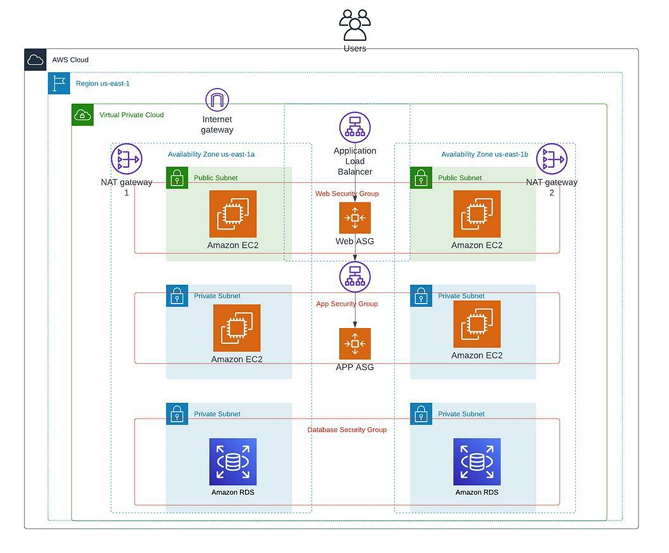
# **Creating a Highly Available 3-Tier Architecture for Web Applications in AWS**

AWS provides a wide range of resources for developing and managing cloud applications, which can be customized to construct highly dependable and resilient cloud infrastructures. Suppose you are tasked with developing a three-tier architecture that is readily available for your organization’s new web application. This tutorial is extensive but comprehensive. You may want to bookmark this guide for future reference on creating web, application, and data tiers.

**What is a 3-Tier Architecture?**

A three-tier architecture comprises three layers, namely the presentation tier, the application tier, and the data tier. The presentation tier serves as the front-end, hosting the user interface, such as the website that users or clients interact with. The application tier, commonly referred to as the back-end, processes the data. Finally, the data tier is responsible for data storage and management.



**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 an AMI (image).

4. Create Autoscaling group, Create launch template.

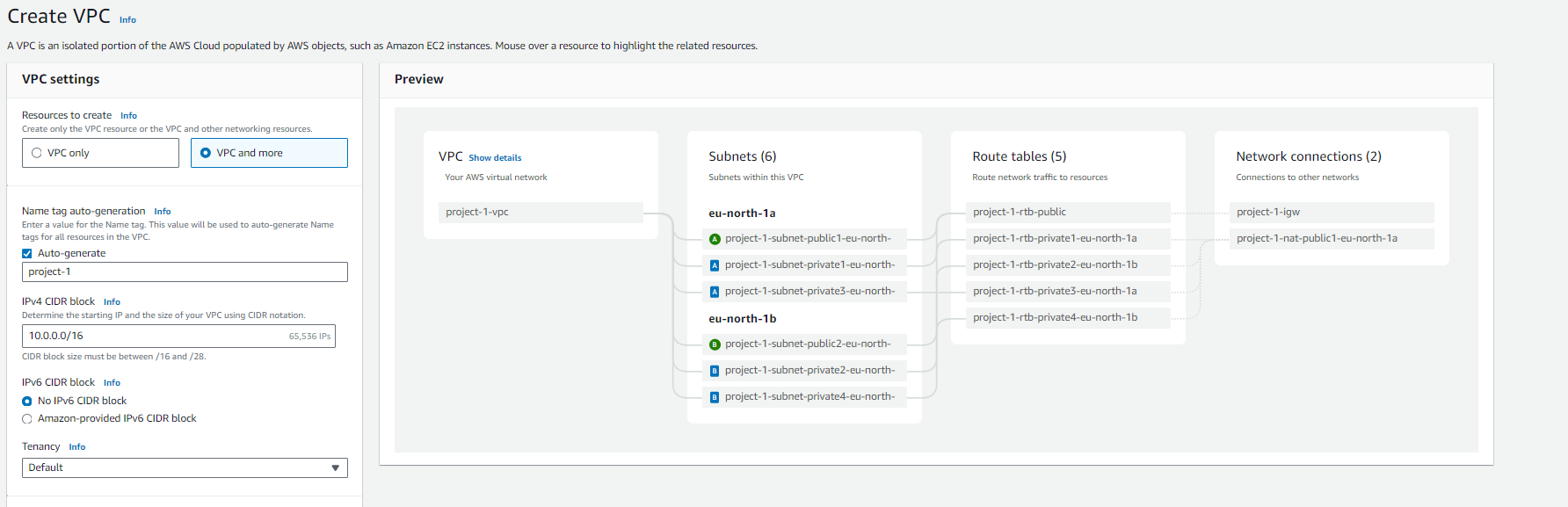
5. Create Subnet group.

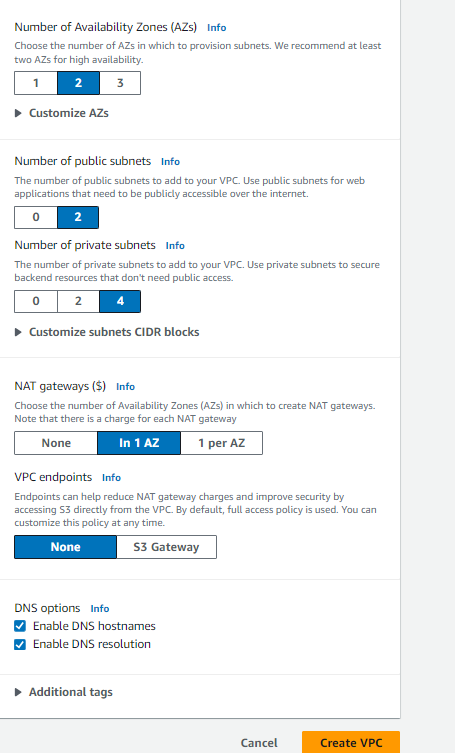
6. Create Database (RDS).

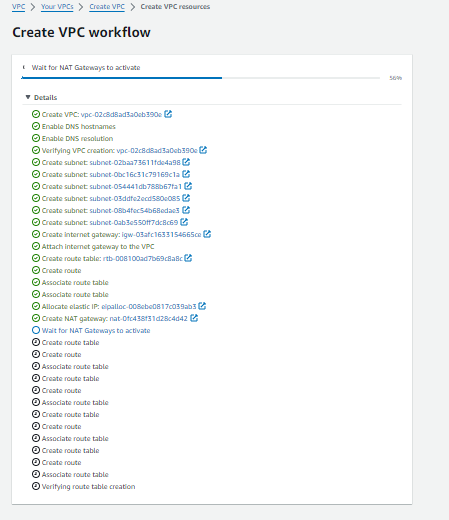
7. Establish connection.

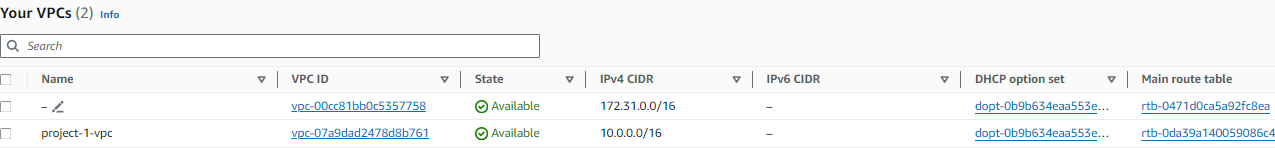
**Step 1 : Create VPC and its components**

* Goto VPC dashboard and create VPC more
* By using 6subnets 2as public and 4 as private
* 1 internet gateway connect VPC and public subnets
* Create in two availability zone’s in eu-north-1a and eu-north-1b as below in fig

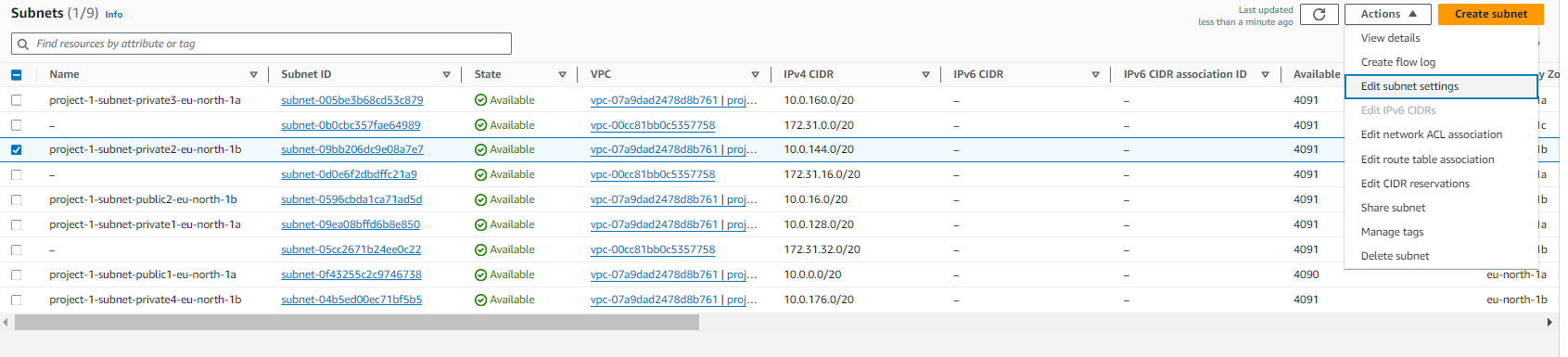


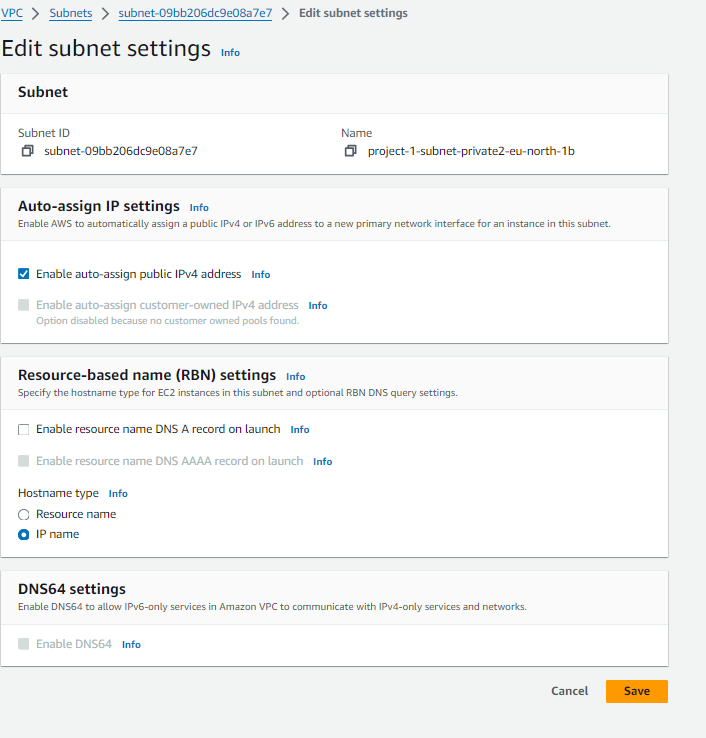






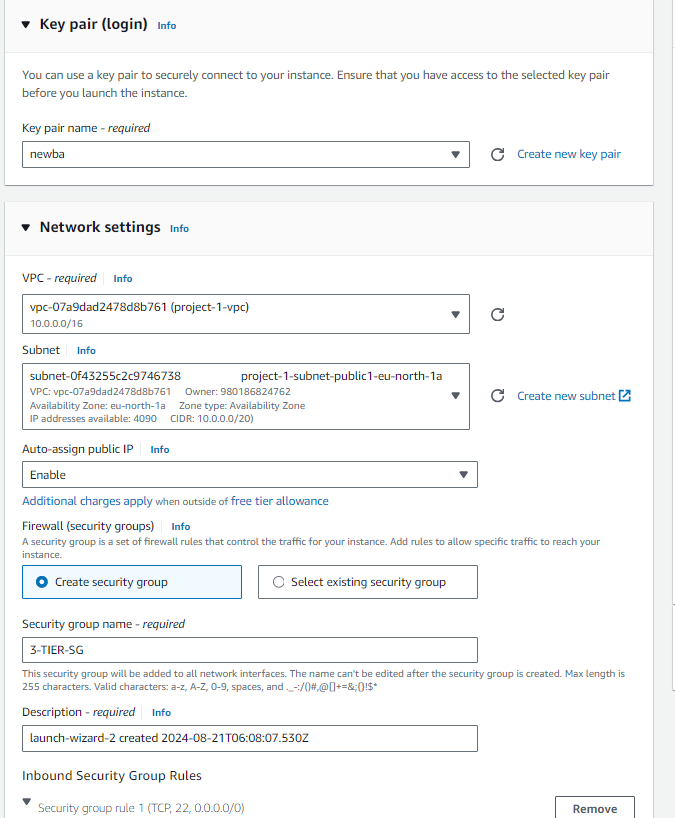
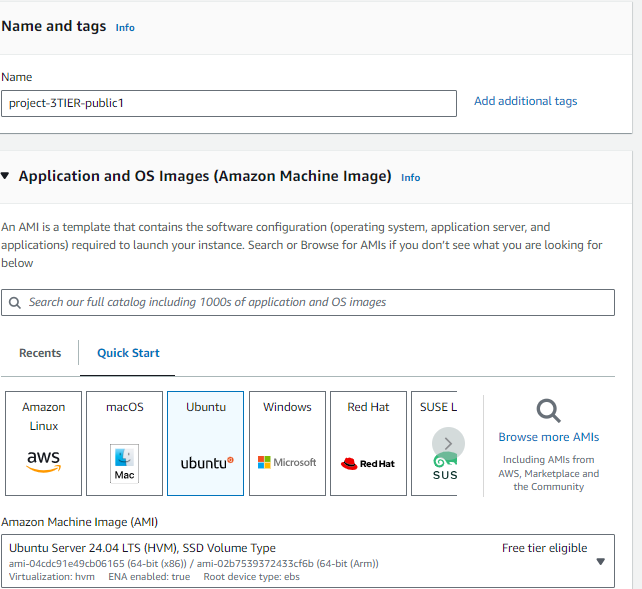
* Next click on the Subnets tab in the VPC console. Select one of the new subnets that was created, then under the “Actions” tab, expand the down arrow and select “Edit subnet settings.”
* Check “Enable auto-assign IPv4 address” and “Save.” We need to do this for all 6 new subnets that were created.

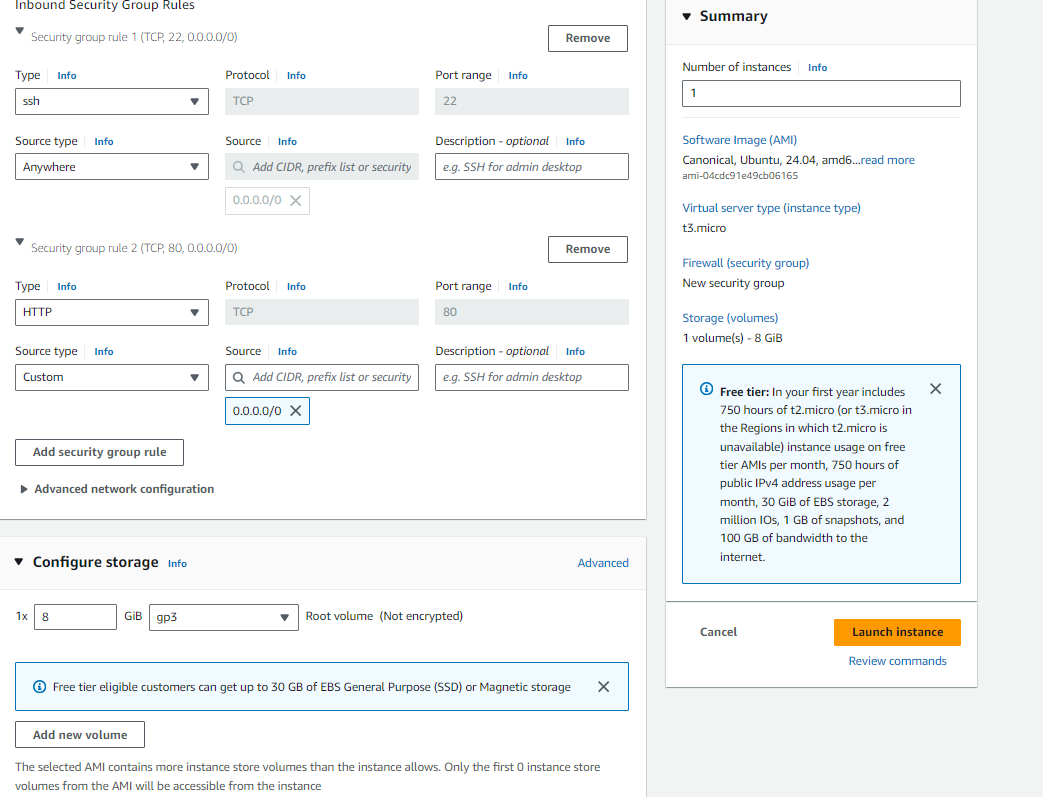




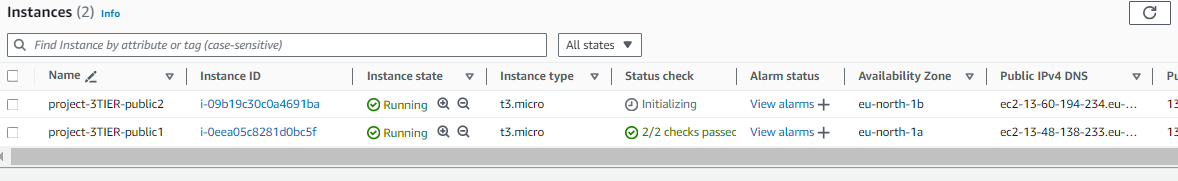
**Step-2 : Create an Ec2 instance**

* Create instance of name project-3 tier public1 and select AMI as ubuntu.
* Instance type as t2.micro and key pair as nameba .
* Click on edit network settings, select our VPC and public subnet.
* Auto assign IP enable and create a security group as 3-tier-sg.
* add http port number 80 for security group.
* click on launch instance.

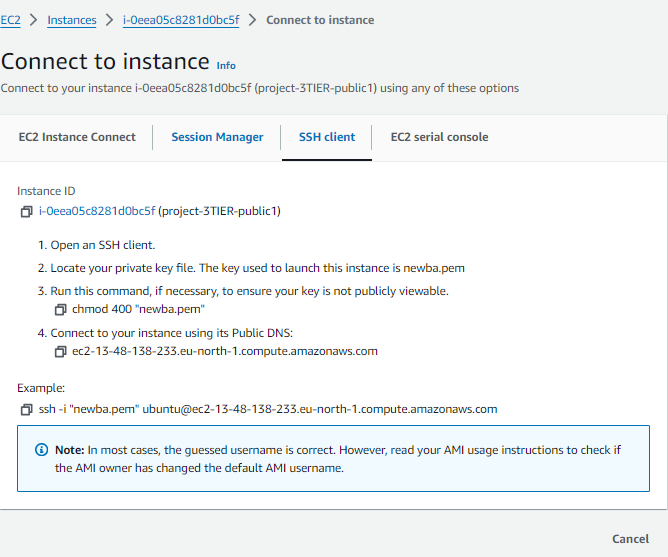




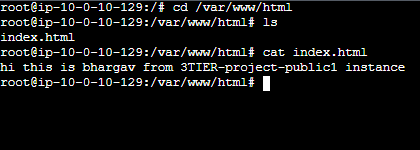
* create same instance as project-3tier-public1 using subnet as public 2 and name as project 3tier-public-2 as follow above



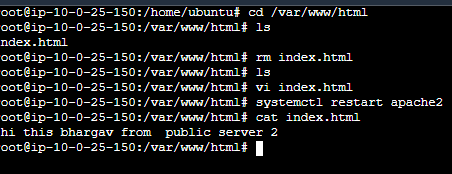
* Once created the instance click on connect and copy the ssh command in SSH client field and then go to git bash and connect to the server.



* Once connected change to root user->sudo su.
* Then update the server->apt update -y.
* Install apache2->apt install apache2 -y.
* Change directory path->cd /var/www/html.
* Remove the default index.html and create new file.
* Restart->systemctl restart apache2.
* As follows.



* By the same process connect to PROJECT-3TIER-PUBLIC2 and remove file and create new.



**Create Loadbalance for two instance:**

* For load balancer first we need to create Target group.
* Click on create Target group

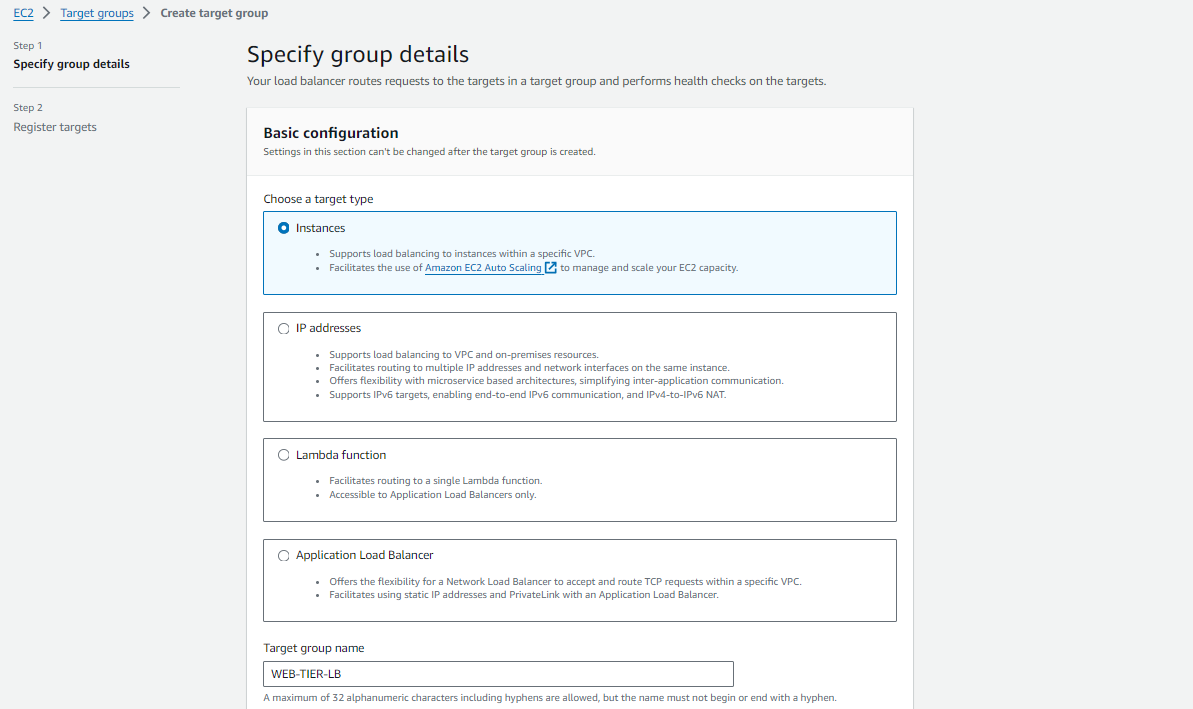
->select instances under choose target type.

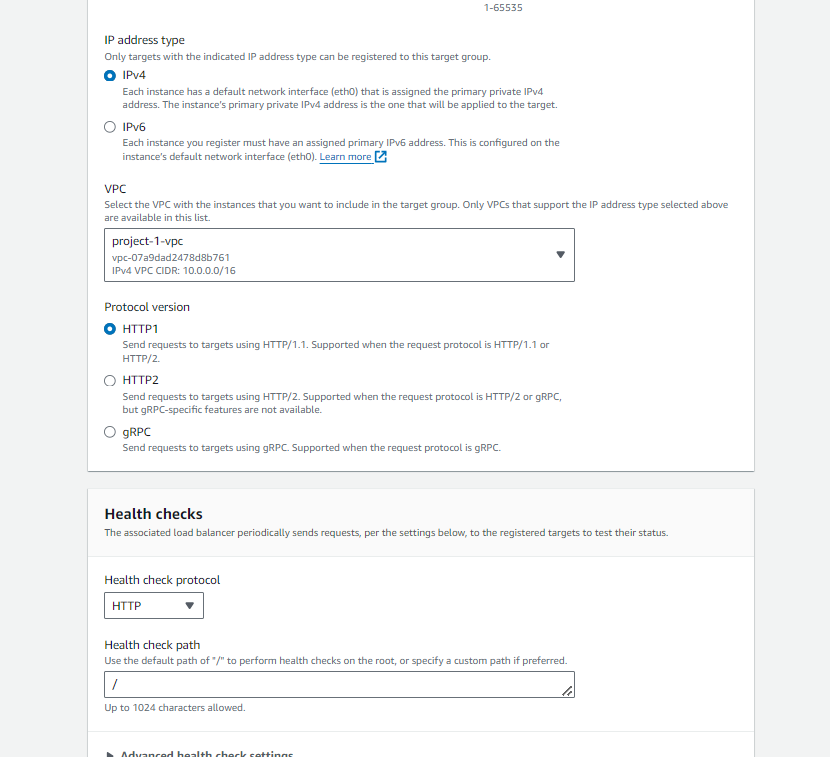
->select VPC as PROJECT-1.

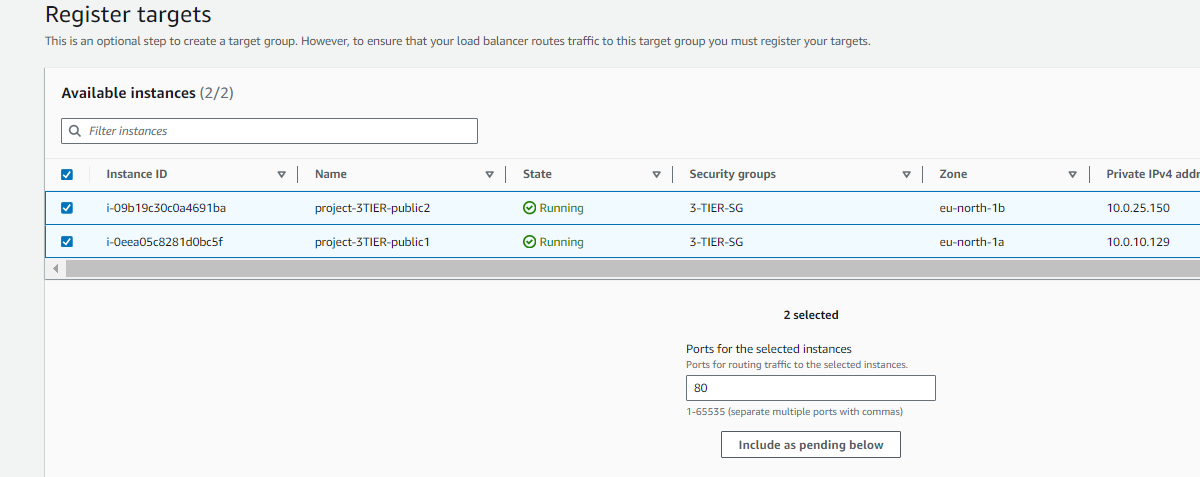
->select instances under register targets.

->click on include as pending below.

->click on create target group.







* Once target group created go to load balancer.
* Go to load balancer and click on create load balancer.
* Select application load balancer.
* Give name tag as WEB-TIER-LB

->click on internet facing under scheme field.

->select VPC PROJECT-1.

->select availability zone us-east-1a & us-east-1b.

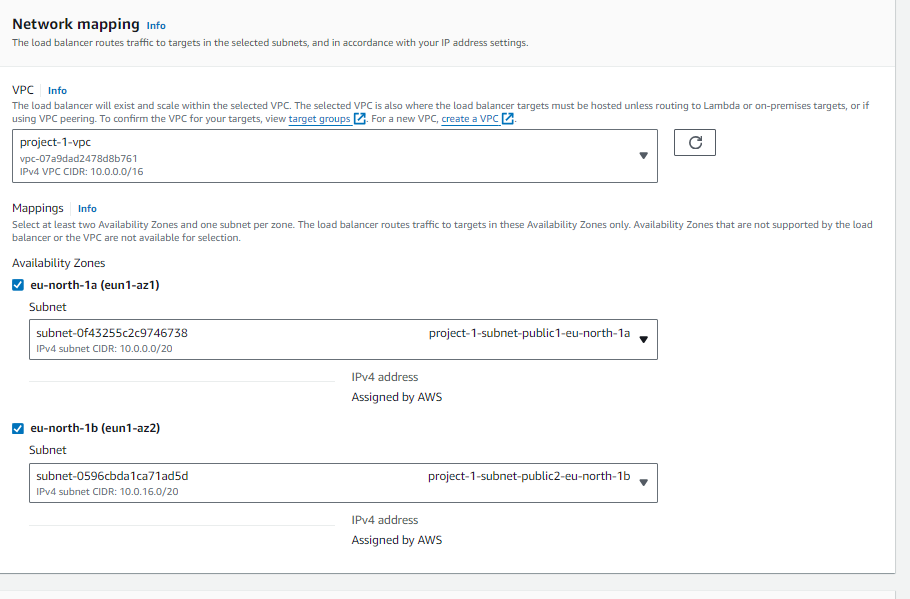
->select subnets public1 & public2 .

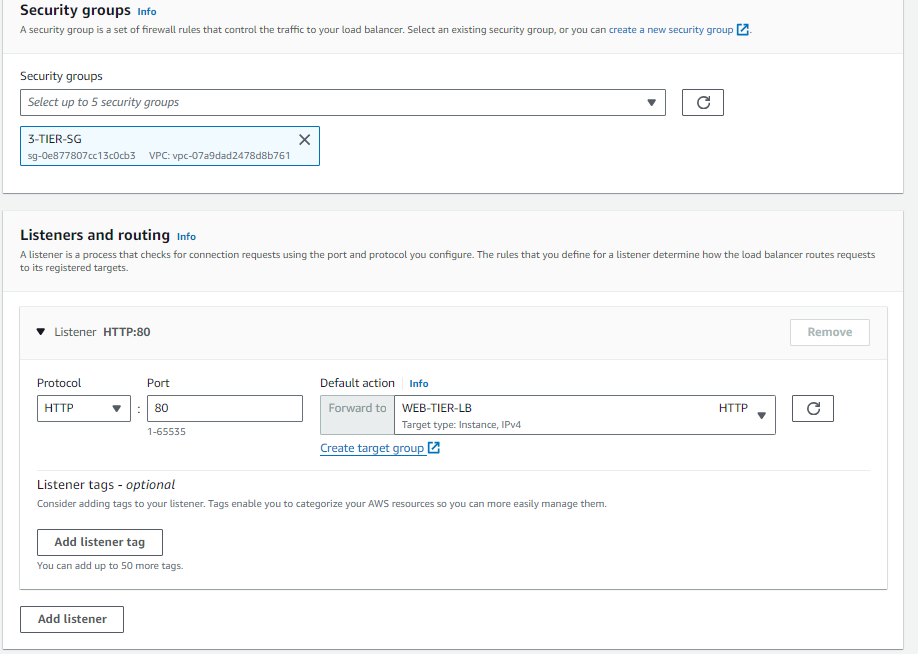
>select security group.

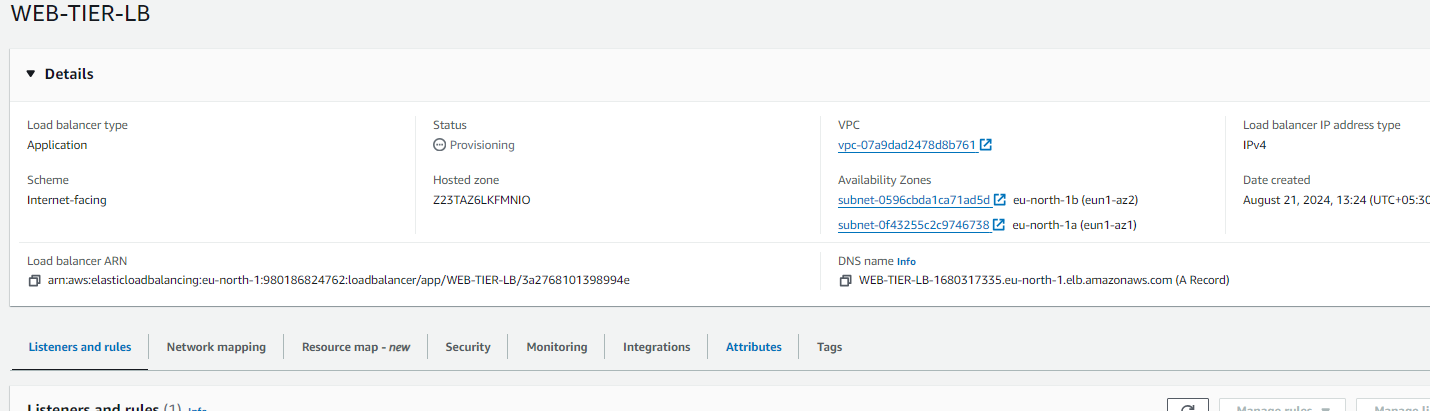
->select target group WEB-TIER-TG).

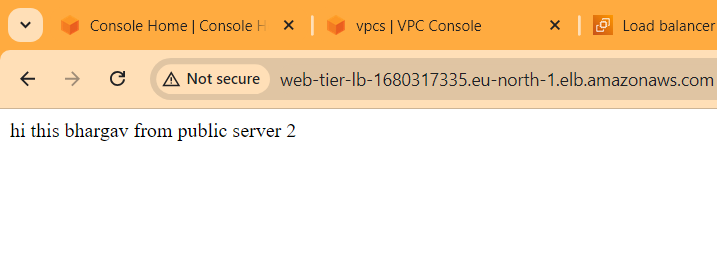
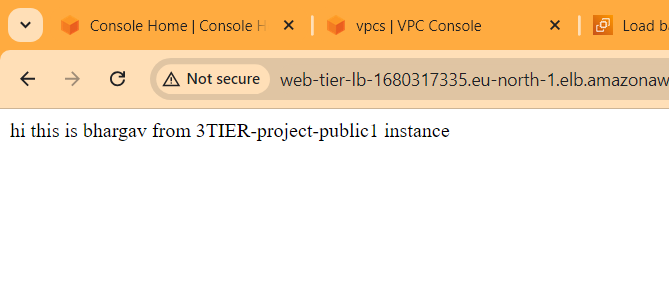
->remaining fields leave as default and click on next.

->click on create load balancer.







**Step-3 :Create AML(image) and Auto Scalling**

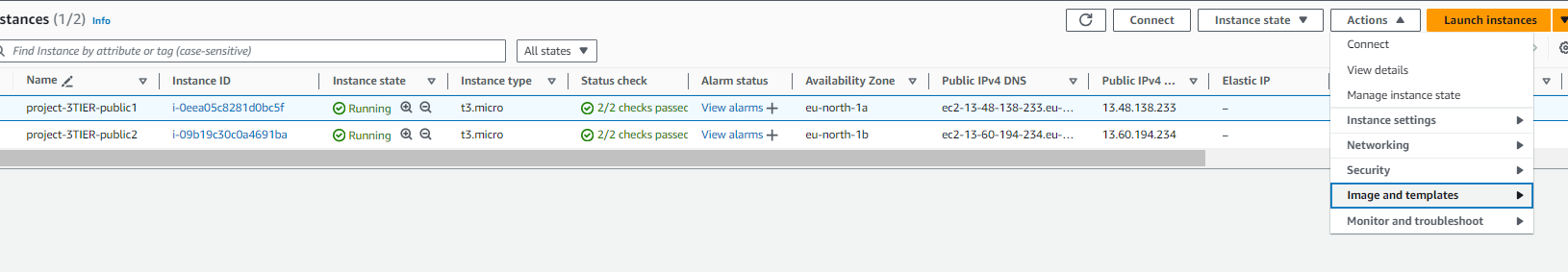
* Create image using instance

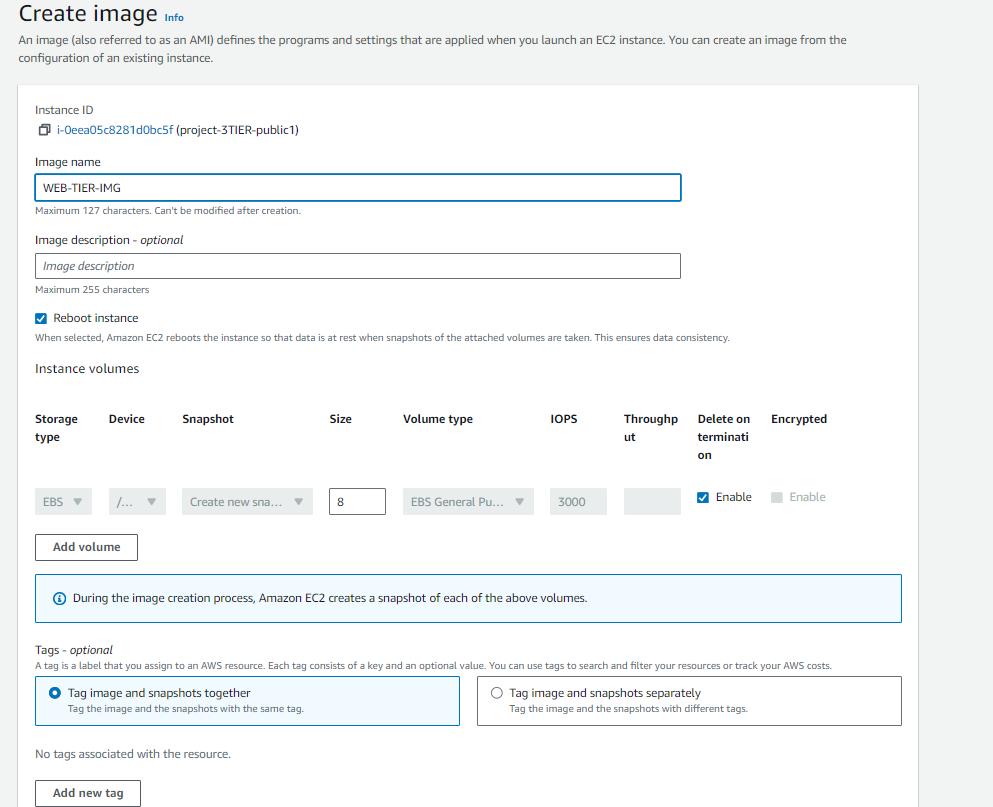
->select instance and click actions.

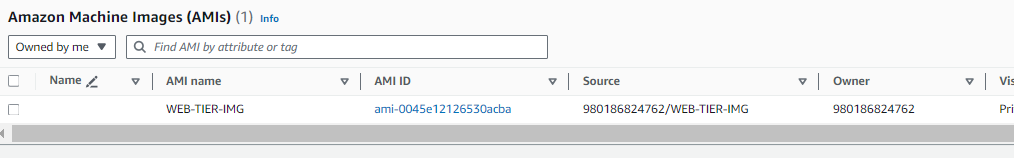
->image and templates.

->create image.

* Give name tag as WEB-TIER-IMG.







**Create Autoscaling group**

• For creating autoscaling group we need to create a launch template.

• After available of image. Click on create a launch template.

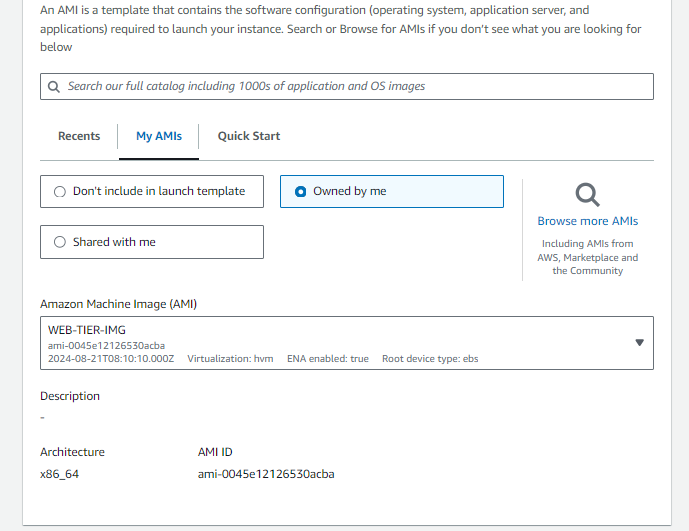
• Template name as my-public-template, description as nothing.

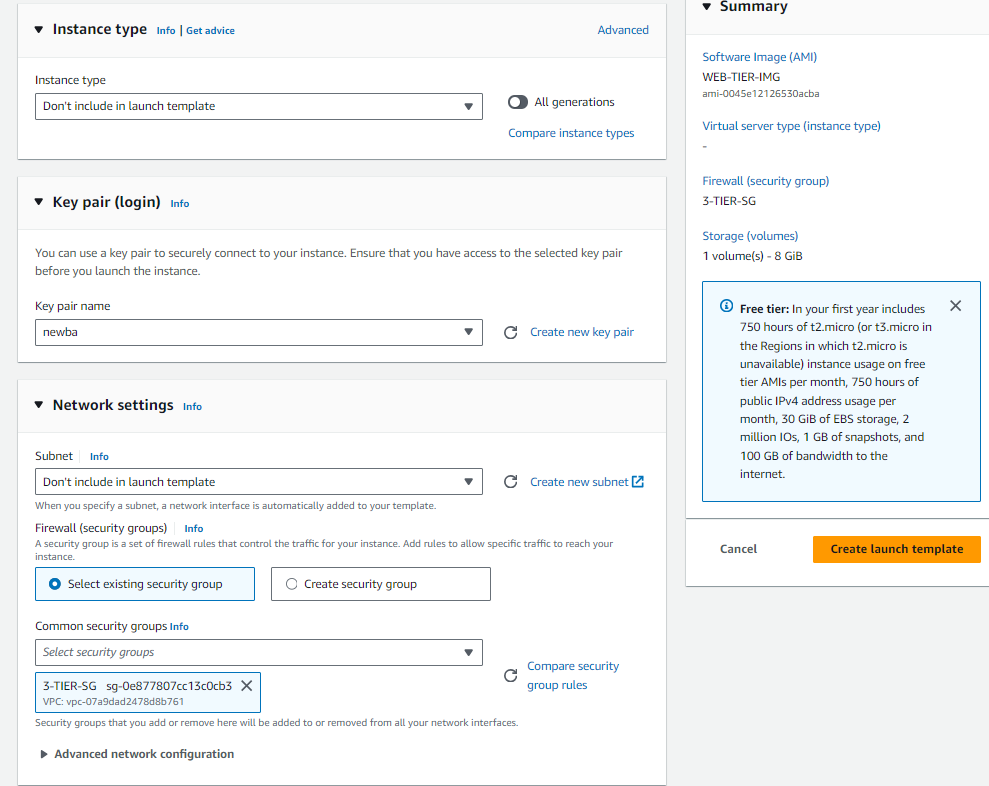
• Select AMI’s as share with me, select my-image.

• Instance type as t2. micro and key pair as project.

• Select existing security group (public) which is used to launch an EC2 instance.

• Now click on create launch template Auto scaling for the public subnets (instances public-1 & public-2



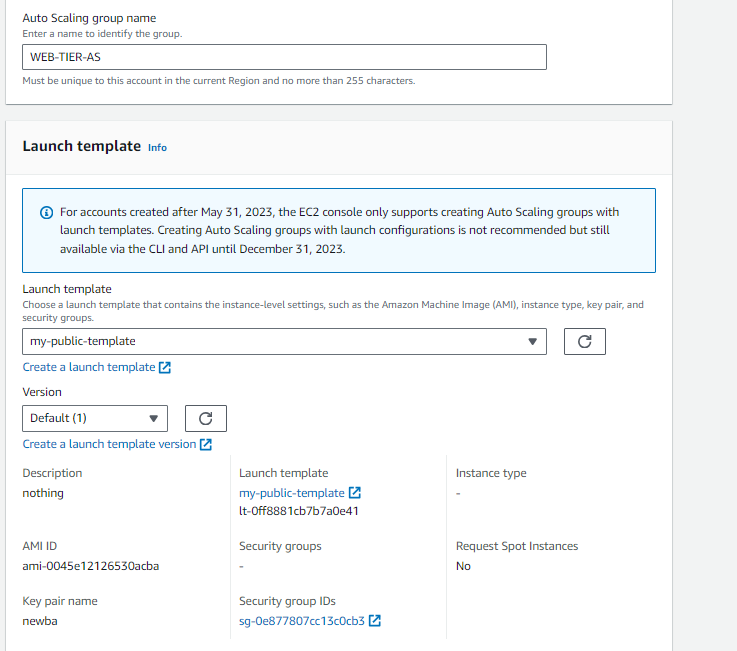


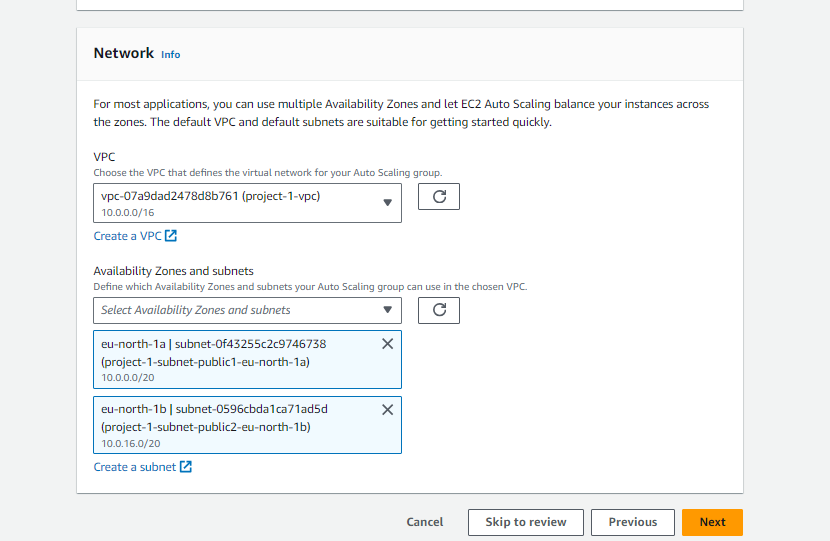
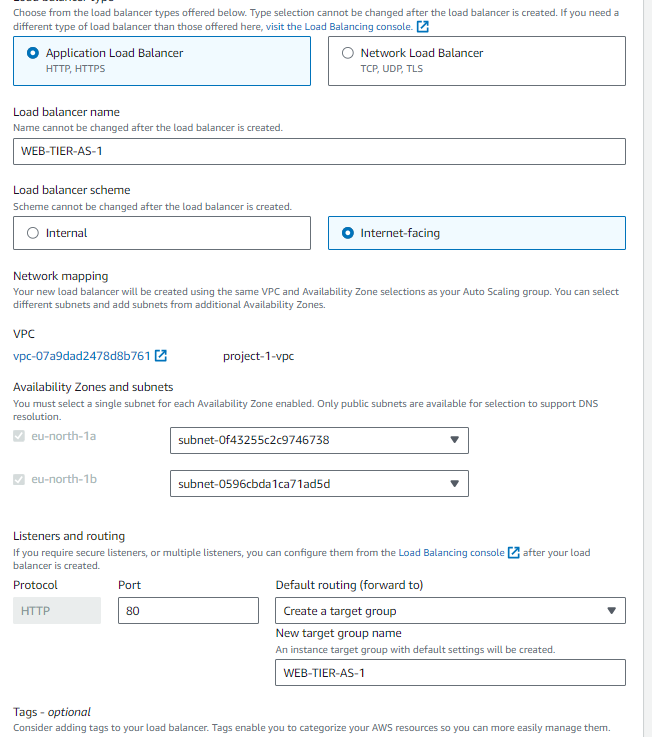
* Go to auto scaling and click on create auto scaling groups.
* Give name tag as WEB-TIER-AS
* Select created target group WEB-TIER-TG.
* Select the VPC

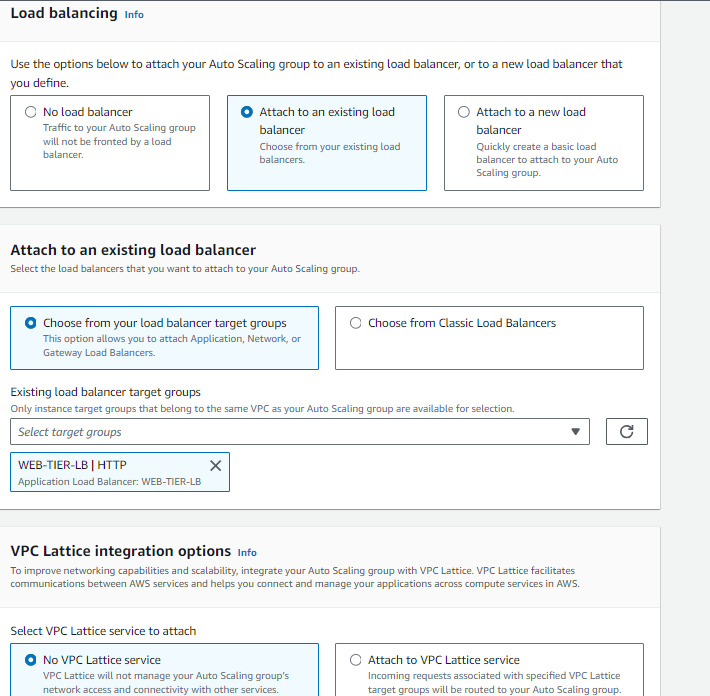
-> select availability zone eu-north-1a (public1 subnet) & eu-north-1b (public2 subnet)

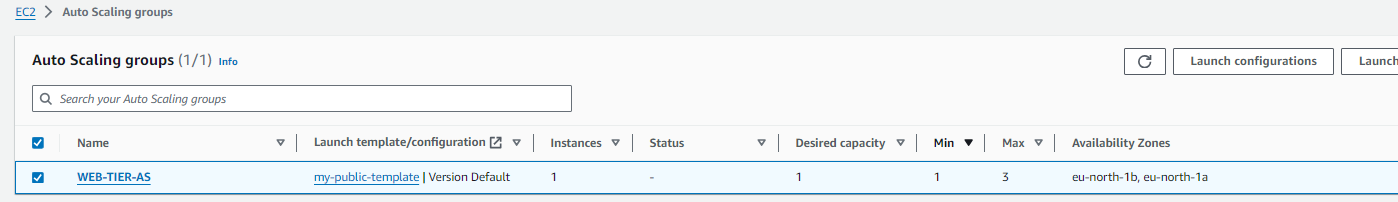
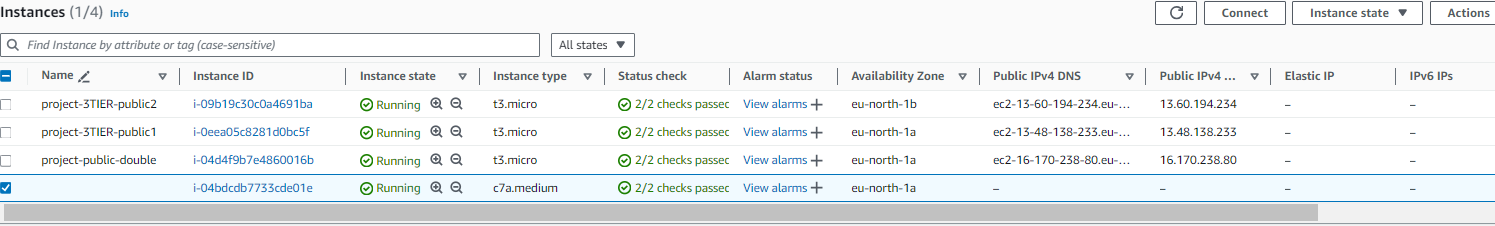
->click attached to an existing load balancer

->select existing target group.



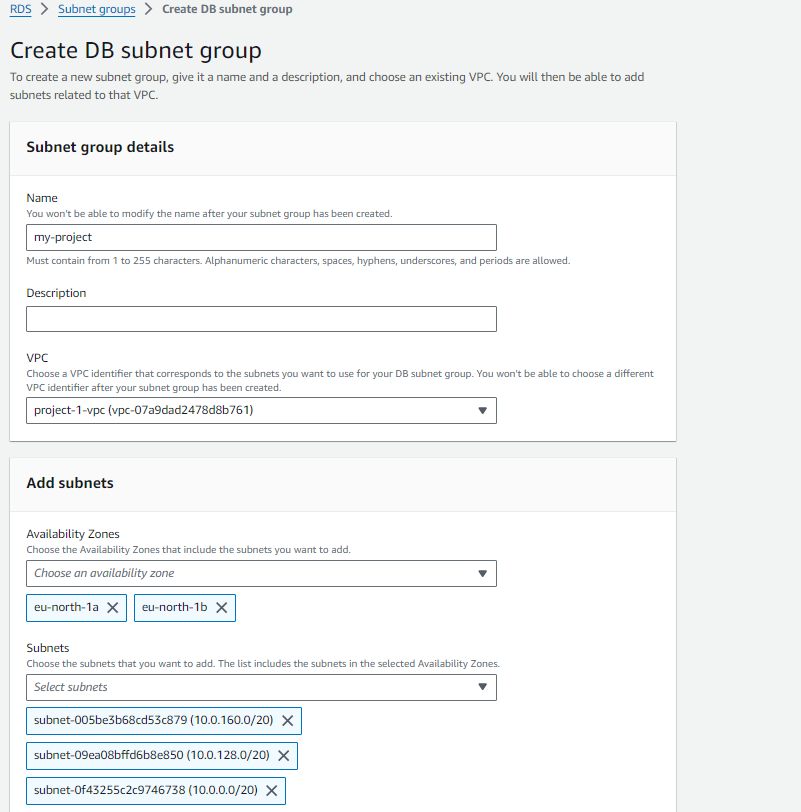
 

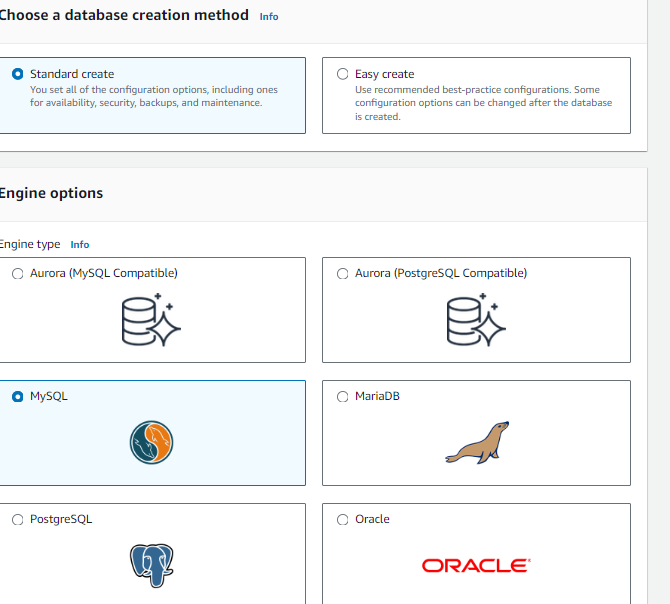


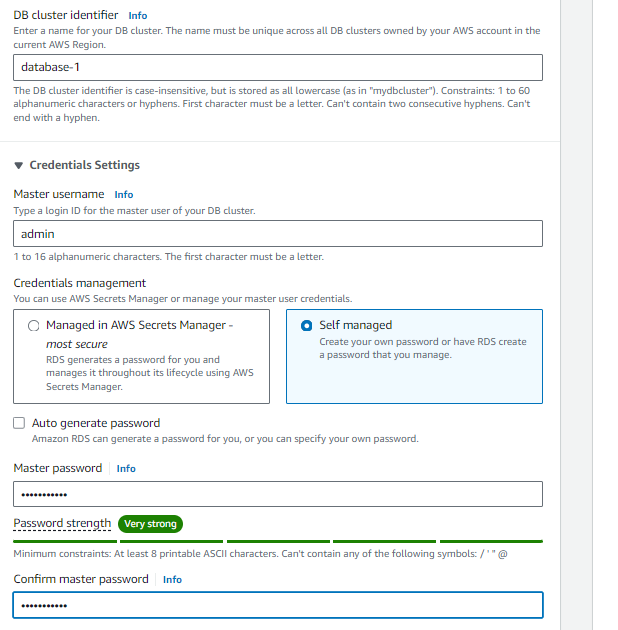
 

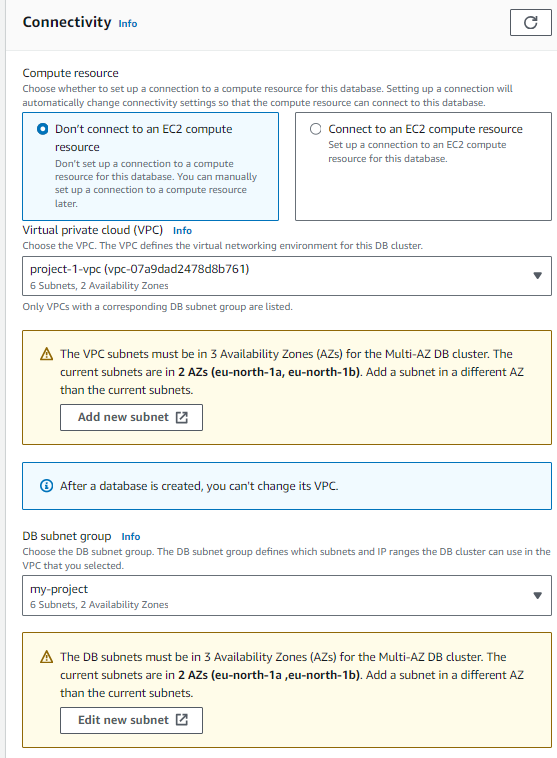
**Step-4 : Creating RDS**

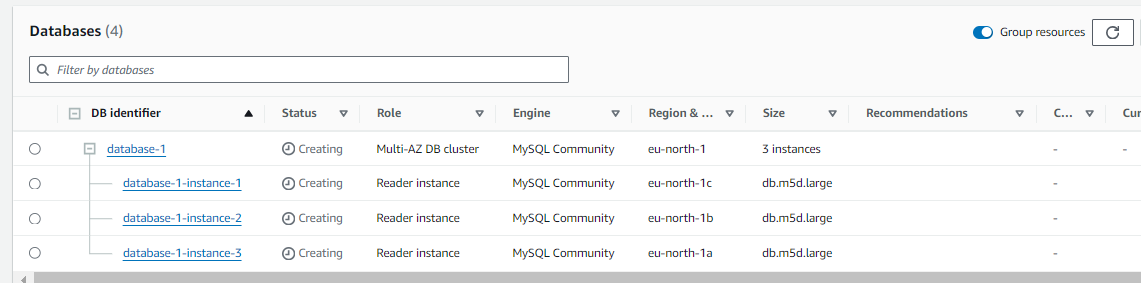
* Open RDS and search for subnet.
* Go to subnet group and click on create DB subnet group.
* Give name tag as my-3tier-project-db.
* Select VPC project-3tier and then click on create DB subnet.

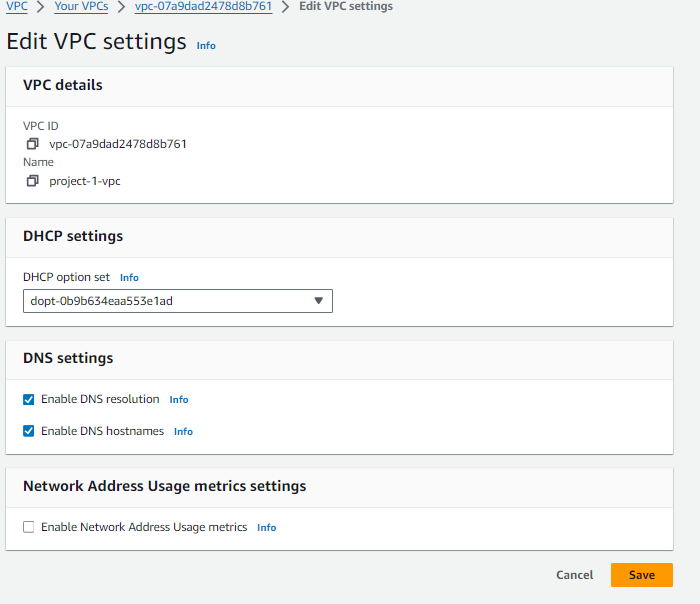
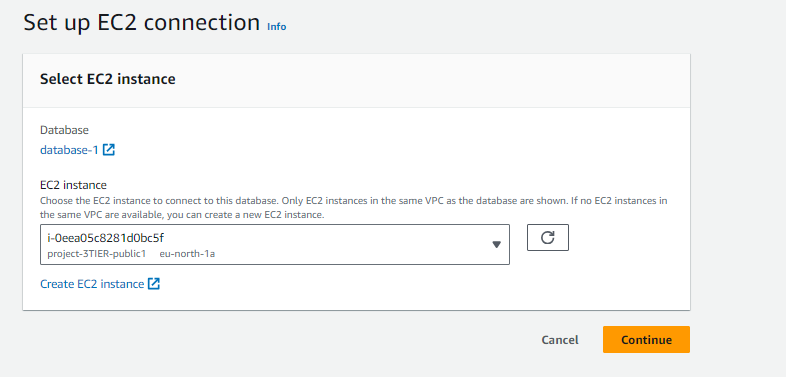


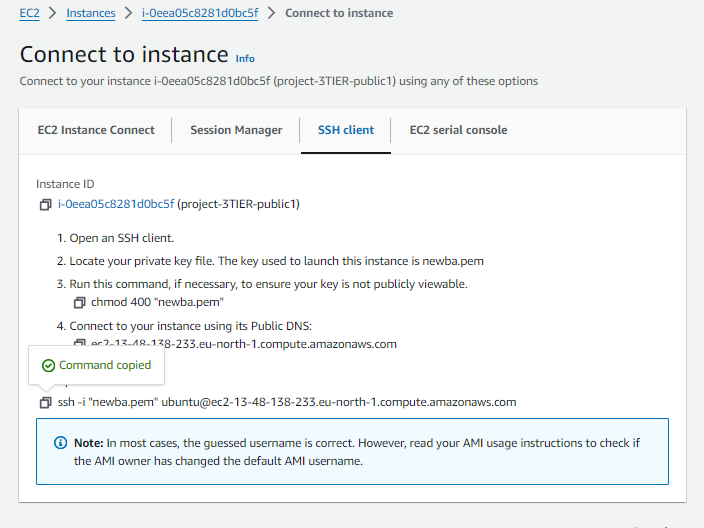


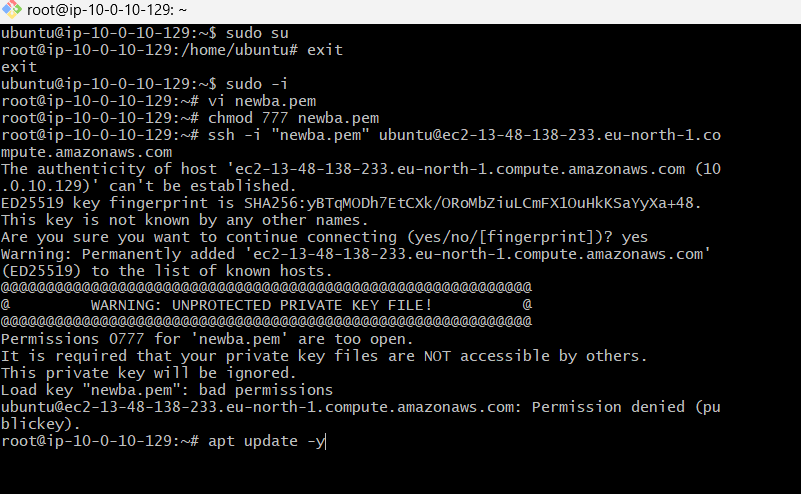




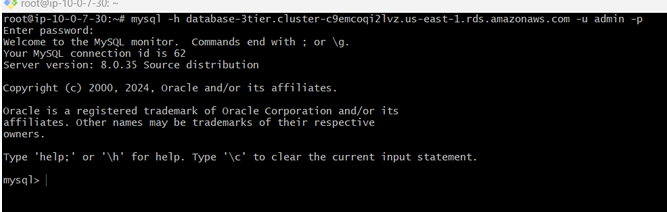
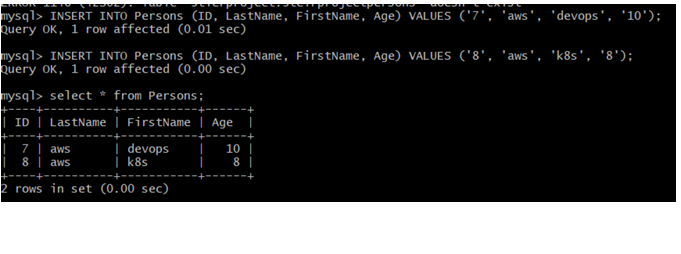








* Install MY SQL->sudo apt install mysql-server.
* Restart ->sudo systemctl start mysql service.
* Mysql -h database-1.cluster-cn2ouu0mwz22.eu-north-1.rds.amazonaws.com -u admin -p->Enter password-> u can connect to my SQL as shown in below.
* For creating table-> CREATE TABLE Persons ( ID int, LastName varchar(255), FirstName varchar(255), Age varchar(255) ).
* Create the table in in my sql.
* The output of creating table as shown in below.

**B.Bhargav**

[**mailto:bargavrambalineni@gmail.com**](mailto:bargavrambalineni@gmail.com)