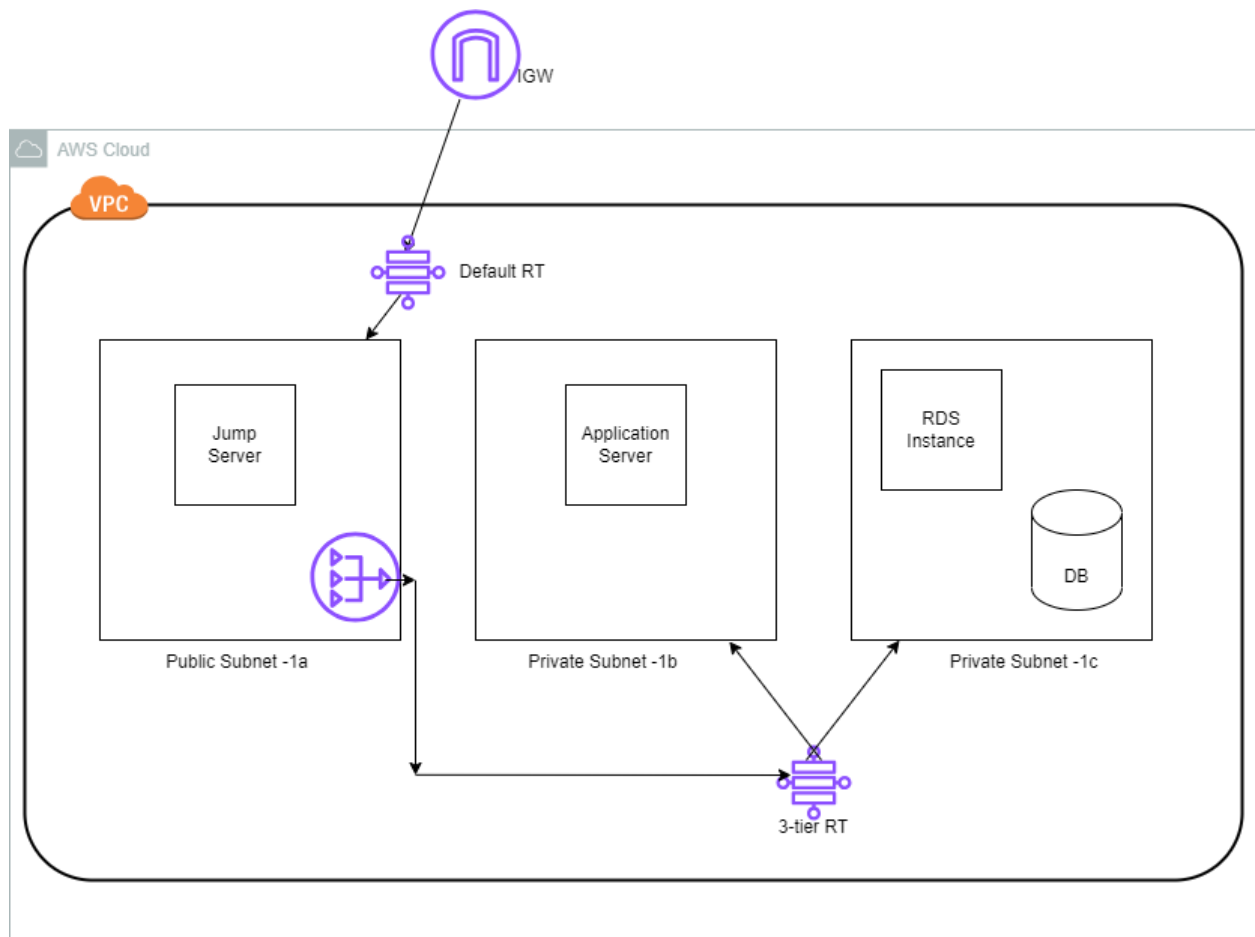


Building a 3-tier web application architecture with AWS

The three-tier architecture is the most popular implementation of a multi-tier architecture and consists of a single Jump server, presentation tier, and data tier. The following illustration shows an example of a simple, generic three-tier application.

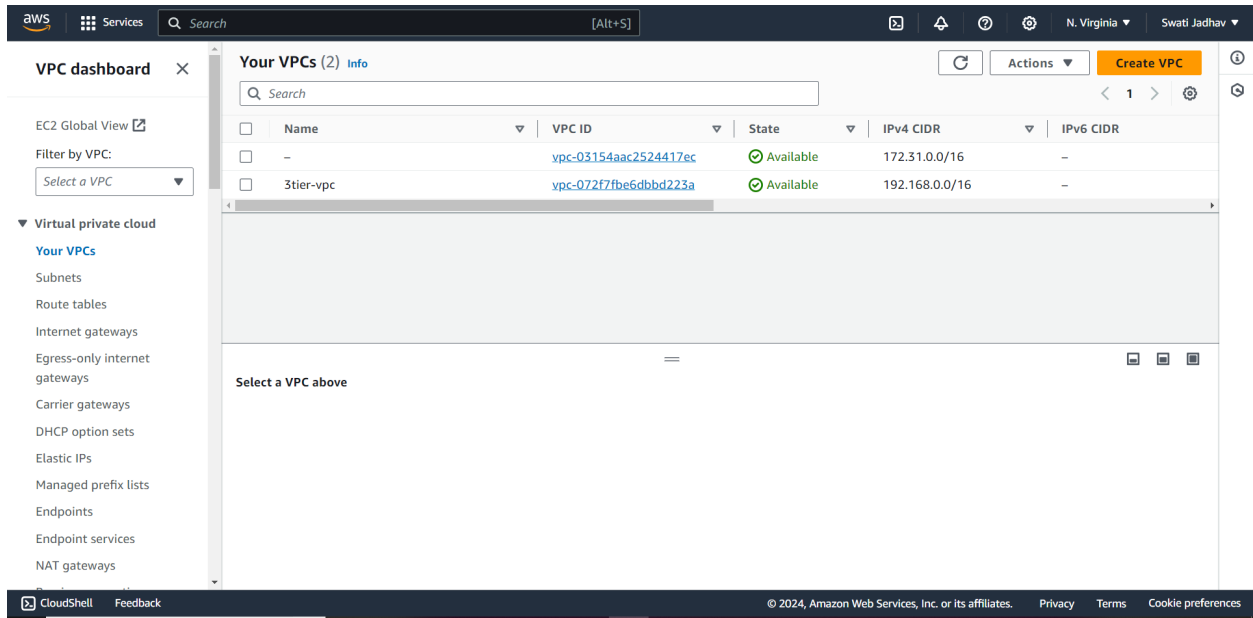
Prerequisites

- Familiarity with the AWS Management Console.
- Familiarity with VPC network structures, EC2 instances, and security groups.
- Familiarity with Linux commands, scripting, and SSH.
- Access to a command line tool.

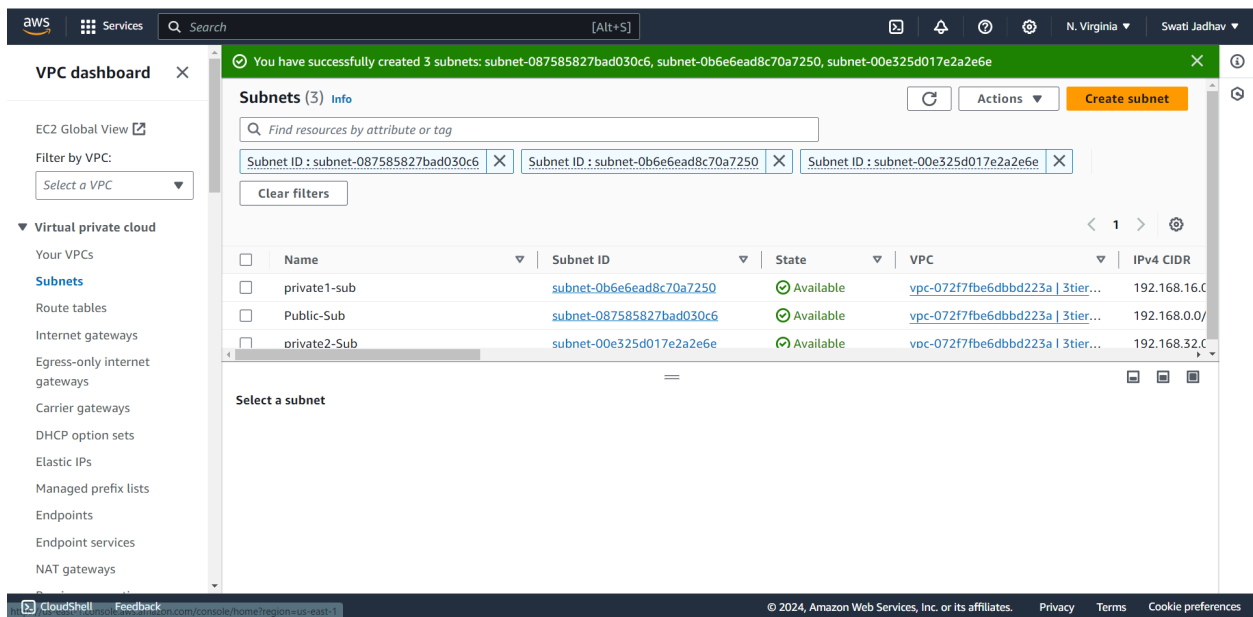


Steps:

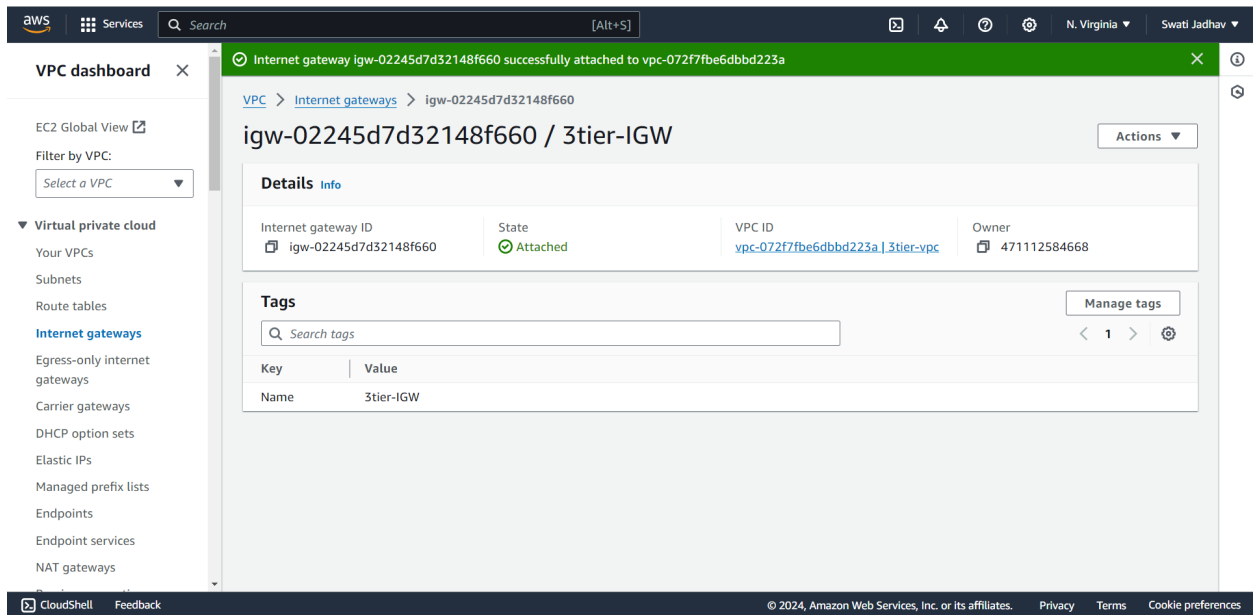
1. Logging to AWS using your login credentials
2. Create a VPC for any reason(N.Virginia) allocate an appropriate CIDR range and give the VPC name as(3 tier-VPC).



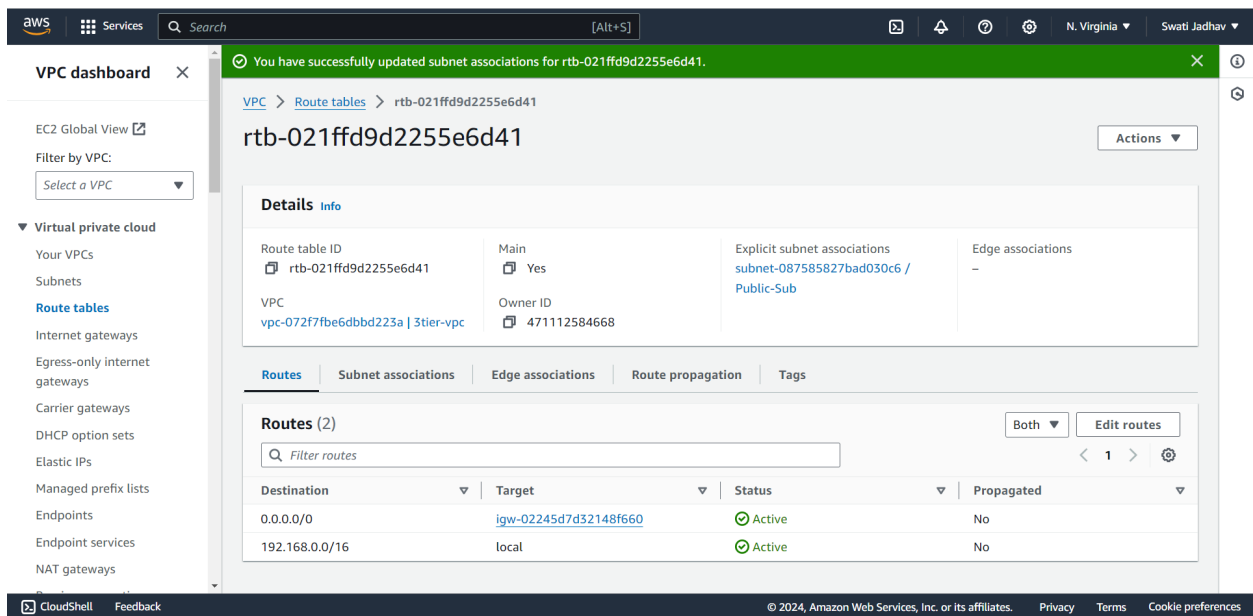
3. Create 3 Subnets into the VPC, 1 Public-Subnet as Public-sub & 2 Private Subnet as private1-sub, private2-sub



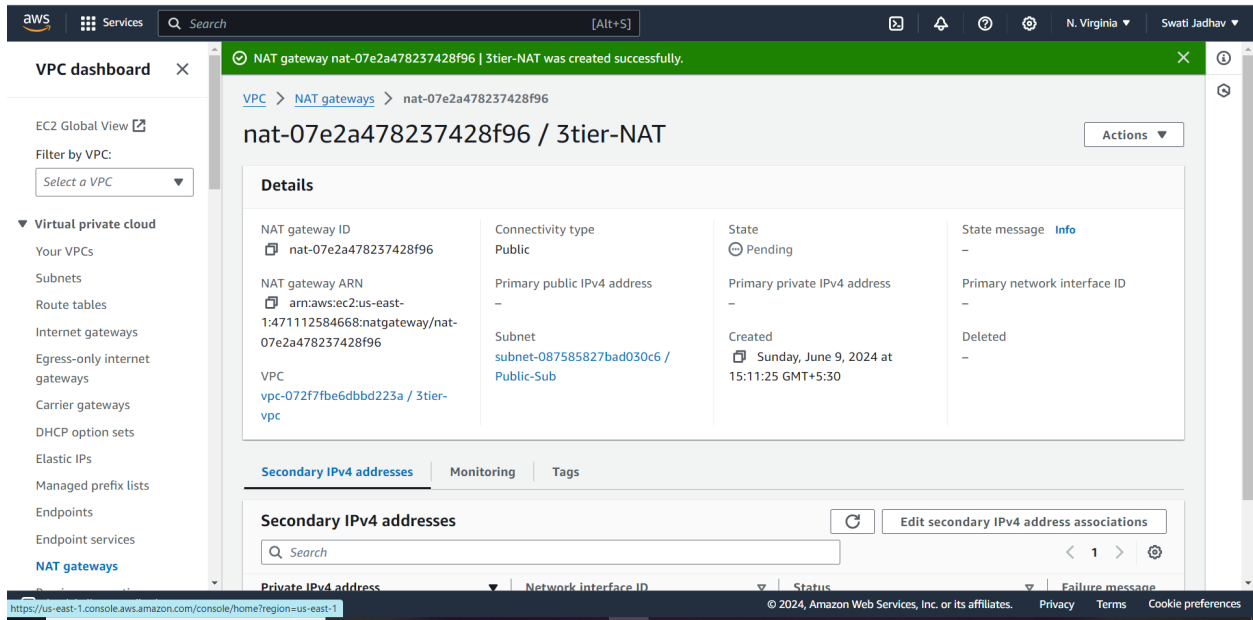
4. Create an Internet Gateway to provide Internet access to the VPC.



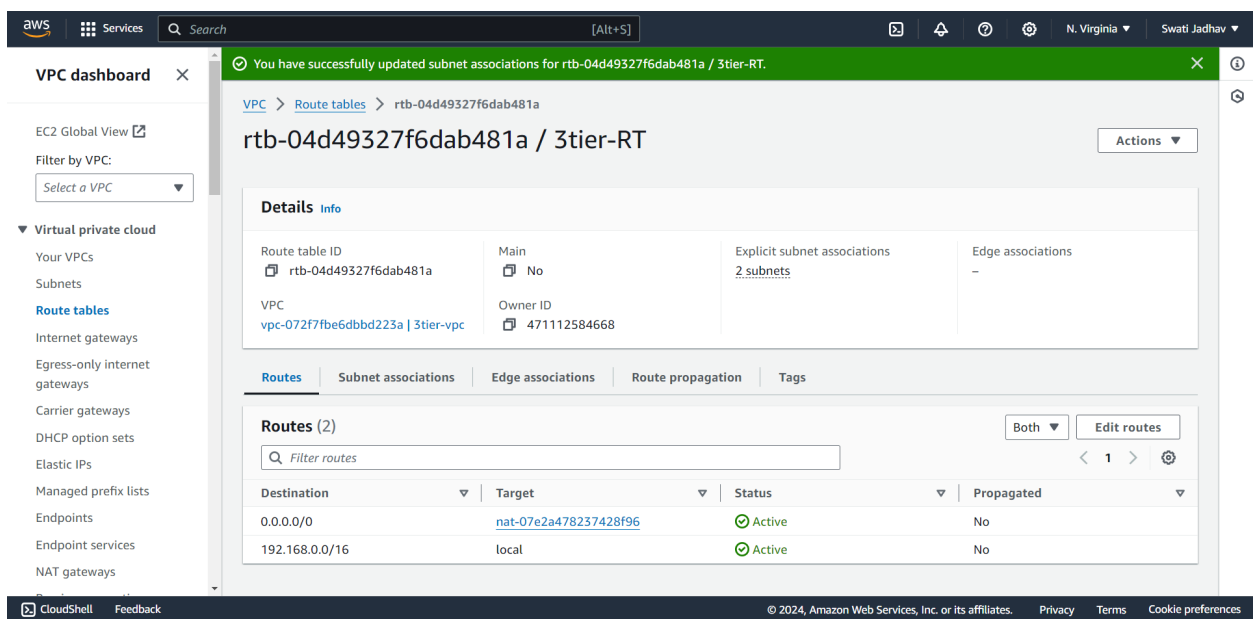
5. Route the Internet gateway to the default (main)route table.



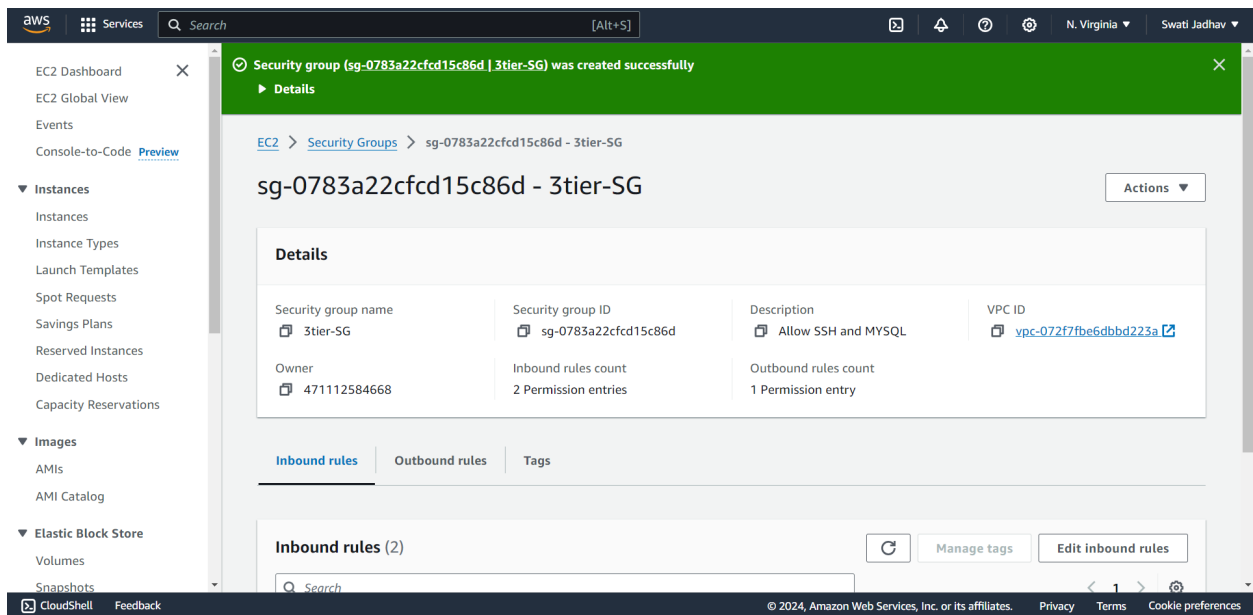
6. Create a NAT gateway(Network Access Translator) and associate it to both the Private subnet(private1-sub, private2-sub)



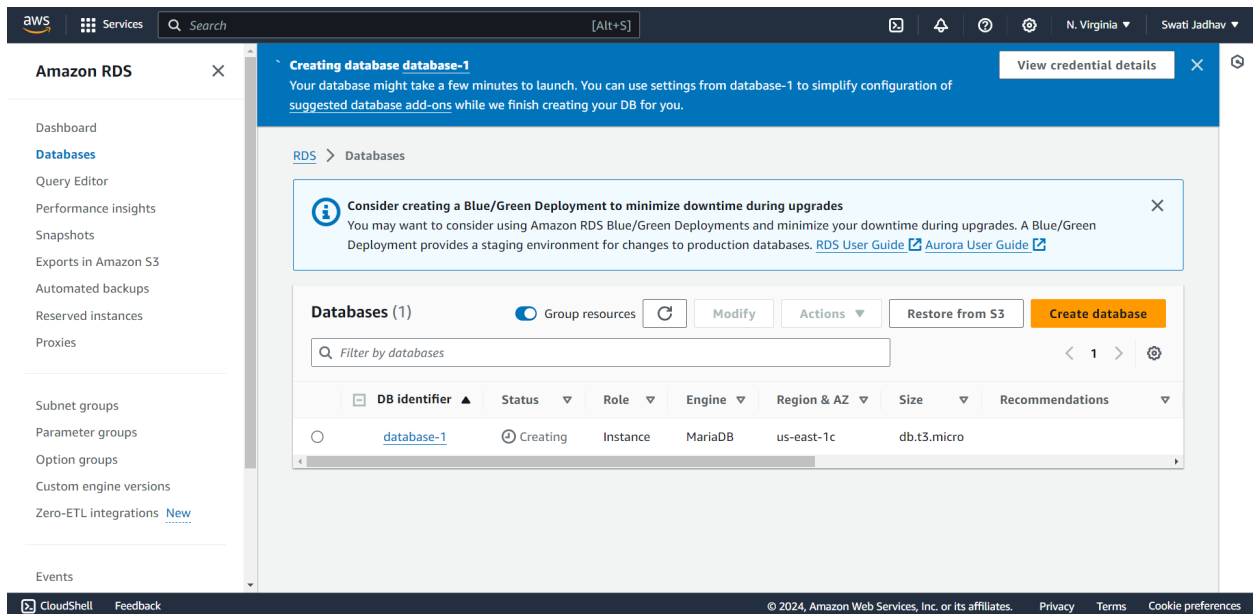
7. Create a Route table (3tier-RT) and route the NAT gateway



8. Create a Security Group with an inbound rule configured in it to allow SSH (22) and MYSQL(3306).



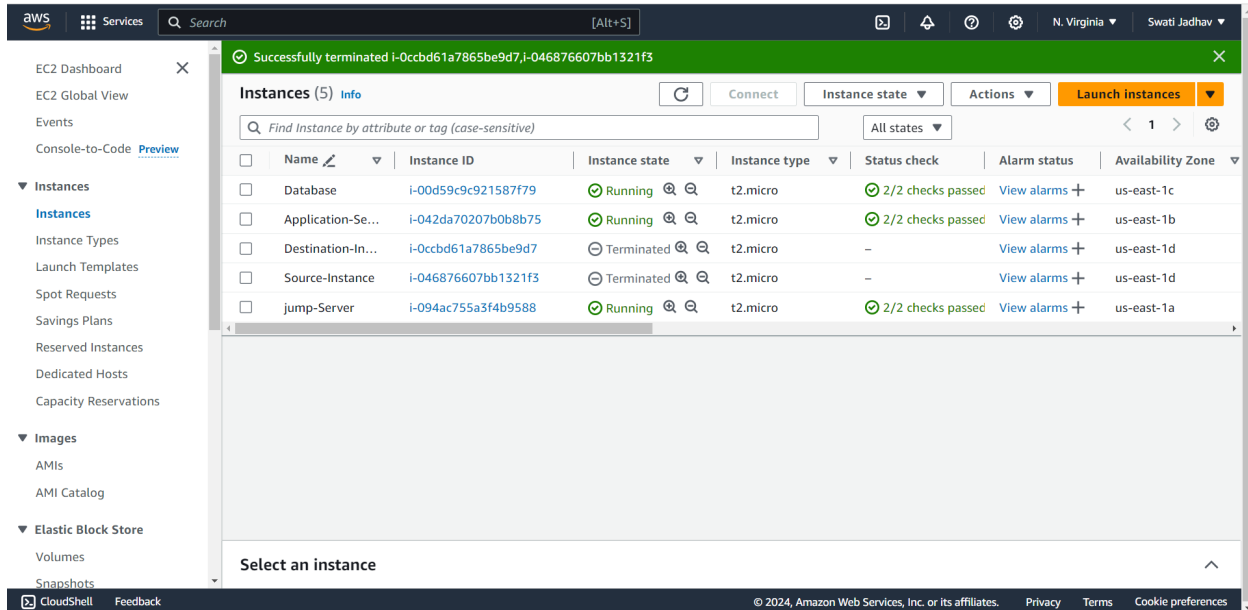
9. Create an RDS by attaching the Security group to it.



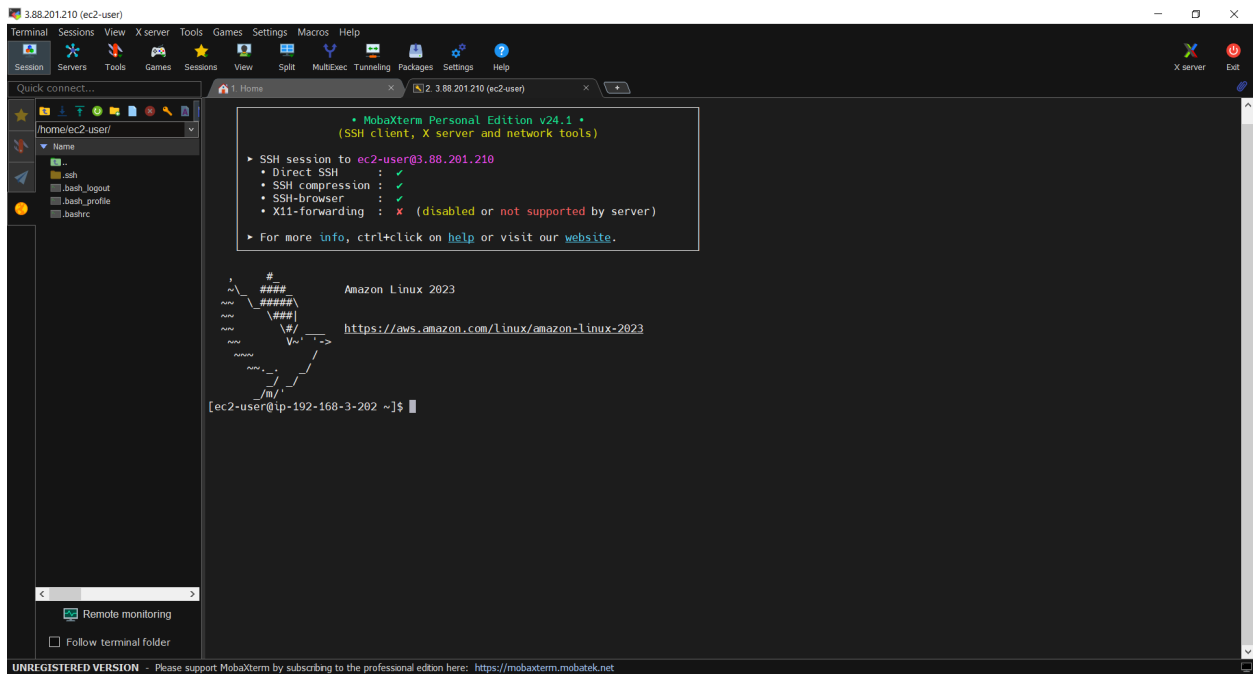
10. Create 3 Instance

- 1. First instance in the public subnet (jump Server) with permission HTTP(80) and SSH(22)

- 2.Second instance is in the private1-subnet (Application-Server)with permission Tomcat(8080) and SSH(22).
- 3.The third instance is in the private2-subnet (Database)with permission MySQL(3306) and SSH(22).



11. Connect to the Jump-server using SSH.



12. Install Nginx on that server

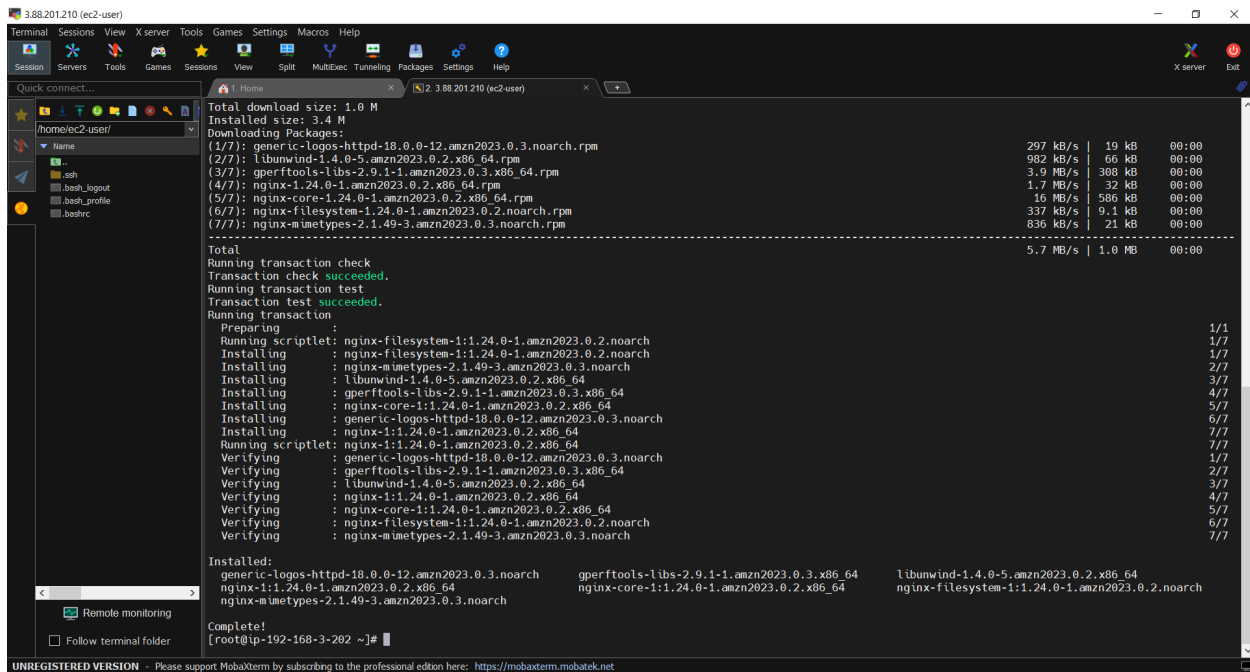
```
sudo -i
yum install nginx -y

vim /etc/nginx/nginx.conf

location /{

proxy_pass http://privateIPofApplicationserver:8080/student/;

}
```

13. Start the Nginx service and copy the key to the jump-server

```
systemctl restart nginx.service
```

To copy the key to the jump server command is

```
scp -i key-name key-name ec2-user@publicIP:/home/ec2-user
```

change the key permission to

```
chmod 400 key-name
```

14. SSH to Application-server


```

18.232.169.67 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Quick connect...
User sessions
PuTTY sessions
18.232.169.67 (ec2-user)
3.88.201.210 (ec2-user)
3.88.201.210 (ec2-user) (1)
3.89.253.2 (ec2-user)

apache-tomcat-8.5.100/webapps/host-manager/images/asf-logo.svg
apache-tomcat-8.5.100/webapps/host-manager/images/tomcat.svg
apache-tomcat-8.5.100/webapps/host-manager/index.jsp
apache-tomcat-8.5.100/webapps/manager/META-INF/context.xml
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/401.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/403.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/404.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/connectorCerts.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/connectorCiphers.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/connectorTrustedCerts.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/sessionDetail.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/jsp/sessionList.jsp
apache-tomcat-8.5.100/webapps/manager/WEB-INF/web.xml
apache-tomcat-8.5.100/webapps/manager/css/manager.css
apache-tomcat-8.5.100/webapps/manager/images/asf-logo.svg
apache-tomcat-8.5.100/webapps/manager/images/tomcat.svg
apache-tomcat-8.5.100/webapps/manager/index.jsp
apache-tomcat-8.5.100/webapps/manager/status.xsd
apache-tomcat-8.5.100/webapps/manager/xform.xsl
apache-tomcat-8.5.100/bin/catalina.sh
apache-tomcat-8.5.100/bin/ciphers.sh
apache-tomcat-8.5.100/bin/configtest.sh
apache-tomcat-8.5.100/bin/daemon.sh
apache-tomcat-8.5.100/bin/digest.sh
apache-tomcat-8.5.100/bin/setclasspath.sh
apache-tomcat-8.5.100/bin/shutdown.sh
apache-tomcat-8.5.100/bin/startup.sh
apache-tomcat-8.5.100/bin/tool-wrapper.sh
apache-tomcat-8.5.100/bin/version.sh
[root@ip-192-168-16-70 ~]# cd /opt
[root@ip-192-168-16-70 opt]# ls
apache-tomcat-8.5.100
[root@ip-192-168-16-70 opt]# cd apache-tomcat-8.5.100/
[root@ip-192-168-16-70 apache-tomcat-8.5.100]# ls
BUILDING.txt CONTRIBUTING.md LICENSE NOTICE README.md RELEASE-NOTES RUNNING.txt bin conf lib logs temp webapps work
[root@ip-192-168-16-70 apache-tomcat-8.5.100]# cd webapps/
[root@ip-192-168-16-70 webapps]# curl -O https://s3-us-west-2.amazonaws.com/studentapi-cit/student.war
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 89423 100 89423 0 0 228k 0 --:--:-- --:--:-- --:--:-- 229k
[root@ip-192-168-16-70 webapps]# cd ../lib
[root@ip-192-168-16-70 lib]#

```

16. Install a connector to connect the application and the database.

- ```
curl -O https://s3-us-west-2.amazonaws.com/studentapi-cit/mysqlconnector.jar
```
- `Ls(mysql-connector.jar)`
  - `Cd ../conf/`

```

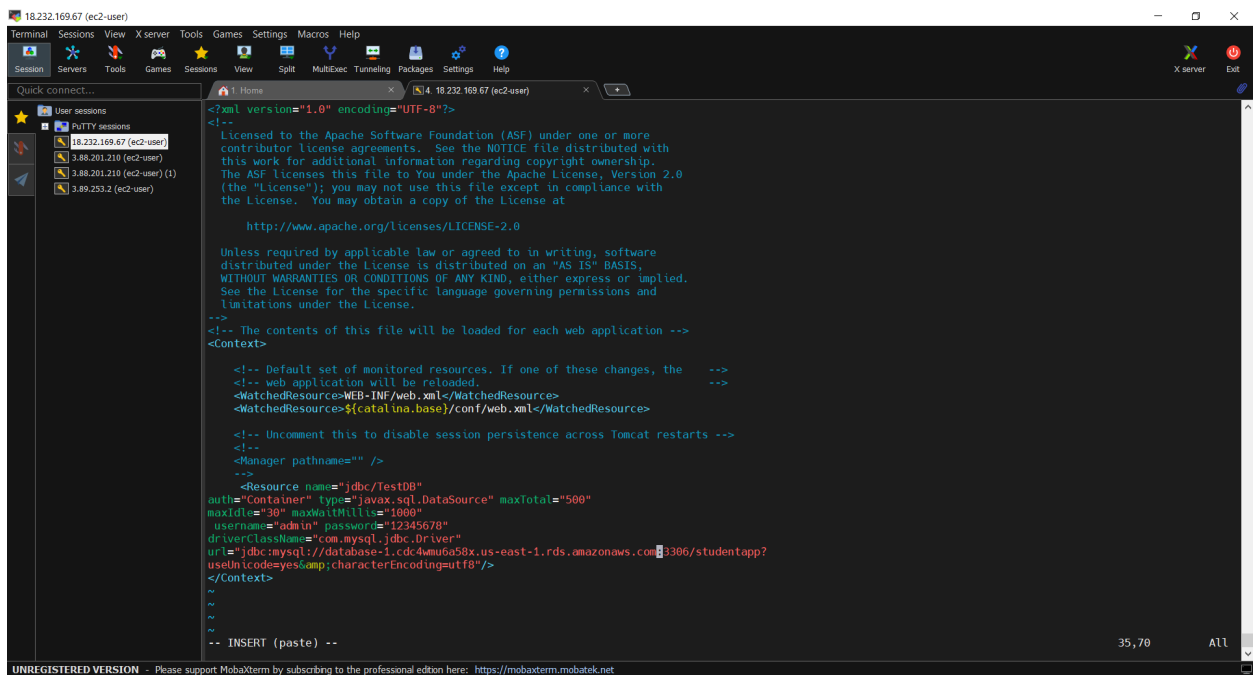
100 303 0 303 0 0 1440 0 --:--:-- --:--:-- --:--:-- 1442
[ec2-user@ip-192-168-16-61 webapps]$ cd ../lib
[ec2-user@ip-192-168-16-61 lib]$ curl -O https://s3-us-west-2.amazonaws.com/studentapi-cit/mysqlconnector.jar
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
100 289 0 289 0 0 1292 0 --:--:-- --:--:-- --:--:-- 1295
[ec2-user@ip-192-168-16-61 lib]$ ls
annotations-api.jar catalina.jar jaspic-api.jar tomcat-coyote.jar tomcat-i18n-ja.jar tomcat-jni.jar
catalina-ant.jar ecj-4.6.3.jar jsp-api.jar tomcat-dbcp.jar tomcat-i18n-ko.jar tomcat-util-scan.jar
catalina-ha.jar el-api.jar mysqlconnector.jar tomcat-i18n-de.jar tomcat-i18n-ru.jar tomcat-util.jar
catalina-storeconfig.jar jasper-el.jar servlet-api.jar tomcat-i18n-es.jar tomcat-i18n-zh-CN.jar tomcat-websocket.jar
catalina-tribes.jar jasper.jar tomcat-api.jar tomcat-i18n-fr.jar tomcat-jdbc.jar websocket-api.jar
[ec2-user@ip-192-168-16-61 lib]$ cd ../conf/
[ec2-user@ip-192-168-16-61 conf]$ vim context.xml
[ec2-user@ip-192-168-16-61 conf]$ vim context.xml
[ec2-user@ip-192-168-16-61 conf]$ cd ../bin
[ec2-user@ip-192-168-16-61 bin]$./catalina.sh start
Using CATALINA_BASE: /home/ec2-user/apache-tomcat-8.5.100
Using CATALINA_HOME: /home/ec2-user/apache-tomcat-8.5.100
Using CATALINA_TMPDIR: /home/ec2-user/apache-tomcat-8.5.100/temp
Using JRE_HOME: /usr
Using CLASSPATH: /home/ec2-user/apache-tomcat-8.5.100/bin/bootstrap.jar:/home/ec2-user/apache-tomcat-8.5.100/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[ec2-user@ip-192-168-16-61 bin]$

```

17.

Vim context.xml

- (in context tab [last of page]) <Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource" maxTotal="500" maxIdle="30" maxWaitMillis="1000" username="admin" password="12345678" driverClassName="com.mysql.jdbc.Driver" url="jdbc:mysql://endpoint:3306/studentapp?useUnicode=yes&characterEncoding=utf8"/>



## 18. Start the Tomcat server.

```
cd ../bin
• ./catalina.sh start
• exit
```

```

catalina.properties jaspic-providers.xml logging.properties tomcat-users.xml web.xml
[root@ip-192-168-16-70 conf]# vim context.xml
[root@ip-192-168-16-70 bin]# cd ../bin/
[root@ip-192-168-16-70 bin]# ./catalina.sh start
Using CATALINA_BASE: /opt/apache-tomcat-8.5.100
Using CATALINA_HOME: /opt/apache-tomcat-8.5.100
Using CATALINA_TMPDIR: /opt/apache-tomcat-8.5.100/temp
Using JRE_HOME: /usr
Using CLASSPATH: /opt/apache-tomcat-8.5.100/bin/bootstrap.jar:/opt/apache-tomcat-8.5.100/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@ip-192-168-16-70 bin]#

```

## 19. SSH to the Database instance

- install mariadb

```

18.232.169.67 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
User sessions
PuTTY sessions
18.232.169.67 (ec2-user)
3.88.201.210 (ec2-user)
3.88.201.210 (ec2-user) (1)
3.89.253.2 (ec2-user)

Transaction Summary
=====
Install 5 Packages
Total download size: 1.8 M
Installed size: 19 M
Downloading Packages:
(1/5): mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch.rpm 144 kB/s | 9.2 kB 00:00
(2/5): mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64.rpm 2.6 MB/s | 196 kB 00:00
(3/5): mariadb105-common-10.5.23-1.amzn2023.0.1.x86_64.rpm 1.3 MB/s | 30 kB 00:00
(4/5): perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64.rpm 883 kB/s | 10 kB 00:00
(5/5): mariadb105-10.5.23-1.amzn2023.0.1.x86_64.rpm 14 MB/s | 1.6 MB 00:00

Total 9.2 MB/s | 1.8 MB 00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
 Preparing : mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch 1/1
 Installing : mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 1/5
 Installing : mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64 2/5
 Installing : perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64 3/5
 Installing : mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 4/5
 Installing : mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 5/5
 Running scriptlet : mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 1/5
 Verifying : mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 2/5
 Verifying : mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch 3/5
 Verifying : mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 4/5
 Verifying : mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64 5/5
 Verifying : perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64 5/5

Installed:
mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64
mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64 perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64

Complete!
[root@ip-192-168-32-42 ~]#

```

## 20. Connect to RDS

- create a table.

Ssh -i key-name ec2-user@dbinstanceip (connected to db. instance)

- Yum install mariadb105 -y
- Mysql -h rdsendpoint -u admin -p(connected to RDS)
- Create database;
- Use database;
- CREATE TABLE if not exists students(student\_id INT NOT NULL

```

AUTO_INCREMENT,
student_name VARCHAR(100) NOT NULL,
student_addr VARCHAR(100) NOT NULL,
student_age VARCHAR(3) NOT NULL,
student_qual VARCHAR(20) NOT NULL,
student_percent VARCHAR(10) NOT NULL,
student_year_passed VARCHAR(10) NOT NULL,
PRIMARY KEY (student_id)
);

```

```

18.232.169.67 (ec2-user)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect...
User sessions
 18.232.169.67 (ec2-user)
 3.88.201.210 (ec2-user)
 3.88.201.210 (ec2-user) (1)
 3.89.253.2 (ec2-user)
Putty sessions
Installing : mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 5/5
Running scriptlet: mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 5/5
Verifying : mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 1/5
Verifying : mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch 2/5
Verifying : mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64 3/5
Verifying : mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64 4/5
Verifying : perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64 5/5

Installed:
mariadb-connector-c-3.1.13-1.amzn2023.0.3.x86_64 mariadb-connector-c-config-3.1.13-1.amzn2023.0.3.noarch mariadb105-3:10.5.23-1.amzn2023.0.1.x86_64
mariadb105-common-3:10.5.23-1.amzn2023.0.1.x86_64 perl-Sys-Hostname-1.23-477.amzn2023.0.6.x86_64

Complete!
[root@ip-192-168-32-42 ~]# mysql -h database-1.cdc4amu6a58x.us-east-1.rds.amazonaws.com -u admin -p12345678
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 339
Server version: 10.11.6-MariaDB-log managed by https://aws.amazon.com/rds/

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

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

MariaDB [(none)]> Create database studentapp;
Query OK, 1 row affected (0.003 sec)

MariaDB [(none)]> Use database;
ERROR 1049 (42000): Unknown database 'database'
MariaDB [(none)]> Use studentapp;
Database changed
MariaDB [studentapp]> CREATE TABLE IF NOT EXISTS students(student_id INT NOT NULL
-> AUTO_INCREMENT,
-> student_name VARCHAR(100) NOT NULL,
-> student_addr VARCHAR(100) NOT NULL,
-> student_age VARCHAR(3) NOT NULL,
-> student_qual VARCHAR(20) NOT NULL,
-> student_percent VARCHAR(10) NOT NULL,
-> student_year_passed VARCHAR(10) NOT NULL,
-> PRIMARY KEY (student_id)
->);
Query OK, 0 rows affected (0.014 sec)

MariaDB [studentapp]>

```

21. Hit the Public IP of the Jump-server which is present in the Public-subnet.
22. The webpage is successfully hosted.

Instances | EC2 | us-east-1 x User Data x +

Not secure | 18.232.169.67

## Student Registration Form

Student Name  
Student Address  
Student Age  
Student Qualification  
Student Percentage  
Year Passed

23. The data is stored in the RDS that we had created.

Not secure | 18.232.169.67/viewStudents

[Register Student](#)

## Students List

| Student ID | StudentName    | Student Addr | Student Age | Student Qualification | Student Percentage | Student Year Passed | Edit                 | Delete                 |
|------------|----------------|--------------|-------------|-----------------------|--------------------|---------------------|----------------------|------------------------|
| 1          | Akash Bhau     | Nagar        | 27          | Bsc                   | 97                 | 2022                | <a href="#">edit</a> | <a href="#">delete</a> |
| 2          | Sonali Khurade | katraj       | 27          | Bsc                   | 98                 | 2022                | <a href="#">edit</a> | <a href="#">delete</a> |
| 3          | Swati Jadhav   | Ravet        | 25          | Bsc                   | 95                 | 2022                | <a href="#">edit</a> | <a href="#">delete</a> |

## Summary:

Hosted a website and stored the data in the database (RDS). Created a **VPC** and **three subnets** in different **Availability Zones**. which **one public subnet** and **two private subnets**. In the public subnet, we created a **jump server** and in one private subnet where we were going to host the application in the **application server** and in another private subnet the **database server** for storing data. Configured **Nginx in the jump server, java, and application into the application server. Install MariaDB in the database and create a table**. This is how we hosted an application in one server and stored data into another server.

## Errors in 3-tier application