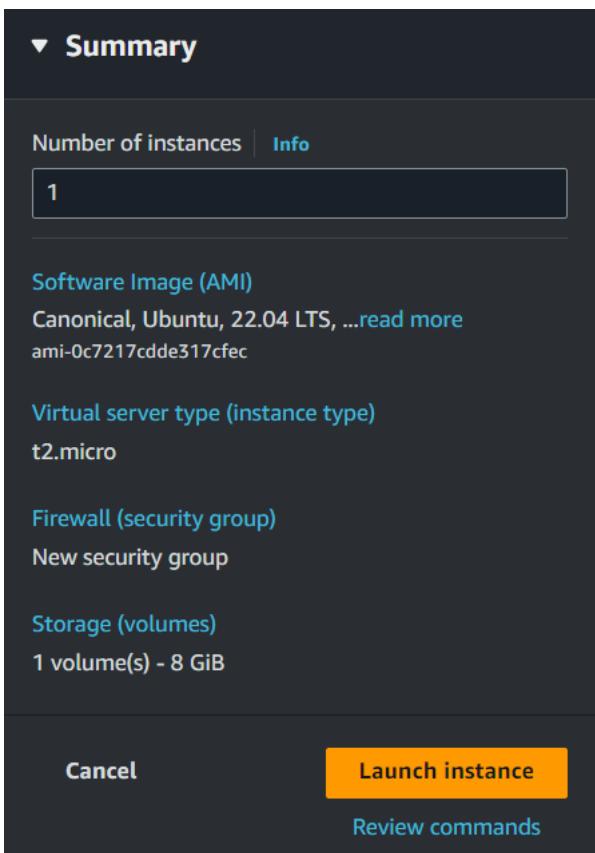
▼ Security group rule 1 (TCP, 22, 0.0.0.0/0) [Remove](#)

Type   <a href="#">Info</a>	Protocol   <a href="#">Info</a>	Port range   <a href="#">Info</a>
ssh	TCP	22
Source type   <a href="#">Info</a>	Source   <a href="#">Info</a>	Description - optional   <a href="#">Info</a>
Anywhere	Add CIDR, prefix list or security 0.0.0.0/0 X	e.g. SSH for admin desktop

▼ Security group rule 2 (TCP, 80, 0.0.0.0/0) [Remove](#)

Type   <a href="#">Info</a>	Protocol   <a href="#">Info</a>	Port range   <a href="#">Info</a>
HTTP	TCP	80
Source type   <a href="#">Info</a>	Source   <a href="#">Info</a>	Description - optional   <a href="#">Info</a>
Anywhere	Add CIDR, prefix list or security 0.0.0.0/0 X	e.g. SSH for admin desktop

7. Now we will click on Launch Instance.



8. As we could see that the instance has been launched.

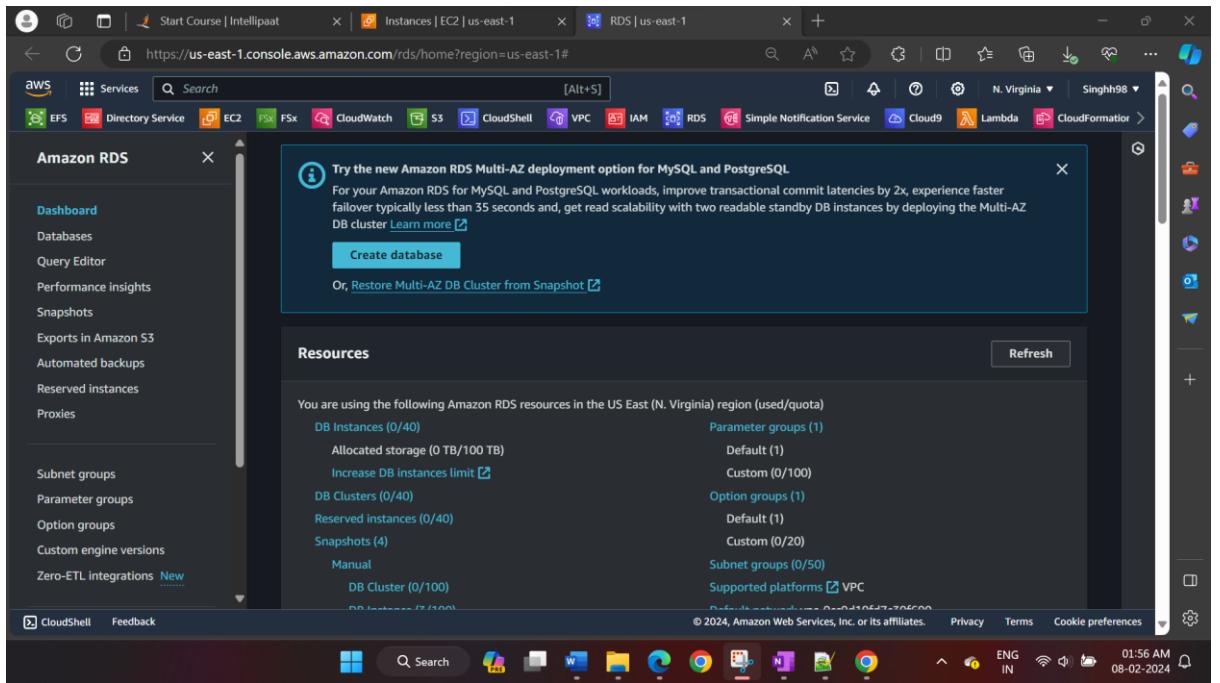
Instances (1) <a href="#">Info</a>		<a href="#">C</a>	Connect	Instance state ▾	Actions ▾	Launch instances ▾	▼
<input type="text"/> <a href="#">Find Instance by attribute or tag (case-sensitive)</a>							
<input type="checkbox"/>	Any state	▼		▼	▼	▼	▼
<input type="checkbox"/>	Name <a href="#">Edit</a>	▼	Instance ID	Instance state	Instance type	Status check	Alarm status

Details for the single instance:

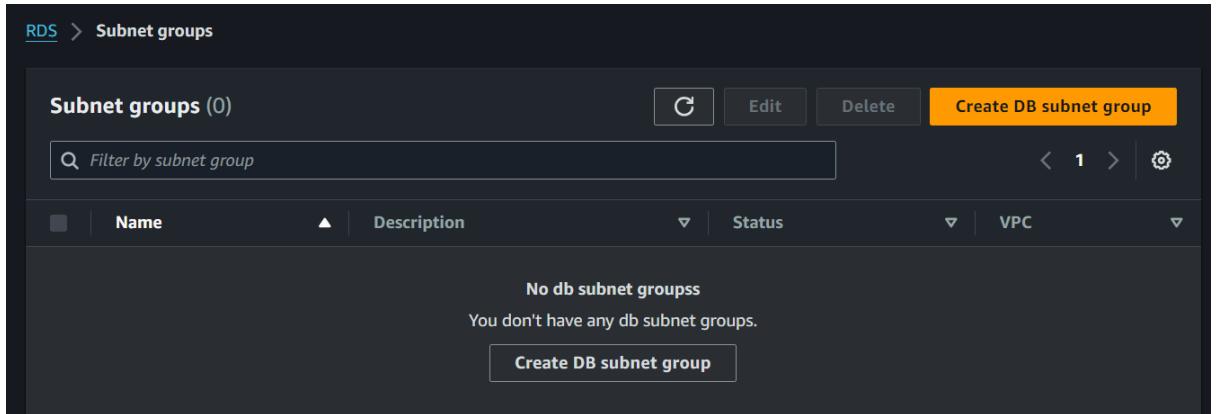
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability
Hari-Project-Instance	i-0f3b97daedfb57f79	<span>Running</span> <a href="#">View</a>	t2.micro	<span>2/2 checks passed</span> <a href="#">View alarms</a> +	us-east-1a	

## Task – 3: Creation of Subnet-Group in RDS.

1. Search the service **RDS** in AWS console.
2. As we could see below after clicking on **RDS** service, there is RDS dashboard showing.



3. Now we will click on the **Subnet groups** in the left panel.
4. Then will click on **Create DB subnet group**.



5. We have given name to the subnet group is **Project-Subnet-Group-RDS**.  
And we are taking the VPC which we created in this project.

## Create DB subnet group

To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.

### Subnet group details

#### Name

You won't be able to modify the name after your subnet group has been created.

Project-Subnet-Group-RDS

Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.

#### Description

Project-Subnet-Group-RDS

#### VPC

Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.

VPC-Project-vpc (vpc-0cb9901f4b2b3d69f)

6. We are taking two AZs. And one-one private subnet from each zone in which the RDS will going to be created.

### Add subnets

#### Availability Zones

Choose the Availability Zones that include the subnets you want to add.

Choose an availability zone

us-east-1a X

us-east-1b X

#### Subnets

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

Select subnets

subnet-0cfdafeba31e5cd7c (10.0.0.160/28) X

subnet-0c0c3bf784419c4be (10.0.0.176/28) X

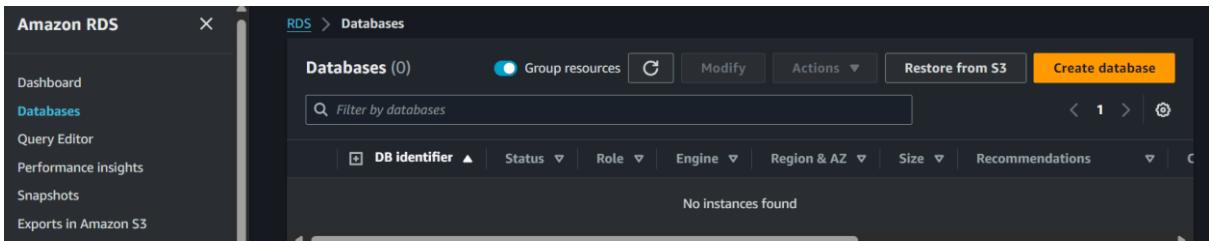
7. Now will click on Create.

Subnets selected (2)		
Availability zone	Subnet ID	CIDR block
us-east-1a	subnet-0cfdfafeba31e5cd7c	10.0.0.160/28
us-east-1b	subnet-0c0c3bf784419c4be	10.0.0.176/28

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

## Task – 4: Creation of RDS database.

- Now we will click on Databases in the left panel in the RDS dashboard. And then click on Create database.



The screenshot shows the RDS dashboard with the 'Databases' section selected. The top navigation bar includes 'RDS > Databases', 'Group resources' (radio button selected), 'Actions', 'Restore from S3', and a prominent 'Create database' button. The main area displays 'Databases (0)' and a search bar labeled 'Filter by databases'. Below is a table header with columns: DB identifier, Status, Role, Engine, Region & AZ, Size, and Recommendations. A message 'No instances found' is displayed at the bottom. The URL in the browser is 'RDS > Create database'.

**Create database**

**Choose a database creation method** Info

**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.

- As per the assignment we have to create MySQL database. So, we have selected MySQL.

## Engine options

Engine type [Info](#)

Aurora (MySQL Compatible)



Aurora (PostgreSQL Compatible)



MySQL



MariaDB



Edition

MySQL Community



### Known issues/limitations

Review the [Known issues/limitations](#) to learn about potential compatibility issues with specific database versions.

Engine version [Info](#)

View the engine versions that support the following database features.

[▼ Hide filters](#)

Show versions that support the Multi-AZ DB cluster [Info](#)

Create a Multi-AZ DB cluster with one primary DB instance and two readable standby DB instances. Multi-AZ DB clusters provide up to 2x faster transaction commit latency and automatic failover in typically under 35 seconds.

Show versions that support the Amazon RDS Optimized Writes [Info](#)

Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Engine Version

MySQL 8.0.35

### 3. We are going for Free tier.

#### Templates

Choose a sample template to meet your use case.

Production

Use defaults for high availability and fast, consistent performance.

Dev/Test

This instance is intended for development use outside of a production environment.

Free tier

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.  
[Info](#)

4. We have given name is **Hari-Project-MySQL-RDS**.

**Settings**

**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.

**Hari-Project-MySQL-RDS**

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.

5. Given credentials as username is **admin** and the password.

**▼ Credentials Settings**

**Master username** [Info](#)  
Type a login ID for the master user of your DB instance.

**admin**

1 to 16 alphanumeric characters. The 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.

**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.

DB instance class [Info](#)

▼ Hide filters

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

Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.

Include previous generation classes

Standard classes (includes m classes)

Memory optimized classes (includes r and x classes)

Burstable classes (includes t classes)

db.t3.micro

2 vCPUs 1 GiB RAM Network: 2,085 Mbps

6. Expanded the storage autoscaling and unchecked it, because we don't want any autoscaling as of now.

## Storage

Storage type [Info](#)

General Purpose SSD (gp2)

Baseline performance determined by volume size

Allocated storage [Info](#)

20

GiB

The minimum value is 20 GiB and the maximum value is 6,144 GiB

**ⓘ After you modify the storage for a DB instance, the status of the DB instance will be in storage-optimization. Your instance will remain available as the storage-optimization operation completes.**

[Learn more](#)

▼ Storage autoscaling

Storage autoscaling [Info](#)

Provides dynamic scaling support for your database's storage based on your application's needs.

Enable storage autoscaling

Enabling this feature will allow the storage to increase after the specified threshold is exceeded.

7. We are taking the VPC and subnet group which is created earlier in this project.

## Connectivity Info



### Compute resource

Choose whether to set up a connection to a compute resource for this database. Setting up a connection will automatically change connectivity settings so that the compute resource can connect to this database.

**Don't connect to an EC2 compute resource**

Don't set up a connection to a compute resource for this database. You can manually set up a connection to a compute resource later.

**Connect to an EC2 compute resource**

Set up a connection to an EC2 compute resource for this database.

### Virtual private cloud (VPC) Info

Choose the VPC. The VPC defines the virtual networking environment for this DB instance.

VPC-Project-vpc (vpc-0cb9901f4b2b3d69f)

6 Subnets, 2 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 Info

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

project-subnet-group-rds

2 Subnets, 2 Availability Zones



8. Public access is NO. And creating the new security group named **Hari-RDS-SG**.

#### Public access [Info](#)

##### Yes

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

##### No

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

#### VPC security group (firewall) [Info](#)

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

#### New VPC security group name

Hari-RDS-SG

#### Availability Zone [Info](#)

No preference



#### RDS Proxy

RDS Proxy is a fully managed, highly available database proxy that improves application scalability, resiliency, and security.

##### Create an RDS Proxy [Info](#)

RDS automatically creates an IAM role and a Secrets Manager secret for the proxy. RDS Proxy has additional costs. For more information, see [Amazon RDS Proxy pricing](#).

#### ▼ Additional configuration

#### Database port [Info](#)

TCP/IP port that the database will use for application connections.

3306

## Database authentication

#### Database authentication options [Info](#)

##### Password authentication

Authenticates using database passwords.

##### Password and IAM database authentication

Authenticates using the database password and user credentials through AWS IAM users and roles.

##### Password and Kerberos authentication

Choose a directory in which you want to allow authorized users to authenticate with this DB instance using Kerberos Authentication.

## Monitoring

### Enable Enhanced Monitoring

Enabling Enhanced Monitoring metrics are useful when you want to see how different processes or threads use the CPU.

9. We will expand the Additional Configuration & will disable the automated backups.

### ▼ Additional configuration

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

#### Database options

Initial database name [Info](#)

If you do not specify a database name, Amazon RDS does not create a database.

DB parameter group [Info](#)

▼

Option group [Info](#)

▼

#### Backup

##### Enable automated backups

Creates a point-in-time snapshot of your database

**Encryption**

**Enable encryption**  
Choose to encrypt the given instance. Master key IDs and aliases appear in the list after they have been created using the AWS Key Management Service console. [Info](#)

**AWS KMS key** [Info](#)

(default) aws/rds ▾

**Account**  
101304436132

**KMS key ID**  
alias/aws/rds

**Log exports**  
Select the log types to publish to Amazon CloudWatch Logs

Audit log  
 Error log  
 General log  
 Slow query log

**IAM role**  
The following service-linked role is used for publishing logs to CloudWatch Logs.

RDS service-linked role

10. Deletion protection will be disabled. Otherwise, we need to disable it first before deleting this.

**Maintenance**

Auto minor version upgrade [Info](#)

**Enable auto minor version upgrade**  
Enabling auto minor version upgrade will automatically upgrade to new minor versions as they are released. The automatic upgrades occur during the maintenance window for the database.

**Maintenance window** [Info](#)  
Select the period you want pending modifications or maintenance applied to the database by Amazon RDS.

Choose a window  
 No preference

**Deletion protection**

**Enable deletion protection**  
Protects the database from being deleted accidentally. While this option is enabled, you can't delete the database.

11. Now will click on Create database.

**Estimated monthly costs**

The Amazon RDS Free Tier is available to you for 12 months. Each calendar month, the free tier will allow you to use the Amazon RDS resources listed below for free:

- 750 hrs of Amazon RDS in a Single-AZ db.t2.micro, db.t3.micro or db.t4g.micro Instance.
- 20 GB of General Purpose Storage (SSD).
- 20 GB for automated backup storage and any user-initiated DB Snapshots.

[Learn more about AWS Free Tier.](#)

When your free usage expires or if your application use exceeds the free usage tiers, you simply pay standard, pay-as-you-go service rates as described in the [Amazon RDS Pricing page](#).

**Cancel** **Create database**

12. It will take some time to create the RDS database.

13. Now we could see that the database has been created.

The screenshot shows the 'Databases' section of the Amazon RDS console. A new database entry is listed:

DB identifier	Status	Role	Engine	Region & AZ	Size	Recom
hari-project-mysql-rds	Available	Instance	MySQL Community	us-east-1b	db.t3.micro	

## Task – 5: Creation of Endpoint.

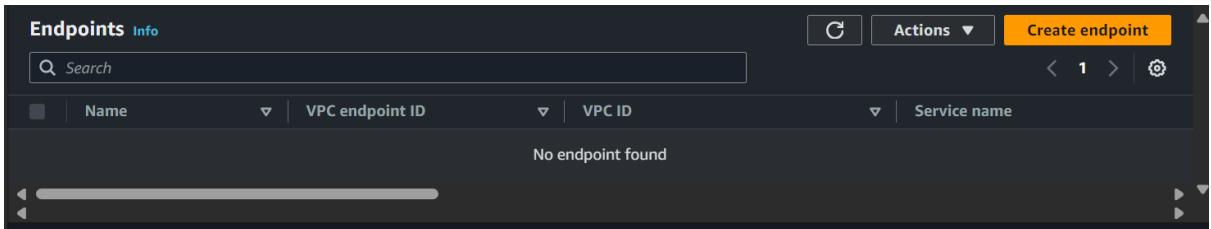
1. Search the service **VPC** in AWS console.
2. As we could see below after clicking on **VPC** service, there is VPC dashboard showing.

The screenshot shows the VPC dashboard in the AWS Management Console. Key sections include:

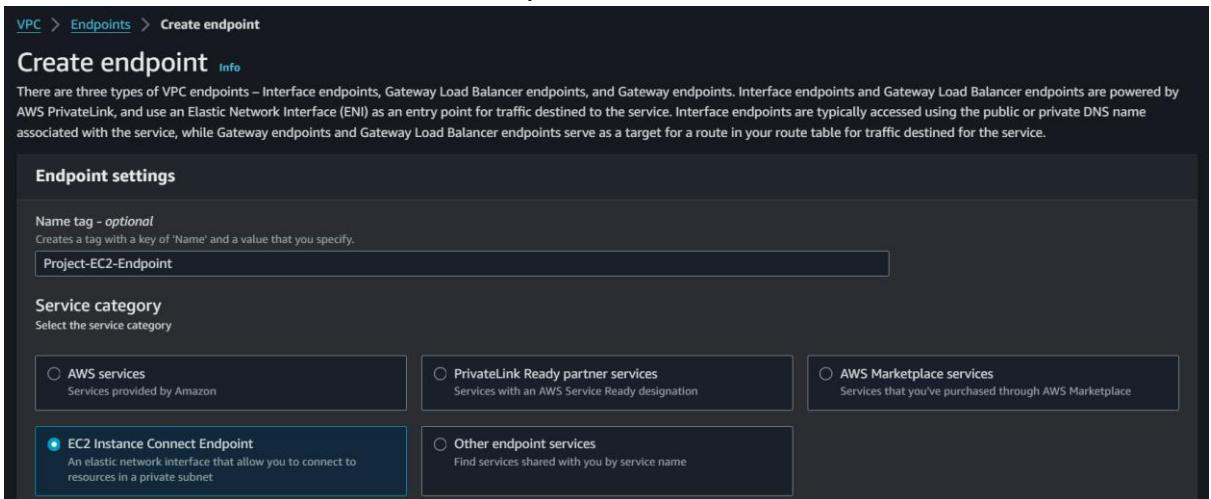
- Create VPC** and **Launch EC2 Instances** buttons.
- Service Health** status.
- Settings** for Zones and Console Experiments.
- Additional Information** links to VPC Documentation, All VPC Resources, Forums, and Report an Issue.
- AWS Network Manager** information and a 'Get started with Network Manager' link.

The left sidebar lists VPC-related services: EFS, Directory Service, EC2, FSx, CloudWatch, S3, CloudShell, VPC, IAM, RDS, Simple Notification Service, Cloud9, Lambda, CloudFormation.

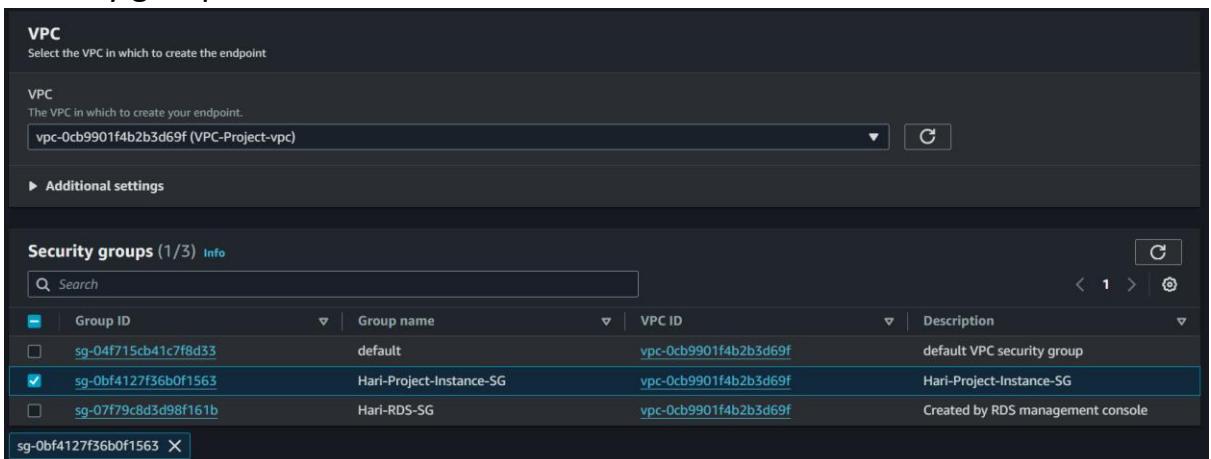
- From the left panel will click on Endpoints → Create endpoint.



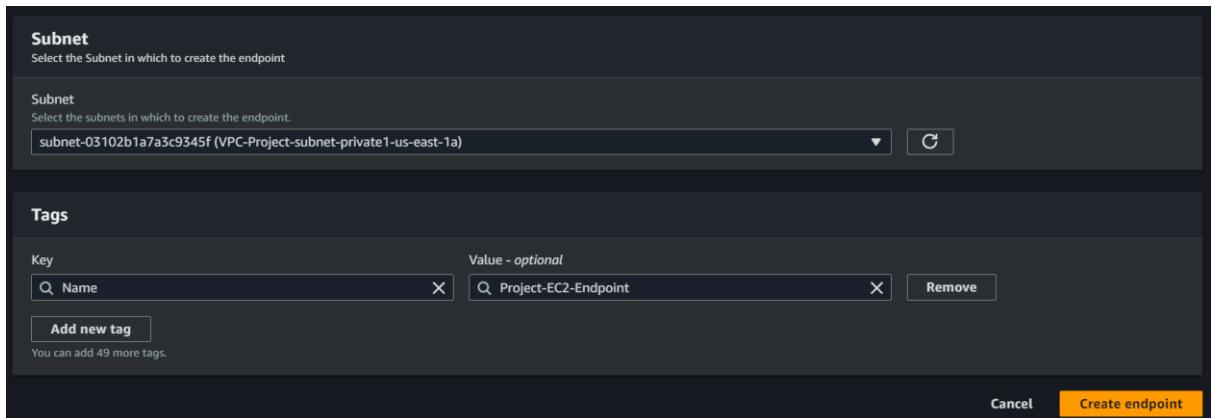
- We have given name to the endpoint is **Project-EC2-Endpoint**. And have selected EC2 instance connect endpoint.



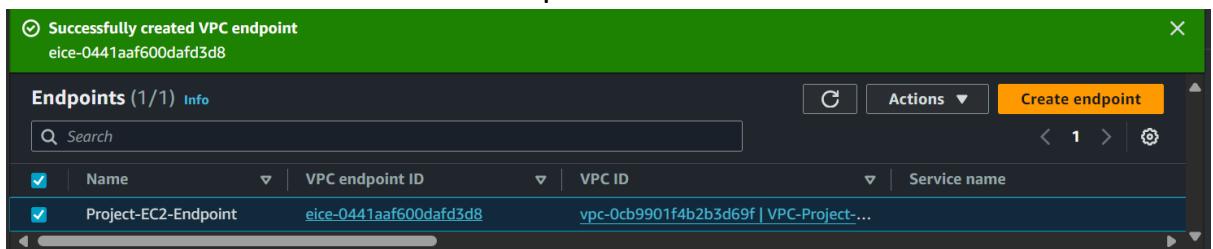
- Selected the VPC which is created above. And selected the instance security group.



- Selected the same subnet in which the instance has been created.
- Then will click on Create endpoint.

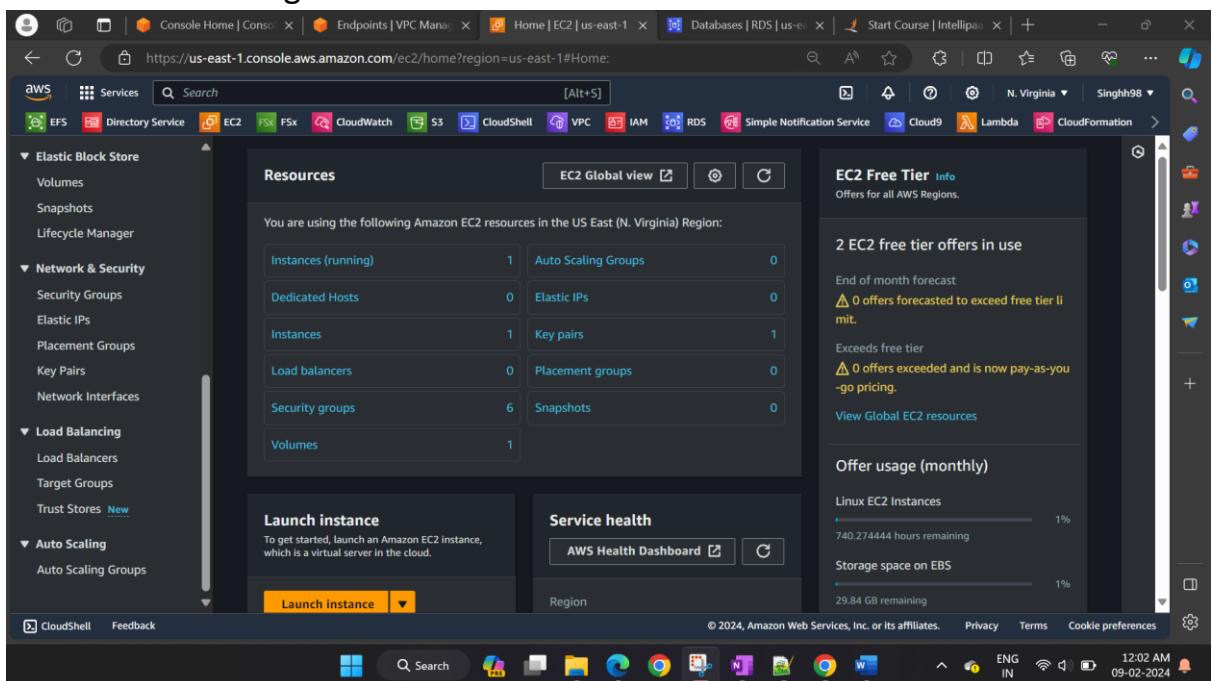


- Now we could see that the EC2 endpoint has been created.

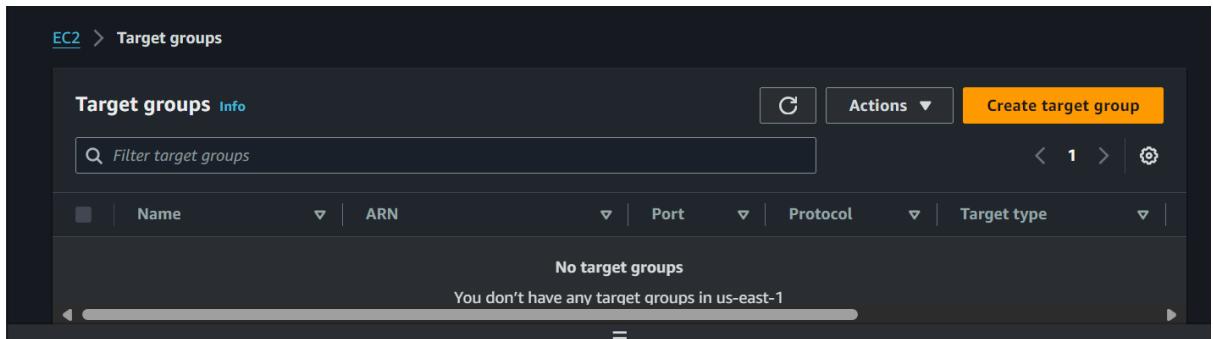


## Task – 6: Creation of Target Group.

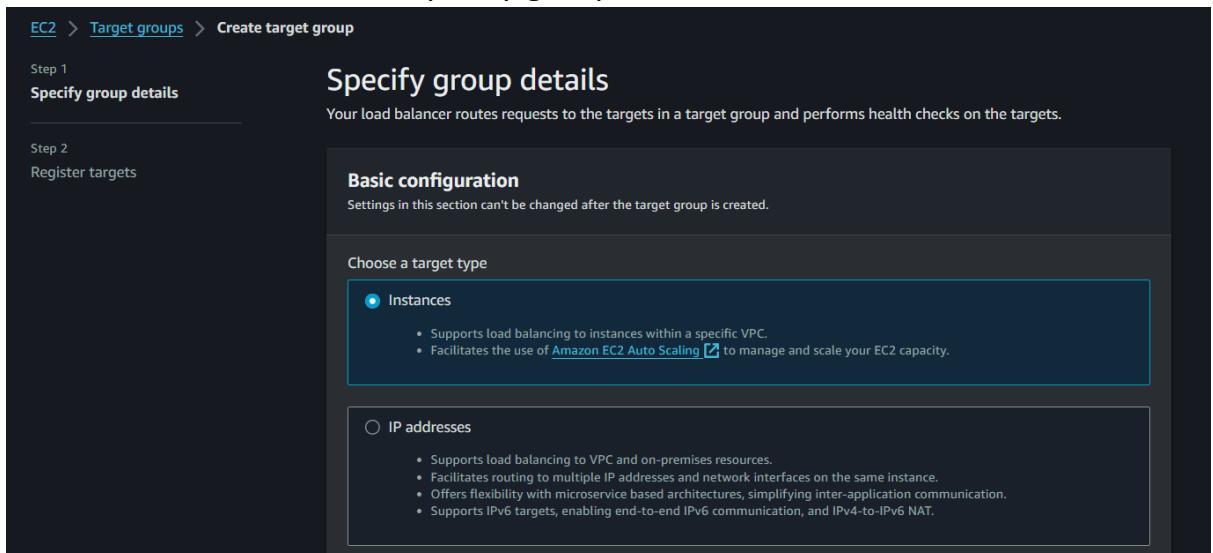
- Search the service **EC2** in AWS console.
- As we could see below after clicking on **EC2** service, there is EC2 dashboard showing.



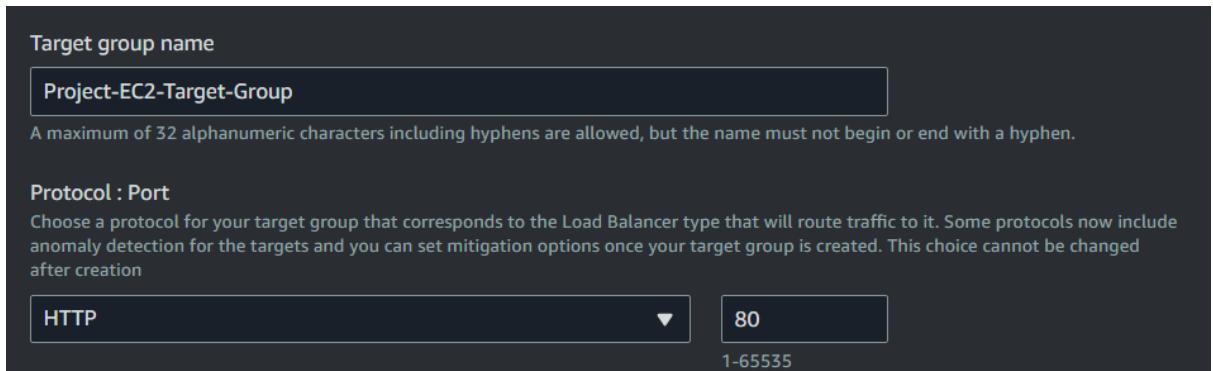
- From the left panel, will click on Target Groups → Create target group.



#### 4. We will select Instances in Specify group details.



#### 5. We have given name to the target group is Project-EC2-Target-Group.



#### 6. We will be selecting the our VPC which is created in this project.

**VPC**  
Select the VPC with the instances that you want to include in the target group. Only VPCs that support the IP address type selected above are available in this list.

**VPC-Project-vpc**  
vpc-0cb9901f4b2b3d69f  
IPv4: 10.0.0.0/24

**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.

7. Now will scroll down and will click on Next.

**► Tags - optional**  
Consider adding tags to your target group. Tags enable you to categorize your AWS resources so you can more easily manage them.

Cancel      **Next**

8. As of now we are not registering any targets. Simply we will click on Create target group.

**Review targets**

**Targets (0)** Remove all pending

Filter targets Show only pending

Instance ID ▾ | Name ▾ | Port ▾ | State ▾ | Security groups ▾ | Zone ▾ | Private IPv4 address | Subnet ID ▾ | Launch time ▾

No instances added yet  
Specify instances above, or leave the group empty if you prefer to add targets later.

0 pending Cancel Previous **Create target group**

9. We could see that the target group has been created successfully.

Successfully created the target group: Project-EC2-Target-Group. Anomaly detection is automatically applied to all registered targets. Results can be viewed in the Targets tab.

[EC2](#) > [Target groups](#) > Project-EC2-Target-Group

## Project-EC2-Target-Group

[Actions ▾](#)

Details					
<a href="#">arn:aws:elasticloadbalancing:us-east-1:101304436132:targetgroup/Project-EC2-Target-Group/7647188ce2c6c634</a>					
Target type Instance	Protocol : Port HTTP: 80	Protocol version HTTP1	VPC <a href="#">vpc-0cb9901f4b2b3d69f</a>		
IP address type IPv4	Load balancer <a href="#">None associated</a>				
0 Total targets	<a href="#">0 Healthy</a>	<a href="#">0 Unhealthy</a>	<a href="#">0 Unused</a>	<a href="#">0 Initial</a>	<a href="#">0 Draining</a>
0 Anomalous					

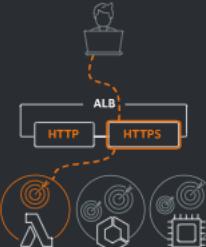
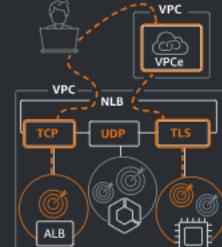
## Task – 7: Creation of Load Balancer.

1. In the EC2 dashboard, from the left panel in bottom will click on Load Balancers → Create load balancers.
2. Then will click on Create in Application load balancer.

[EC2](#) > [Load balancers](#) > Compare and select load balancer type

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

### Compare and select load balancer type

Load balancer types		
<b>Application Load Balancer</b> <a href="#">Info</a>	<b>Network Load Balancer</b> <a href="#">Info</a>	<b>Gateway Load Balancer</b> <a href="#">Info</a>
		
Choose an Application Load Balancer when you need a flexible feature set for your applications with HTTP and HTTPS traffic. Operating at the request level, Application Load Balancers provide advanced routing and visibility features targeted at application architectures, including microservices and containers. <a href="#">Create</a>	Choose a Network Load Balancer when you need ultra-high performance, TLS offloading at scale, centralized certificate deployment, support for UDP, and static IP addresses for your applications. Operating at the connection level, Network Load Balancers are capable of handling millions of requests per second securely while maintaining ultra-low latencies. <a href="#">Create</a>	Choose a Gateway Load Balancer when you need to deploy and manage a fleet of third-party virtual appliances that support GENEVE. These appliances enable you to improve security, compliance, and policy controls. <a href="#">Create</a>
<a href="#">▶ Classic Load Balancer - previous generation</a>		

- We have given name to the load balancer is **Hari-Project-Load-Balancer**. And selecting internet-facing.

**Create Application Load Balancer** Info

The Application Load Balancer distributes incoming HTTP and HTTPS traffic across multiple targets such as Amazon EC2 instances, microservices, and containers, based on request attributes. When the load balancer receives a connection request, it evaluates the listener rules in priority order to determine which rule to apply, and if applicable, it selects a target from the target group for the rule action.

▶ How Application Load Balancers work

**Basic configuration**

Load balancer name  
Name must be unique within your AWS account and can't be changed after the load balancer is created.  
**Hari-Project-Load-Balancer**

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

Scheme | **Info**  
Scheme can't be changed after the load balancer is created.

- 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.

- VPC** is taken which is created in this project. And we are taking the public subnets from each availability zones.

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

**VPC-Project-vpc**  
vpc-0cb9901f4b2b3d69f  
IPv4: 10.0.0.0/24

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

**us-east-1a (use1-az2)**  
Subnet  
**subnet-03102b1a7a3c9345f** VPC-Project-subnet-private1-us-east-1a ▾

**⚠** The selected subnet does not have a route to an internet gateway. This means that your load balancer will not receive internet traffic.  
You can proceed with this selection; however, for internet traffic to reach your load balancer, you must update the subnet's route table in the [VPC console](#).

**IPv4 address**  
Assigned by AWS

**us-east-1b (use1-az4)**  
Subnet  
**subnet-0e2ad0ed601f8fa0c** VPC-Project-subnet-private2-us-east-1b ▾

- Security group we can take default. And the target group we will take which is created earlier in this project.

**Security groups** [Info](#)  
A security group is a set of firewall rules that control the traffic to your load balancer. Select an existing security group, or you can [create a new security group](#).

**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.

- Now everything will be set as default, will scroll down and click on Create load balancer.

**Creation workflow and status**

**Server-side tasks and status**  
After completing and submitting the above steps, all server-side tasks and their statuses become available for monitoring.

[Cancel](#) [Create load balancer](#)

- We could see that the load balancer has successfully created.

**HariProject-Load-Balancer**

**Details**

Load balancer type <b>Application</b>	Status <b>Provisioning</b>	VPC <a href="#">vpc-0cb9901f4b2b3d69f</a>	IP address type <b>IPv4</b>
Scheme <b>Internet-facing</b>	Hosted zone <a href="#">Z355XDDOTRQ7X7K</a>	Availability Zones <a href="#">subnet-02b5a1f3b38f11bd6</a> us-east-1a (use1-a22) <a href="#">subnet-084bc7d2dd5732f66</a> us-east-1b (use1-a24)	Date created February 9, 2024, 00:32 (UTC+05:30)
Load balancer ARN <a href="#">arn:aws:elasticloadbalancing:us-east-1:101304436132:loadbalancer/app/HariProject-Load-Balancer/65bdfb093d6504da</a>	DNS name <a href="#">Info</a> <a href="#">HariProject-Load-Balancer-697324288.us-east-1.elb.amazonaws.com</a> (A Record)		

## Task – 8: Creation of NAT Gateway.

1. Go to VPC dashboard → from the left panel click on NAT gateways → Create NAT gateway.

The screenshot shows the AWS VPC NAT gateways list interface. At the top, there's a search bar with placeholder text "Find resources by attribute or tag". Below the search bar are several filter columns: Name, NAT gateway ID, Connectivity..., State, State message, Primary public IP..., and Primary private IP... Each column has a dropdown arrow icon. To the right of these filters is a "Create NAT gateway" button. Below the filter row, a message says "No NAT gateways found".

2. We have given name to the NAT gateway is **Project-NAT-Gateway**. Subnet will be taking as any public. And will allocate the Elastic IP.

The screenshot shows the "Create NAT gateway" settings page. At the top, a green banner displays the message "Elastic IP address 52.71.93.54 (eipalloc-0159776c7eaa59c60) allocated.". Below the banner, the breadcrumb navigation shows "VPC > NAT gateways > Create NAT gateway". The main title is "Create NAT gateway" with an "Info" link. A descriptive text states: "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." The "NAT gateway settings" section contains the following fields:

- Name - optional:** A text input field containing "Project-NAT-Gateway".

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

The name can be up to 256 characters long.
- Subnet:** A dropdown menu showing "subnet-084bc7d2dd5732f66 (VPC-Project-subnet-public2-us-east-1b)".

Select a subnet in which to create the NAT gateway.
- Connectivity type:** A radio button group where "Public" is selected.

Select a connectivity type for the NAT gateway.

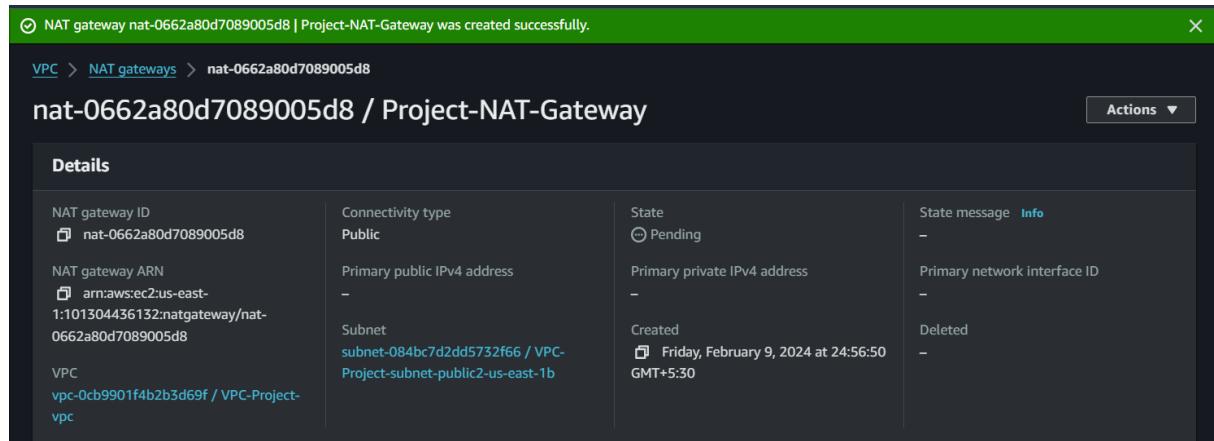
Public  
 Private
- Elastic IP allocation ID - Info:** A dropdown menu showing "eipalloc-0159776c7eaa59c60".

Assign an Elastic IP address to the NAT gateway.

3. Now will scroll down and click on Create NAT gateway.

The screenshot shows the "Tags" configuration page for creating a NAT gateway. The title is "Tags". A sub-instruction says: "A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs." The "Key" field is "Name" and the "Value - optional" field is "Project-NAT-Gateway". There is a "Remove" button next to the value field. Below these fields is a "Add new tag" button. A note at the bottom says "You can add 49 more tags." At the bottom right are "Cancel" and "Create NAT gateway" buttons.

- We could see that NAT gateway has been created successfully.



NAT gateway nat-0662a80d7089005d8 | Project-NAT-Gateway was created successfully.

VPC > NAT gateways > nat-0662a80d7089005d8

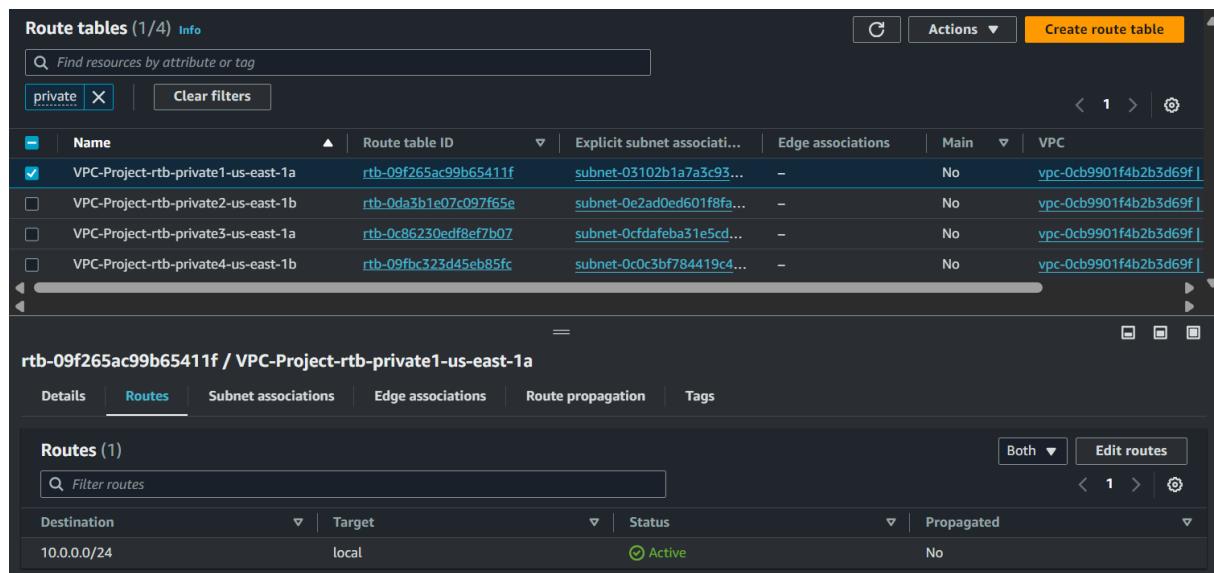
**nat-0662a80d7089005d8 / Project-NAT-Gateway**

**Actions ▾**

Details			
NAT gateway ID nat-0662a80d7089005d8	Connectivity type Public	State Pending	State message -
NAT gateway ARN arn:aws:ec2:us-east-1:101304436132:natgateway/nat-0662a80d7089005d8	Primary public IPv4 address -	Primary private IPv4 address -	Primary network interface ID -
VPC vpc-0cb9901f4b2b3d69f / VPC-Project-vpc	Subnet subnet-084bc7d2dd5732f66 / VPC-Project-subnet-public2-us-east-1b	Created Friday, February 9, 2024 at 24:56:50 GMT+5:30	Deleted -

## Task – 9: Attaching NAT Gateway to the Private Subnet.

- From the VPC dashboard in the left panel, will click on Route tables.
- Now will select rtb-private1 → will go to the Routes section → Edit routes.



**Route tables (1/4) Info**

Find resources by attribute or tag

private X Clear filters

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
✓ VPC-Project-rtb-private1-us-east-1a	rtb-09f265ac99b65411f	subnet-03102b1a7a3c93...	-	No	vpc-0cb9901f4b2b3d69f
□ VPC-Project-rtb-private2-us-east-1b	rtb-0da3b1e07c097f65e	subnet-0e2ad0ed601f8fa...	-	No	vpc-0cb9901f4b2b3d69f
□ VPC-Project-rtb-private3-us-east-1a	rtb-0c86230edf8ef7b07	subnet-0cfdfab31e5cd...	-	No	vpc-0cb9901f4b2b3d69f
□ VPC-Project-rtb-private4-us-east-1b	rtb-09fbcc323d45eb85fc	subnet-0c0c3bf784419c4...	-	No	vpc-0cb9901f4b2b3d69f

**rtb-09f265ac99b65411f / VPC-Project-rtb-private1-us-east-1a**

Details Routes Subnet associations Edge associations Route propagation Tags

**Routes (1)**

Filter routes

Destination	Target	Status	Propagated
10.0.0.0/24	local	Active	No

- Now will click on Add route. IP will be taking from Anywhere 0.0.0.0/0. And selected the NAT gateway which is need to attach.
- Now will click on Save changes.

VPC > Route tables > rtb-09f265ac99b65411f > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/24	local	Active	No
Q 0.0.0.0/0	NAT Gateway	Active	No

Add route

Cancel Preview Save changes

5. Now we will repeat the same above steps from 2-4 for other route table of private subnets.
6. And now we have attached the NAT gateway for private subnet route tables.
7. We could see in the below snips.

Route tables (1/4) Info

Find resources by attribute or tag

private X Clear filters

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
VPC-Project-rtb-private1-us-east-1a	rtb-09f265ac99b65411f	subnet-03102b1a7a3c93...	-	No	vpc-0cb9901f4b2b3d69f
<b>VPC-Project-rtb-private2-us-east-1b</b>	<b>rtb-0da3b1e07c097f65e</b>	<b>subnet-0e2ad0ed601f8fa...</b>	-	No	<b>vpc-0cb9901f4b2b3d69f</b>
VPC-Project-rtb-private3-us-east-1a	rtb-0c86230edf8ef7b07	subnet-0cfdafeba31e5cd...	-	No	vpc-0cb9901f4b2b3d69f
VPC-Project-rtb-private4-us-east-1b	rtb-09fbcb323d45eb85fc	subnet-0c0c3bf784419c4...	-	No	vpc-0cb9901f4b2b3d69f

rtb-0da3b1e07c097f65e / VPC-Project-rtb-private2-us-east-1b

Details Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0662a80d7089005d8	Active	No
10.0.0.0/24	local	Active	No

8.

Route tables (1/4) Info

Find resources by attribute or tag

private X Clear filters

Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC
VPC-Project-rtb-private1-us-east-1a	rtb-09f265ac99b65411f	subnet-03102b1a7a3c93...	-	No	vpc-0cb9901f4b2b3d69f
VPC-Project-rtb-private2-us-east-1b	rtb-0da3b1e07c097f65e	subnet-0e2ad0ed601f8fa...	-	No	vpc-0cb9901f4b2b3d69f
<b>VPC-Project-rtb-private3-us-east-1a</b>	<b>rtb-0c86230edf8ef7b07</b>	<b>subnet-0cfdafeba31e5cd...</b>	-	No	<b>vpc-0cb9901f4b2b3d69f</b>
VPC-Project-rtb-private4-us-east-1b	rtb-09fbcb323d45eb85fc	subnet-0c0c3bf784419c4...	-	No	vpc-0cb9901f4b2b3d69f

rtb-0c86230edf8ef7b07 / VPC-Project-rtb-private3-us-east-1a

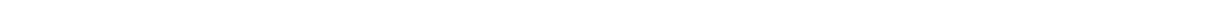
Details Routes Subnet associations Edge associations Route propagation Tags

Routes (2)

Filter routes

Destination	Target	Status	Propagated
0.0.0.0/0	nat-0662a80d7089005d8	Active	No
10.0.0.0/24	local	Active	No

9.



Route tables (1/4) <a href="#">Info</a>						
<input type="text"/> Find resources by attribute or tag						
<a href="#">Actions</a>	<a href="#">Create route table</a>					
Name	Route table ID	Explicit subnet associations	Edge associations	Main	VPC	
VPC-Project-rtb-private1-us-east-1a	rtb-09f265ac99b65411f	subnet-03102b1a7a3c93...	-	No	vpc-0cb9901f4b2b3d69f	
VPC-Project-rtb-private2-us-east-1b	rtb-0da3b1e07c097f65e	subnet-0e2ad0ed601f8fa...	-	No	vpc-0cb9901f4b2b3d69f	
VPC-Project-rtb-private3-us-east-1a	rtb-0c86230edf8ef7b07	subnet-0cfdafeba31e5cd...	-	No	vpc-0cb9901f4b2b3d69f	
<b>VPC-Project-rtb-private4-us-east-1b</b>	<b>rtb-09fbc323d45eb85fc</b>	<b>subnet-0c0c3bf784419c4...</b>	<b>-</b>	<b>No</b>	<b>vpc-0cb9901f4b2b3d69f</b>	

rtb-09fbc323d45eb85fc / VPC-Project-rtb-private4-us-east-1b						
Details	<b>Routes</b>	Subnet associations	Edge associations	Route propagation	Tags	
<b>Routes (2)</b>						
<input type="text"/> Filter routes						
Destination	Target	Status				
0.0.0.0/0	nat-0662a80d7089005d8	Active				
10.0.0.0/24	local	Active				

## Task – 10: Addition of EC2 instance in the Target Group.

- From the EC2 dashboard, click on target group from the left panel.
- We will click on this target group Project-EC2-Target-Group.

Target groups (1) <a href="#">Info</a>						
<input type="text"/> Filter target groups						
<a href="#">Actions</a>	<a href="#">Create target group</a>					
Name	ARN	Port	Protocol	Target type	Load balancer	
Project-EC2-Target-Group	arn:aws:elasticloadbalancing:us-east-1:123456789012:targetgroup/Project-EC2-Target-Group/1234567890123456	80	HTTP	Instance	HariProject-Load-Balancer	

- We will go to the Targets section → Register targets.

Targets	Monitoring	Health checks	Attributes	Tags
<b>Registered targets (0) <a href="#">Info</a></b>				
<small>Anomaly mitigation: Not applicable</small> <input type="button" value="Deregister"/> <input type="button" value="Register targets"/>				
Instance ID	Port	Zone	Health status	Health status details
No registered targets You have not registered targets to this group yet				
<input type="button" value="Register targets"/>				

- Now will select the available instance and will click on Include as pending below.

**Available instances (1/1)**

Instance ID	Name	State	Security groups	Zone	Private IPv4 address
i-0f3b97daedfb57f79	Hari-Project-Instance	Running	Hari-Project-Instance-SG	us-east-1a	10.0.0.141

1 selected

**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**

- Now will click on Register pending targets.

**Review targets**

**Targets (1)**

Instance ID	Name	Port	State	Security groups	Zone	Private IPv4 address	Subnet ID	Launch time
i-0f3b97daedfb57f79	Hari-Project-Instance	80	Running	Hari-Project-Instance-SG	us-east-1a	10.0.0.141	subnet-03102b1a7a3c9345f	February 8, 2024

1 pending

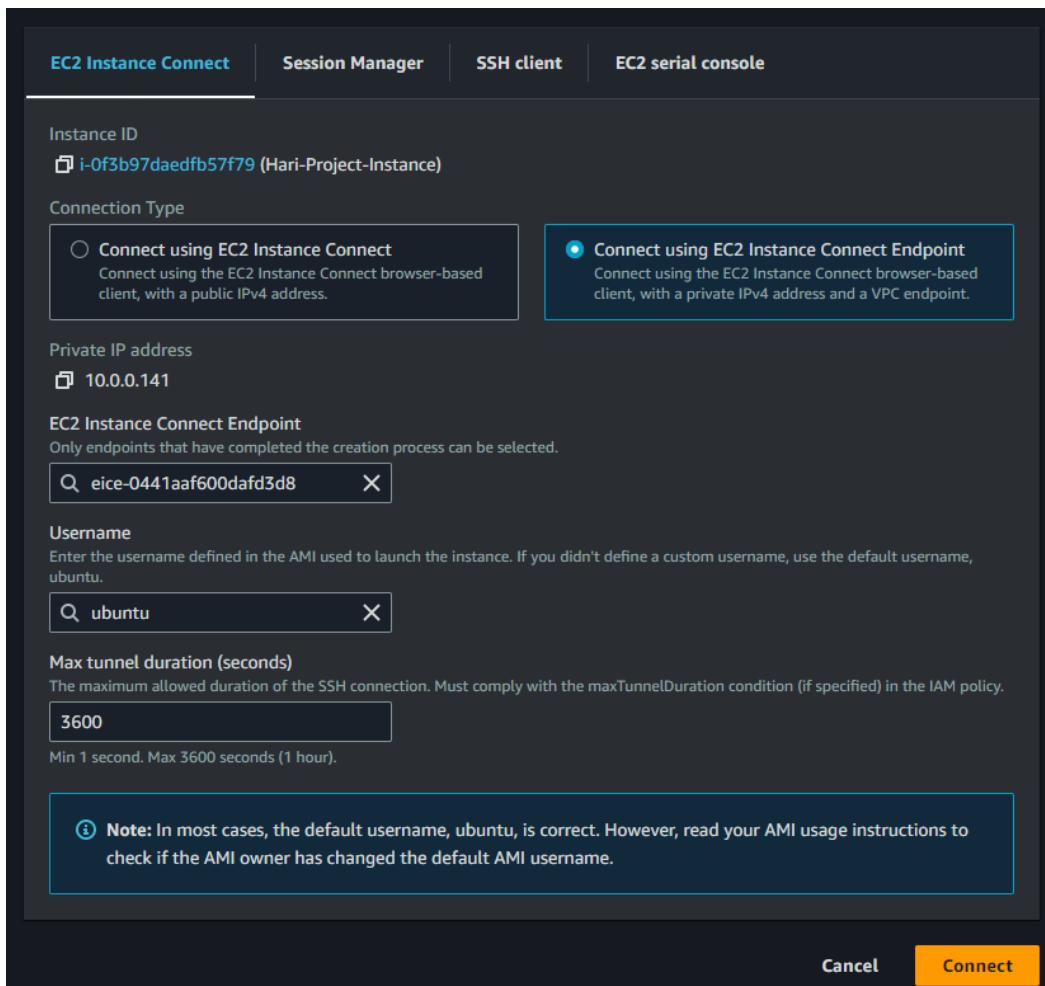
**Cancel** **Register pending targets**

- We could see the highlighted message that target registered successfully.

One target registered successfully to Project-EC2-Target-Group.

## Task – 11: Installing Apache-2 web server in EC2 instance and running it.

- First, we will go to the EC2 instance, select it and will click on Connect.
- We will connect this instance with the help of endpoint which is created earlier.



3. First, we will update the machine by using command: **sudo apt-get update**.

```
individual files in /usr/share/doc/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-0-141:~$ sudo apt-get update
```

i-0f3b97daedfb57f79 (Hari-Project-Instance)  
PrivateIPs: 10.0.0.141

4. Now will install apache2 webserver by command: **sudo apt-get install apache2 -y**

```

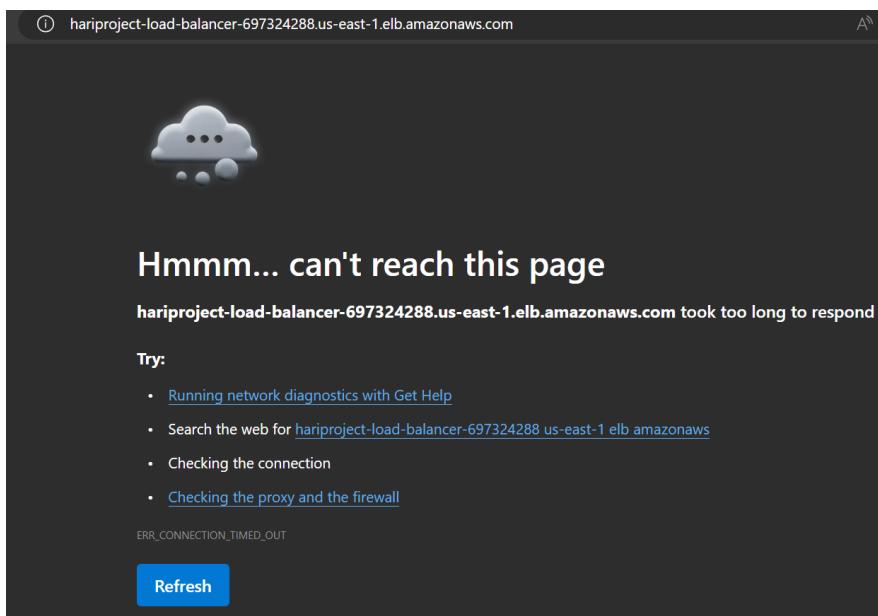
Reading package lists... Done
ubuntu@ip-10-0-0-141:~$ sudo apt-get install apache2 -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils bzip2 libapr1 libaprutil1 libaprutil1-dbd-sqlite3
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser bzip2-doc
The following NEW packages will be installed:

  i-0f3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

```

5. Now the server is installed, we will see the apache2 default web page with the help of load balancer URL.
6. We will copy the dns name of the load balancer and paste it in the browser.
7. The DNS name of the load balancer is **HariProject-Load-Balancer-697324288.us-east-1.elb.amazonaws.com**
8. But we are getting the below error, like the client is not able to reach the server.



9. Now will go the security section of the load balancer, and will click on the security group.

Listeners and rules	Network mapping	<b>Security</b>	Monitoring	Integrations	Attributes	Tags
<b>Security groups (1)</b>						
A security group is a set of firewall rules that control the traffic to your load balancer.						
Security Group ID	▼	Name	▼	Description		
sg-04f715cb41c7f8d33	[	default		default VPC security group		

10. Now will go to the Edit inbound rule.

The screenshot shows the AWS EC2 Security Groups page. The top navigation bar includes 'EC2' and 'Security Groups'. Below it, the title 'sg-04f715cb41c7f8d33 - default' is displayed. On the right, there is an 'Actions' dropdown menu. The main area is titled 'Details' and contains the following information:

Security group name	default	Security group ID	sg-04f715cb41c7f8d33	Description	default VPC security group
Owner	101304436132	Inbound rules count	1 Permission entry	Outbound rules count	1 Permission entry
VPC ID vpc-0cb9901f4b2b3d69f					

Below the details, there are three tabs: 'Inbound rules' (selected), 'Outbound rules', and 'Tags'. Under 'Inbound rules', the heading 'Inbound rules (1)' is shown. A search bar and a toolbar with icons for 'C' (Create), 'Manage tags', and 'Edit inbound rules' are present. The table lists one rule:

Name	Security group rule ID	Type	Protocol	Port range	Source	Description
-	sgr-0edfa00e4e55ce325	All traffic	All	All	All traffic	sg-04f715cb41c7f8d33

11. This is the current configuration in below snip.

The screenshot shows the 'Edit inbound rules' page for the default security group. The title is 'Edit inbound rules' with a 'Info' link. Below it, a note says 'Inbound rules control the incoming traffic that's allowed to reach the instance.' The main area is titled 'Inbound rules' with an 'Info' link. It displays a table with one rule:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
sgr-0edfa00e4e55ce325	All traffic	All	All	Custom	sg-04f715cb41c7f8d33

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

12. After modification the rule in security group is in the below snip. Now will click on Save rules.

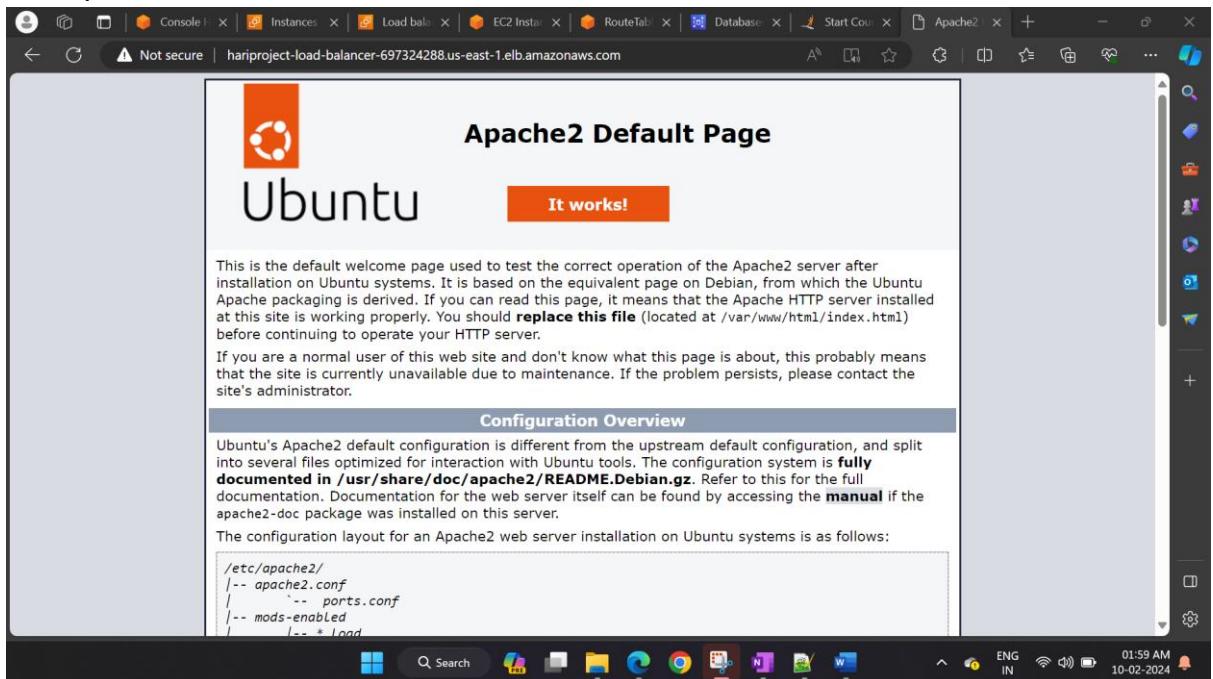
The screenshot shows the 'Edit inbound rules' page after modifying the rule. The title is 'Edit inbound rules' with a 'Info' link. Below it, a note says 'Inbound rules control the incoming traffic that's allowed to reach the instance.' The main area is titled 'Inbound rules' with an 'Info' link. It displays a table with one rule:

Security group rule ID	Type	Protocol	Port range	Source	Description - optional
-	HTTP	TCP	80	Anyw...	0.0.0.0/0

A warning message at the bottom states: '⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.' Buttons at the bottom include 'Add rule', 'Cancel', 'Preview changes', and 'Save rules'.

13. Now we will repeat the steps 6 & 7 and paste the DNS name **HariProject-Load-Balancer-697324288.us-east-1.elb.amazonaws.com** of

the load balancer in the browser. And we could see the default page of the apache2 server.



14. So, testing of apache2 server is done. Now we will run our code which is required in the assignment.

## Task – 12: Downloading the PHP code in EC2 instance & running it.

1. We will go the console of EC2 instance. And will run the command:

**wget <https://lms.intellipaat.com/mediaFiles/2020/10/code.zip>**

And will do ls to see the code.zip file.

```
ubuntu@ip-10-0-0-141:~$ wget https://lms.intellipaat.com/mediaFiles/2020/10/code.zip
--2024-02-09 20:34:56-- https://lms.intellipaat.com/mediaFiles/2020/10/code.zip
Resolving lms.intellipaat.com (lms.intellipaat.com)... 104.18.27.176, 104.18.26.176, 2606:4700::6812:1ab0, ...
Connecting to lms.intellipaat.com (lms.intellipaat.com)|104.18.27.176|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 787844 (769K) [application/zip]
Saving to: 'code.zip'

code.zip                                              0%[=====] 769.38K  --.-KB/s   in 0.006s

2024-02-09 20:34:56 (124 MB/s) - `code.zip' saved [787844/787844]

ubuntu@ip-10-0-0-141:~$ ls
code.zip
ubuntu@ip-10-0-0-141:~$ 
```

i-0f3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

2. Now will install unzip package to unzip the code.zip file by command  
**sudo apt-get install unzip -y**

```
ubuntu@ip-10-0-0-141:~$ ls
code.zip
ubuntu@ip-10-0-0-141:~$ sudo apt-get install unzip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
```

i-Of3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

3. Now will unzip the code.zip file by command: **unzip code.zip**

```
ubuntu@ip-10-0-0-141:~$ unzip code.zip
Archive: code.zip
  creating: 1243/images/
  inflating: 1243/images/1.png
  inflating: 1243/images/2.png
  inflating: 1243/index.php
ubuntu@ip-10-0-0-141:~$ ls
1243  code.zip
ubuntu@ip-10-0-0-141:~$
```

i-Of3b97daedfb57f79 (Hari-Project-Instance)

4. Now will go to another folder 1243 after unzip and will do ls to see the files in this folder.

- **cd 1243**
- **ls**

```
ubuntu@ip-10-0-0-141:~$ ls
1243  code.zip
ubuntu@ip-10-0-0-141:~$ cd 1234/
-bash: cd: 1234/: No such file or directory
ubuntu@ip-10-0-0-141:~$ cd 1243/
ubuntu@ip-10-0-0-141:~/1243$ ls
images  index.php
ubuntu@ip-10-0-0-141:~/1243$
```

i-Of3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

5. Now will move all this file to the var/www/html by command:

**sudo mv \* /var/www/html**, And could see the file reached in that folder.

```

ubuntu@ip-10-0-0-141:~$ ls
1243 code.zip
ubuntu@ip-10-0-0-141:~$ cd 1234/
-bash: cd: 1234/: No such file or directory
ubuntu@ip-10-0-0-141:~$ cd 1243/
ubuntu@ip-10-0-0-141:~/1243$ ls
images index.php
ubuntu@ip-10-0-0-141:~/1243$ sudo mv * /var/www/html
ubuntu@ip-10-0-0-141:~/1243$ cd /var/www/html
ubuntu@ip-10-0-0-141:/var/www/html$ ls
images index.html index.php
ubuntu@ip-10-0-0-141:/var/www/html$ []

```

i-Of3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

6. Now will remove the index.html file from here. So that other file data can be shown.
7. Now we could see the expected output through the load balancer DNS URL.

The screenshot shows a web application interface. At the top right is the Intellipaat logo. Below it is a form with two input fields: 'Name:' and 'Email:', each with a corresponding text input box. A green 'Submit' button is positioned below the email field. The background of the page is white with some light blue decorative elements like triangles and circles.

connect\_error) { die("Connection failed: " . \$conn->connect\_error); } if(isset(\$\_POST['firstname']) && isset(\$\_POST['email'])){ \$sql = "INSERT INTO data (firstname,email) VALUES ('".\$\_firstname.", '".\$\_email."'); if (\$conn->query(\$sql) === TRUE) { echo "New record created successfully"; } else { echo "Error: " . \$sql . " " . \$conn->error; } \$conn->close(); } ?>

## Task -13: Making a PHP server & running PHP website.

1. Now we could see there is some connection error in the above snip. Because we are running a PHP website on a HTML server. So, we need to make a PHP server to run the PHP website.
2. We will use below two commands:
3. **sudo add-apt-repository -y ppa:ondrej/php**
4. **sudo apt install php5.6 mysql-client php5.6-mysqli -y**

```

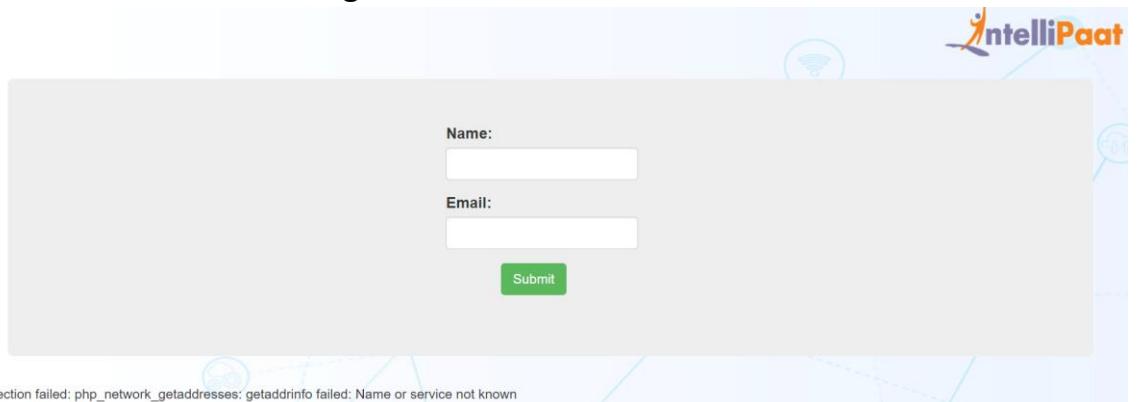
ubuntu@ip-10-0-0-141:~$ cd /var/www/html
ubuntu@ip-10-0-0-141:/var/www/html$ sudo add-apt-repository -y ppa:ondrej/php
sudo apt install php5.6 php5.6-mysqli -y
PPA publishes dbgsym, you may need to include 'main/debug' component
Repository: 'deb https://ppa.launchpadcontent.net/ondrej/php/ubuntu/ jammy main'
Description:
Co-installable PHP versions: PHP 5.6, PHP 7.x, PHP 8.x and most requested extensi
tp://php.net/supported-versions.php) for Supported Ubuntu Releases (https://wiki.
-of-life PHP versions or Ubuntu release, they won't be provided.

Debian oldstable and stable packages are provided as well: https://deb.sury.org/#

You can get more information about the packages at https://deb.sury.org

```

- Now the error for running the website is resolved.



## Task – 13: Connecting to RDS database.

- Now we could see that the other error like the database is not connected.
- We will modify the index.php file with the correct credentials of the database.

```

<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "intelli.coghw13fheqo.us-east-2.rds.amazonaws.com";
$username = "intel";
$password = "intel123";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);

```

- Servername will be endpoint of the database:  
**hari-project-mysql-rds.cn2j180cuva.us-east-1.rds.amazonaws.com**
- Username = admin
- Password = admin1234
- Now after modification we will save the index.php file.

```

</div>
<?php
$firstname=$_POST['firstname'];
$email=$_POST['email'];
$servername = "hari-project-mysql-rds.cn2j180cuva.us-east-1.rds.amazonaws.com";
$username = "admin";
$password = "admin1234";
$db = "intel";
// Create connection
$conn = new mysqli($servername, $username, $password, $db);

```

7. Now we will get exit from the index.php file. & will run DNS again in the browser.
8. And now we are getting connection time out error.



## Task – 14: Changing Security Group rule of the MySQL RDS database.

1. Now we will open the security group of the RDS Database **hari-project-mysql-rds**, which is created earlier in this project.

A screenshot of the AWS EC2 Security Groups page. The top navigation bar shows 'EC2 > Security Groups > sg-07f79c8d3d98f161b - Hari-RDS-SG'. The main section is titled 'sg-07f79c8d3d98f161b - Hari-RDS-SG'. Below it is a 'Details' card with the following information:

Security group name	Hari-RDS-SG	Security group ID	sg-07f79c8d3d98f161b	Description	Created by RDS management console	VPC ID	vpc-0cb9901f4b2b3d69f
Owner	101304436132	Inbound rules count	1 Permission entry	Outbound rules count	1 Permission entry		

Below the details are tabs for 'Inbound rules', 'Outbound rules', and 'Tags'. The 'Inbound rules' tab is selected, showing a table with one rule:

Name	Security group rule...	IP version	Type	Protocol	Port range
-	sqr-06c5b8d02a3cd7712	IPv4	MySQL/Aurora	TCP	3306

2. We will click on the Edit inbound rule. And could see the current rule of the firewall.

The screenshot shows the 'Edit inbound rules' page for a security group. A single rule is listed:

- Security group rule ID:** sgr-06c5b8d02a3cd7712
- Type:** MySQL/Aurora
- Protocol:** TCP
- Port range:** 3306
- Source:** 49.15.230.115/32

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

- Now we will be taking source as instance security group. We could see the below snip after changing security group rule. Now will click on Save rules.

The screenshot shows the 'Edit inbound rules' page for a security group. A single rule is listed:

- Security group rule ID:** -
- Type:** All traffic
- Protocol:** All
- Port range:** All
- Source:** sg-0bf4127f36b0f1563

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

- Now will run DNS again in the browser. And could see the below output.

The screenshot shows a web browser displaying an error message: "Connection failed: Unknown database 'intel'". The page contains form fields for "Name:" and "Email:", and a "Submit" button.

- Now we are getting the error is Connection failed: Unknown database 'intel'.

## Task – 15: Installing MySQL client for RDS database.

- Now we are installing mysql-client by using command:  
**sudo apt-get install mysql-client -y**

```
ubuntu@ip-10-0-0-141:/var/www/html$ sudo apt-get install mysql-client -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

i-0f3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141
```

## Task – 16: Connecting to MySQL RDS database.

1. We will use the below command for connection.

```
mysql -h hari-project-mysql-rds.cnn2j180cuva.us-east-1.rds.amazonaws.com -u admin -p
```

2. Now we could see we have connected the MySQL database.

```
ubuntu@ip-10-0-0-141:/var/www/html$ mysql -h hari-project-mysql-rds.cnn2j180cuva.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 119
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> i-0f3b97daedfb57f79 (Hari-Project-Instance)
PrivateIPs: 10.0.0.141
```

## Task – 17: Creation of intel mysql database.

1. We will use the command to create database:

```
Create database intel;
```

```
mysql> create database intel;
Query OK, 1 row affected (0.00 sec)

mysql> show database;
```

2. Now we could see the intel database using command:

```
Show databases;
```

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| intel          |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.00 sec)

mysql> []
i-0f3b97daedfb57f79 (Hari-Project-Instance)
Private IPs: 10.0.0.141
```

3. Now will run DNS again in the browser. And could see the below output with no error.

The screenshot shows a web page with a light blue header containing the IntelliPaat logo. Below the header is a form with two input fields: 'Name:' and 'Email:', each with a corresponding text input box. A green 'Submit' button is positioned below the email field. The background of the page features a faint, stylized network or cloud-like diagram.

4. Now we try to enter the record in the database.

The screenshot shows the same web page as the previous one, but now with an error message displayed at the bottom of the page area. The message reads: "Error: INSERT INTO data (firstname,email) VALUES ('Hari', 'hari@gmail.com') Table 'intel.data' doesn't exist". The rest of the page, including the form and background, remains the same.

5. Now we are getting the error that the table does not exist.

## Task – 18: Creation of table data in intel mysql database and entering the records.

1. We will use the below command to create the data table.
  - use intel;
  - create table data (firstname varchar(20), email varchar(25));

```
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| intel          |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0.00 sec)

mysql> use intel;
Database changed
mysql> create table data (firstname varchar(20), email varchar(25));
Query OK, 0 rows affected (0.02 sec)

mysql> []
```

i-0f3b97daedfb57f79 (Hari-Project-Instance)

PrivateIPs: 10.0.0.141

2. Now will run DNS again in the browser. And will enter some records and could see the below output that - new record created successfully.

The screenshot shows a web page with a header containing the IntelliPaat logo. Below the header is a form with two input fields: 'Name:' and 'Email:', each with a corresponding text input box. A green 'Submit' button is located below the email field. At the bottom of the page, there is a success message: 'New record created successfully'.

3. Now will check the table **intel.data** that the record has entered. And could see in the below snip that the record has been entered.

```

mysql> select * from data;
+-----+-----+
| firstname | email           |
+-----+-----+
| Hari      | hari@gmail.com   |
| Rishit    | rishitsingh@gmail.com |
+-----+-----+
2 rows in set (0.00 sec)

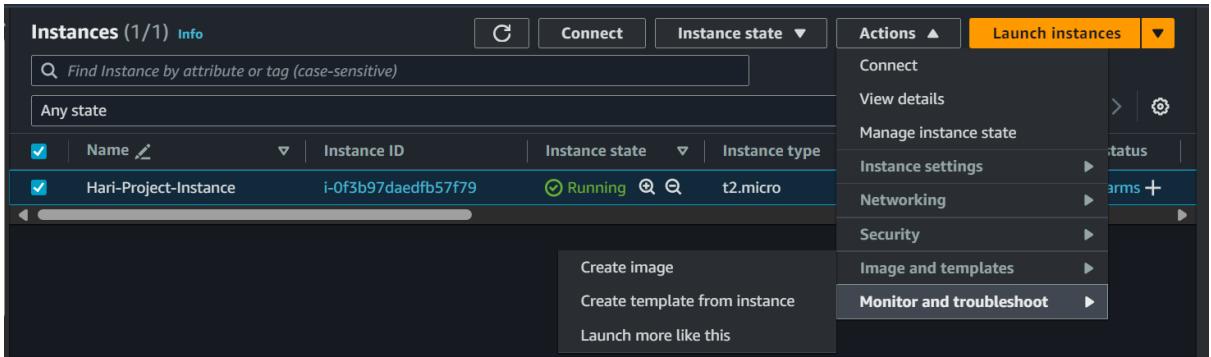
mysql> []

```

**i-0f3b97daedfb57f79 (Hari-Project-Instance)**  
Private IPs: 10.0.0.141

## Task – 19: Creation of Image of EC2 Instance.

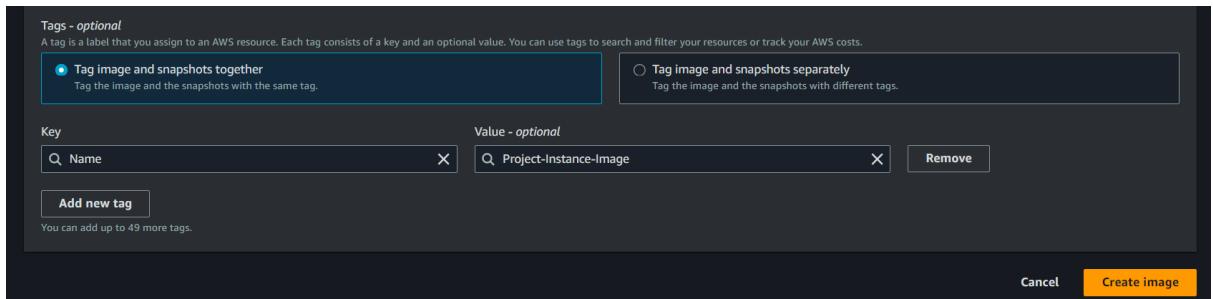
- Now we will go to the EC2 dashboard, will select the Hari-Project-Instance → Actions → Image and templates → Create image.



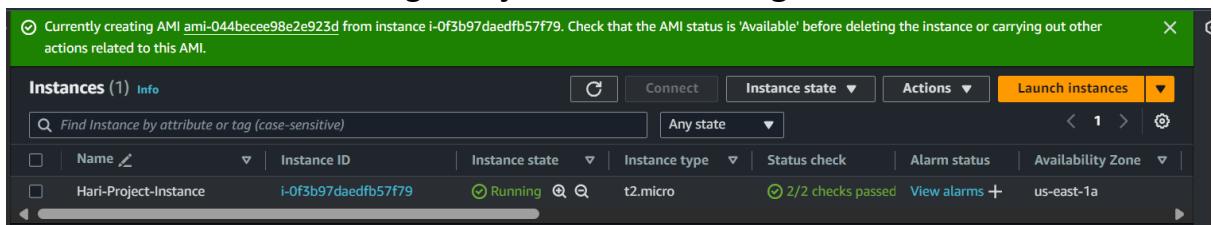
- Now we have given name to the image is **Project-Instance-Image**.

Instance ID	i-0f3b97daedfb57f79 (Hari-Project-Instance)
Image name	Project-Instance-Image
Image description - optional	Image
No reboot	

- We have given tag to the image. Now will click on Create image.

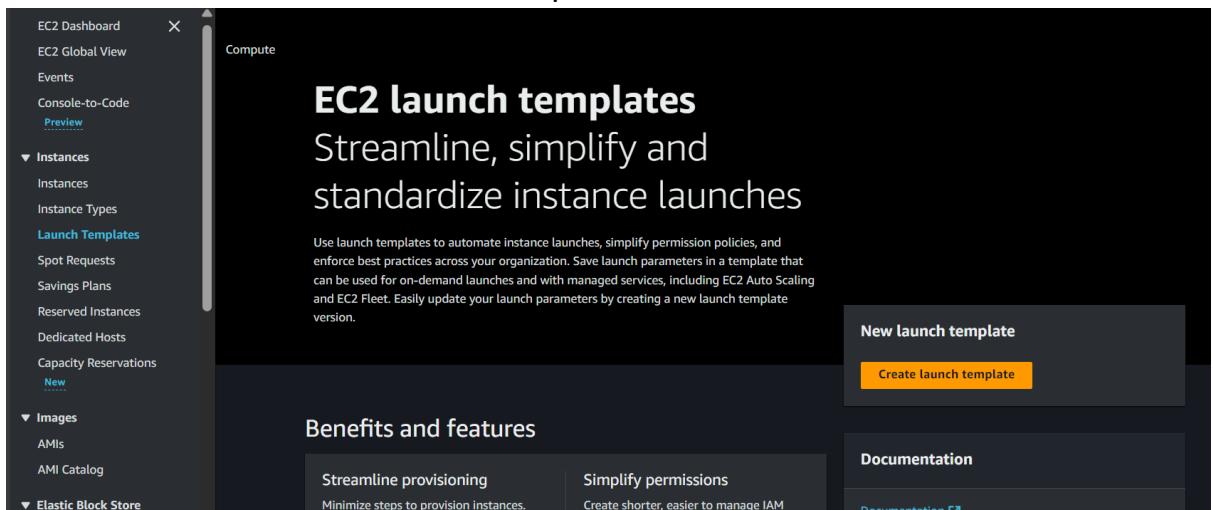


- Now we could see the image **Project-Instance-Image** has been created.



## Task – 20: Creation of Launch Template.

- Now from the left panel of EC2 dashboard, click on the Launch Templates.
- Then will click on Create launch template.



- We have given name to the launch template is **Hari-Project-Launch-Template**.

## Create launch template

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

### Launch template name and description

Launch template name - required

Hari-Project-Launch-Template

Must be unique to this account. Max 128 chars. No spaces or special characters like '&', '\*', '@'.

Template version description

Project-LT

Max 255 chars

Auto Scaling guidance | [Info](#)

Select this if you intend to use this template with EC2 Auto Scaling

Provide guidance to help me set up a template that I can use with EC2 Auto Scaling

4. We will select the AMI which we have created.

### ▼ Application and OS Images (Amazon Machine Image) - required [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

 *Search our full catalog including 1000s of application and OS images*

Recents

My AMIs

Quick Start

Owned by me

Shared with me



Browse more AMIs

Including AMIs from AWS, Marketplace and the Community

#### Amazon Machine Image (AMI)

Project-Instance-Image

ami-044becee98e2e923d

2024-02-10T18:09:41.000Z Virtualization: hvm ENA enabled: true Root device type: ebs

Description

The screenshot shows the 'Instance type' section of the AWS Lambda configuration interface. It highlights the 't2.micro' instance type, which is described as 'Free tier eligible'. Below the instance type, there is a note about additional costs for AMIs with pre-installed software. To the right, there are buttons for 'All generations' and 'Compare instance types'.

**Instance type**

t2.micro      Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

**Key pair (login)**

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name

Instance-key-pair

Create new key pair

5. In Network settings we are not giving any subnet as it all will be taken care through image. And selected the existing group which is Hari-Project-Instance-SG.

The screenshot shows the 'Network settings' section of the AWS Lambda configuration interface. Under 'Subnet', it says 'Don't include in launch template'. There is a note that if a subnet is not specified, a network interface will be automatically added. A 'Create new subnet' button is available. Under 'Firewall (security groups)', it says 'Select existing security group' is selected. A 'Create security group' button is also present. Below this, a list of security groups is shown, with 'Hari-Project-Instance-SG' selected. A note indicates the VPC ID and a 'Compare security group rules' link are also present. An 'Advanced network configuration' link is at the bottom.

**Network settings**

Subnet

Don't include in launch template

Create new subnet

Firewall (security groups)

Select existing security group

Create security group

Security groups

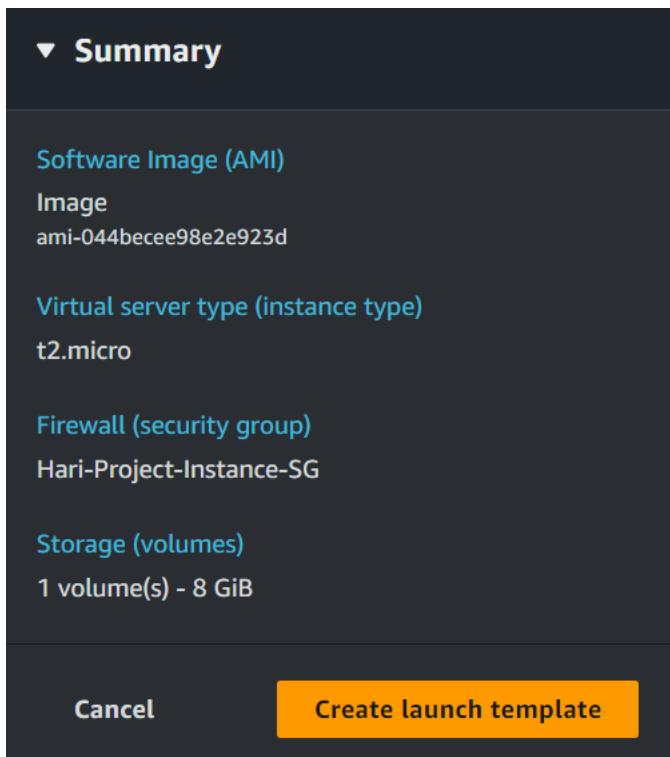
Hari-Project-Instance-SG sg-0bf4127f36b0f1563 X

VPC: vpc-0cb9901f4b2b3d69f

Compare security group rules

Advanced network configuration

6. Now everything will be as default & will click on Create launch template.



- Now we could see the launch template has been created.

Launch Templates (1) <a href="#">Info</a>					<a href="#">Actions</a>	<a href="#">Create launch template</a>
<a href="#">Search</a>					< 1 >	<a href="#">Settings</a>
	Launch Template ID	Launch Template Name	Default Version	Latest Version	Create Time	
<a href="#">lt-070c6bab75ecb59f</a>	Hari-Project-Launch-Template	1	1	2024-02-10T18:24:30.00Z		

## Task – 21: Creation of Auto-Scaling Group.

- Now from the left panel of EC2 dashboard, click on the Auto Scaling Groups.
- Then will click on Create Auto Scaling Group.

- We have given name to the group is **Hari-Project-Auto-Scaling-Group**.

The screenshot shows the 'Choose launch template' step of the 'Create Auto Scaling group' wizard. On the left, a vertical navigation bar lists steps: Step 1 (Choose launch template), Step 2 (Choose instance launch options), Step 3 (optional: Configure advanced options), and Step 4 (optional: Configure group size and scaling). The main area is titled 'Choose launch template' with a 'Info' link. It instructs the user to specify a launch template containing settings common to all EC2 instances. A 'Name' field is shown with the value 'Hari-Project-Auto-Scaling-Group'. Below it, a note states: 'Must be unique to this account in the current Region and no more than 255 characters.'

4. And selected the launch template which is above created.

The screenshot shows the 'Launch template' configuration page. It includes a note: 'For accounts created after May 31, 2023, the EC2 console only supports creating Auto Scaling groups with launch templates. Creating Auto Scaling groups with launch configurations is not recommended but still available via the CLI and API until December 31, 2023.' Below this, a 'Launch template' section shows a dropdown menu set to 'Hari-Project-Launch-Template' with a '▼' icon and a 'C' icon for cloning. A 'Create a launch template' button is also present. A 'Version' section shows 'Default (1)' with a '▼' icon and a 'C' icon.

5. Now we could see all the details regarding the template. And will click on Next.

The screenshot shows the 'Additional details' step of the wizard. It displays configuration details: 'ami-044becee98e2e923d' (Instance ID), 'Key pair name: Instance-key-pair', 'Security group IDs: sg-0bf4127f36b0f1563' (with a copy icon), and 'Additional details' showing 'Storage (volumes): -' and 'Date created: Sat Feb 10 2024 23:54:30 GMT+0530 (India Standard Time)'. At the bottom right are 'Cancel' and 'Next' buttons.

6. On Choose instance launch options page will scroll down and will select the VPC which we have created in this project. And will select the subnet in which the instance has been created. And now will click on Next.

**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-0cb9901f4b2b3d69f (VPC-Project-vpc) ▾ C  
10.0.0.0/24

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 ▾ C

us-east-1a | subnet-03102b1a7a3c9345f (VPC-Project-subnet-private1-us-east-1a) X  
10.0.0.128/28

us-east-1b | subnet-0e2ad0ed601f8fa0c (VPC-Project-subnet-private2-us-east-1b) X  
10.0.0.144/28

Create a subnet ↗

Cancel Skip to review Previous Next

7. On Configure advanced options – optional page, we will select the option of Attach to an existing load balancer.

**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 create a basic load balancer to attach to your Auto Scaling group.

8. Now will select the target group which is created earlier in this project.

**Attach to an existing load balancer**

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

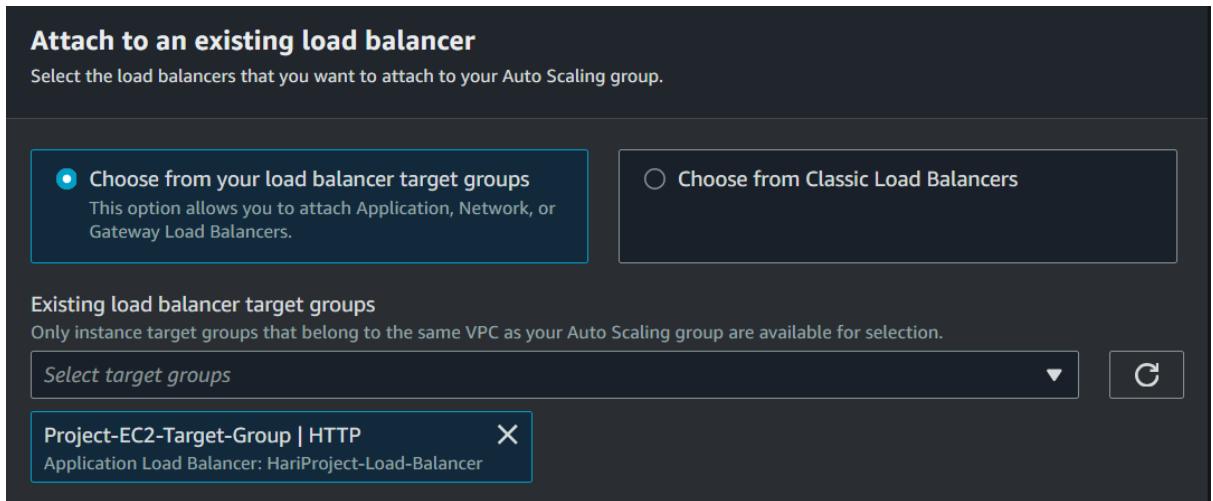
Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.

Choose from Classic Load Balancers

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

Select target groups ▾ ✖ 

Project-EC2-Target-Group | HTTP ✖  
Application Load Balancer: HariProject-Load-Balancer



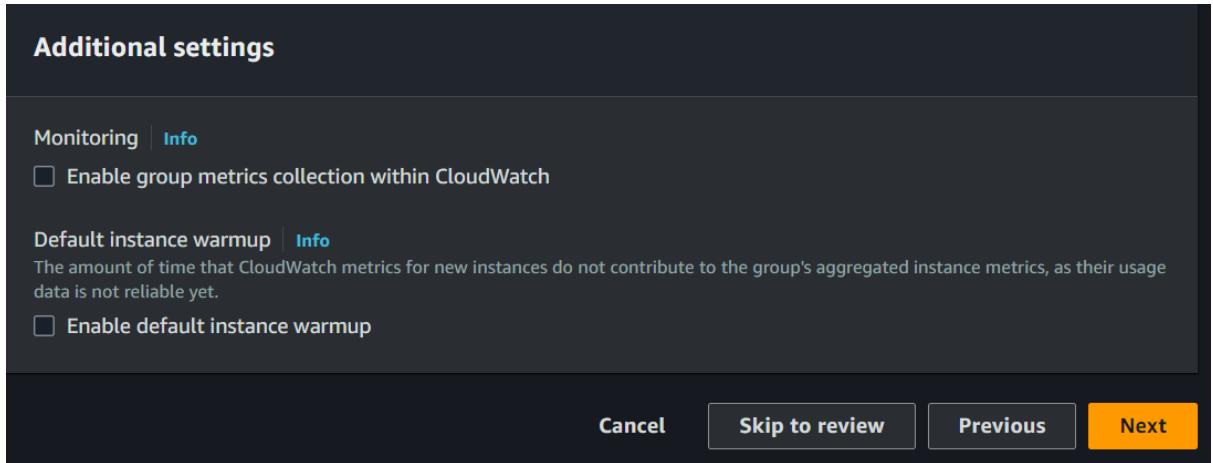
9. Now will scroll down and click on Next.

**Additional settings**

Monitoring | [Info](#)  
 Enable group metrics collection within CloudWatch

Default instance warmup | [Info](#)  
The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.  
 Enable default instance warmup

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)



10. On Configure group size and scaling – optional page we have taken  
Desired capacity is 2. Min is 2 & Max is 5.

**Group size** Info

Set the initial size of the Auto Scaling group. After creating the group, you can change its size to meet demand, either manually or by using automatic scaling.

---

**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** Info

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	5
Equal or less than desired capacity	Equal or greater than desired capacity

11. In Target tracking policies, we have taken target value is 85, means when CPU utilization will go to above 85 % then the new instances will be launched.

**Automatic scaling - optional**

Choose whether to use a target tracking policy | [Info](#)  
You can set up other metric-based scaling policies and scheduled scaling after creating your Auto Scaling group.

**No scaling policies**  
Your Auto Scaling group will remain at its initial size and will not dynamically resize to meet demand.

**Target tracking scaling policy**  
Choose a CloudWatch metric and target value and let the scaling policy adjust the desired capacity in proportion to the metric's value.

Scaling policy name

Metric type | [Info](#)  
Monitored metric that determines if resource utilization is too low or high. If using EC2 metrics, consider enabling detailed monitoring for better scaling performance.  
 ▾

Target value

Instance warmup | [Info](#)  
 seconds

Disable scale in to create only a scale-out policy

12. Now will scroll down and will click on Next.

**Instance scale-in protection**

Scale-in protection prevents newly launched instances from being terminated by scaling activities. Make sure to remove scale-in protection for the group or individual instances when instances are ready to be terminated.

Enable instance scale-in protection

[Cancel](#) [Skip to review](#) [Previous](#) [Next](#)

13. Next.

**Add notifications - optional** [Info](#)

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](#)

14. We have added tags and will click on Next.

## Add tags - optional Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

Tags (1)			
Key	Value - optional	Tag new instances	Remove
Name	Hari-Project-Auto-Scaling-Group	<input checked="" type="checkbox"/>	Remove
<a href="#">Add tag</a>			
49 remaining			

Cancel   Previous   Next

15. Now on Review page, will be reviewing everything

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](#)

### Review Info

Step 1: Choose launch template [Edit](#)

Group details		
Auto Scaling group name	Hari-Project-Auto-Scaling-Group	
Launch template	Version	Description
Hari-Project-Launch-Template <a href="#">Edit</a>	Default	Project-LT

Step 2: Choose instance launch options [Edit](#)

Network	
Network	

16. And will scroll down and Click on Create Auto Scaling Group.

Step 6: Add tags [Edit](#)

Tags (1)		
Key	Value	Tag new instances
Name	Hari-Project-Auto-Scaling-Group	Yes

Cancel   Previous   [Create Auto Scaling group](#)

17. Now we could see the auto scaling group has been created.

Auto Scaling groups (1) <a href="#">Info</a>					
<a href="#">C</a>	<a href="#">Launch configurations</a>	<a href="#">Launch templates</a>	<a href="#">Actions</a> ▾	<a href="#">Create Auto Scaling group</a>	
<a href="#">Search your Auto Scaling groups</a>					
<input type="checkbox"/>	Name	Launch template/configuration	Instances	Status	Desired cap
<input type="checkbox"/>	<a href="#">Hari-Project-Auto-Scaling-Group</a>	Hari-Project-Launch-Template   Version D	2	-	2

18. Now will click on the auto scaling group and in the activity section we could see that the two instances have been launched as per the desired capacity.

Activity history (2)		
<a href="#">Status</a> ▾	<a href="#">Description</a> ▾	<a href="#">Cause</a>
<span>⌚ Successful</span>	Launching a new EC2 instance: i-0009179160b44c315	At 2024-02-10T18:54:27Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-02-10T18:54:31Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.
<span>⌚ Successful</span>	Launching a new EC2 instance: i-09e022dc53d85bbe3	At 2024-02-10T18:54:27Z a user request created an AutoScalingGroup changing the desired capacity from 0 to 2. At 2024-02-10T18:54:31Z an instance was started in response to a difference between desired and actual capacity, increasing the capacity from 0 to 2.

19. Now we could say that the servers are highly available.

\*\*\*\*\* THE END \*\*\*\*\*