Dynamic Website Using Tomcat, RDS & Nginx

(3 tier project)

Actual Project Flow:-

Step 1: Create VPC.

Step 2: Create two subnets (private and public).

Step 3: Create Route Table & Assign private subnet to newly created route table (using Edit route table association in subnet Dashboard).

Step 4: Create 1st instance in private subnet for setting up actual server.

Step 5: Create 2nd instance in public subnet for setting up NGINX server.

Step 6: Create RDS database (note: create RDS database in the same VPC where our instances are created).

Step 7: Perform Tomcat setup in 1st instance (install Tomcat, write or copy Java code in Webapps folder, perform configuration in context.xml file).

Step 8: Configure NGINX in 2nd instance (configure proxy).

Step 9: Access the website using 2nd instance public IP.

What is Tomcat server??

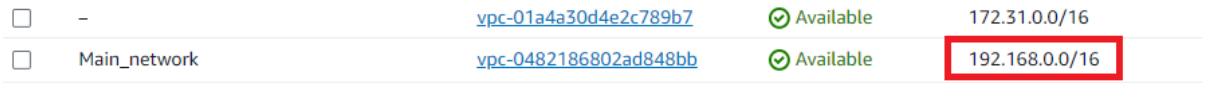
With the help of Tomcat server we can host dynamic websites. It support java language…

What is Nginx Server??

It is proxy server means that any request coming to your NGINX server will be forwarded to your Tomcat server...

❖ **Step1**:- Creating VPC

Here We are creating our own VPC because we want our application in our private network….



VPC Name = Main_network
CIDR = 192.168.0.0/16

❖ **Step2**:- Create two subnets (private and public)

Here we are creating two subnets because we want our main server is in private network and our proxy server is in public network...

Reason:- suppose we gave internet access to all subnets, and launch tomcat server and nginx server then **hacker** can able to access the nginx server as well and our main server as well..... We do not want a hacker to access our main server (tomcat server)
To avoid this issue we create our nginx server in public subnet and tomcat server in private subnet…

What is public subnet??
Public subnet means we are just assigning Internet Gateway and adding Internet Gateway in Route Tables as simple as that….

Process: Select the subnet → Action → Edit route table association → Select route Table → Save

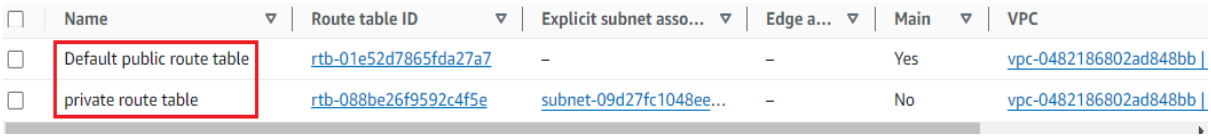
What is private subnet??
Private means we are not assigning Internet Gateway and doesn't assigning Internet Gateway in Route Tables as simple as that….

Process: Select the subnet → Action → Edit route table association → Select route Table → Save

❖ **Step3**:- Create Route Table

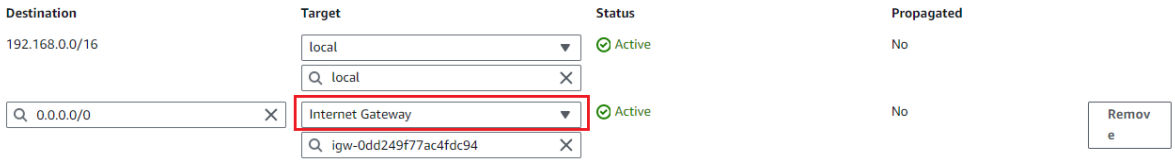
For making one subnet private and one subnet public we need to add our subnets is in different route tables…..

Here we are creating only one route table because when we created our VPC the default Route table is automatically created….



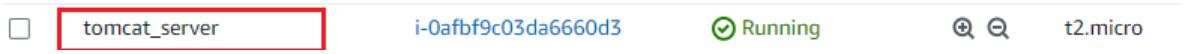
What is private route table?? We are not specify any internet route to the route table… (0.0.0.0/0) ✂

What is public route table?? We are assigning the internet access to route table…….. (0.0.0.0/0) ✓



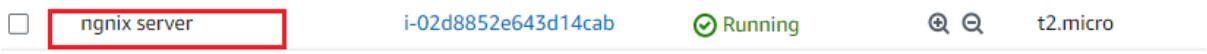
❖ **Step4:-** Create Private instance

While creating private instance make sure to add private subnet which we are created previously…..
Make sure to add ports in security group while creating instance (8080 for tomcat, 22 for ssh , 3306 for MYSQL database)



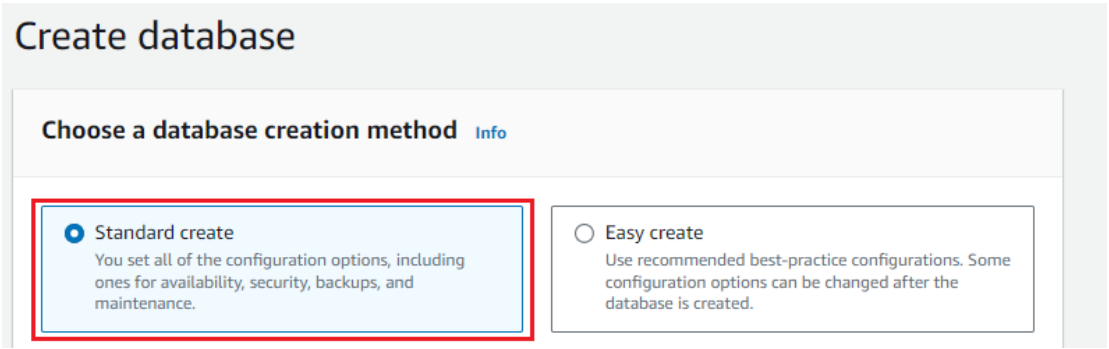
❖ **Step5:-** Create Public instance

While creating private instance make sure to add Public subnet which we are created previously…..
Make sure to add ports in security group while creating instance (80 for nginx proxy , 22 for ssh)

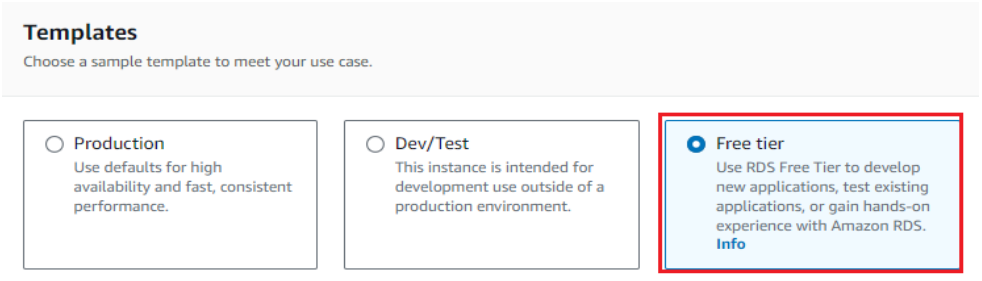


❖ **Step 6:-** Create RDS database

- Search RDS service in Aws dashboard
- Click on create database
- Select Standard create (it will provide more options For us)



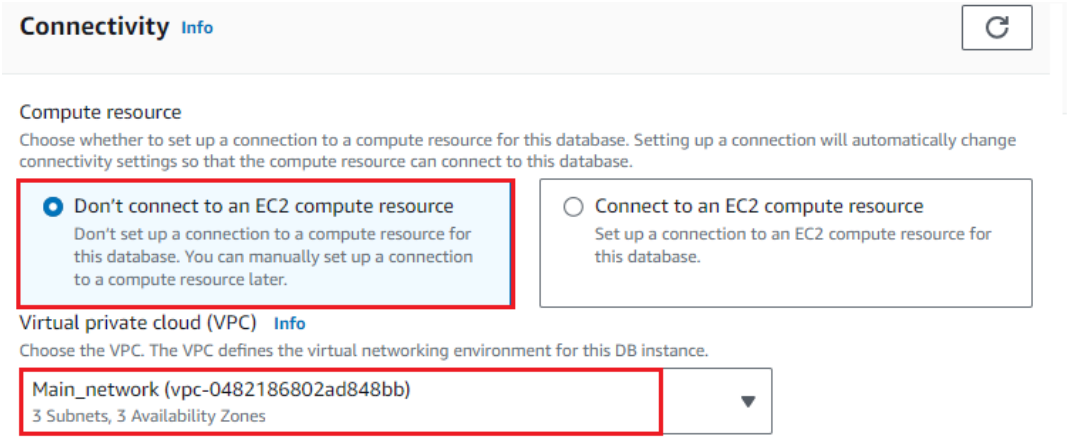
- Select Engine and Engine Version as per your requirement
- Select the below template (because we are not placed yet 😊)



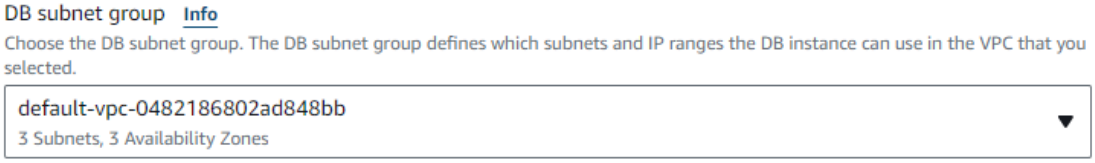
- Give name as per your choice
- Create Master username and password as per your choice
  Username ➔ admin
  Password ➔ 12345678

- Select storage type

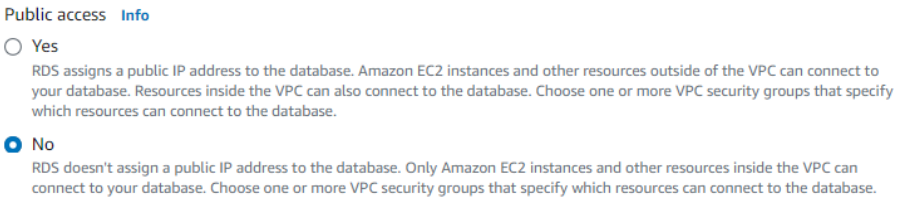
- In Connectivity select the below options



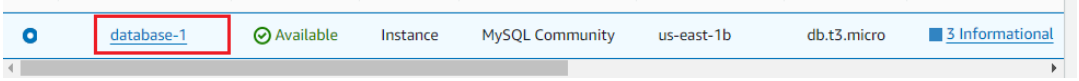
- Select the DB subnet group (here we are selecting default VPC group subnets)



- Select public access = no



- Click on create database
- **Database created successfully……**



❖ **Step 7:-** Configuring application (student.war) in **first instance** using apache Tomcat server…

- Connect to The 1st instance
  We are connecting 1st instance using 2nd instance (because we are not assigned the public subnet or any network to 1St instance)

- I created a bash script for basic setup (just paste the script in example.sh file and run using bash command)…Boom 🔥🔥🔥

```
#!/bin/bash
sudo yum install git -y
sudo yum install java -y
sudo git clone https://github.com/mayur4279/tmp /opt/tmp
sudo wget https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.87/bin/apache-tomcat-9.0.87.zip -P /opt
sudo unzip /opt/apache-tomcat-9.0.87.zip -d /opt
sudo cp -rvf /opt/tmp/student.war /opt/apache-tomcat-9.0.87/webapps
sudo cp -rvf /opt/tmp/mysql-connector.jar /opt/apache-tomcat-9.0.87/lib
sudo chmod +x /opt/apache-tomcat-9.0.87/bin/catalina.sh
sudo  rm -rvf  /opt/tmp
sudo  rm -rvf  /opt/apache-tomcat-9.0.87.zip
# Start Tomcat
sudo bash /opt/apache-tomcat-9.0.87/bin/catalina.sh start
```

- Connectivity configuration For connecting RDS database

```
<Context>
        <Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource"
                maxTotal="100" maxIdle="30" maxWaitMillis="10000"
                username="admin" password="12345678" driverClassName="com.mysql.jdbc.Driver"
                url="jdbc:mysql://database-1.cp008gi66hcz.us-east-1.rds.amazonaws.com:3306/studentapp"/>

    <!-- Default set of monitored resources. If one of these changes, the    -->
    <!-- web application will be reloaded.                                   -->
    <WatchedResource>WEB-INF/web.xml</WatchedResource>
    <WatchedResource>WEB-INF/tomcat-web.xml</WatchedResource>
    <WatchedResource>${catalina.base}/conf/web.xml</WatchedResource>
```

Note: we already copied mysql-connector.jar file using bash script..

- Connect to RDS Database using below command

```
Mysql  -h <endpoint of RDS>   -u admin –p12345678
```

- Create schema for our database

Why we create schema??

A schema is created in a database to define the structure, organization, and relationships of data. It ensures data integrity, optimizes query performance, and facilitates security measures through permissions and access controls. Additionally, the schema serves as documentation for the database structure, aiding in understanding and managing the data effectively.

Enter Below commands for creating schema for our Application.

```
create database studentapp;

use 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)
);
```

- **Schema created successfully….**

Now we can able to append application data in our RDS database....

➢ **Step8:-** Configure NGINX Server in **second** instance

- Connect to the 2nd instance
- Download the nginx service

```
sudo yum install nginx –y
```

- Configure the nginx

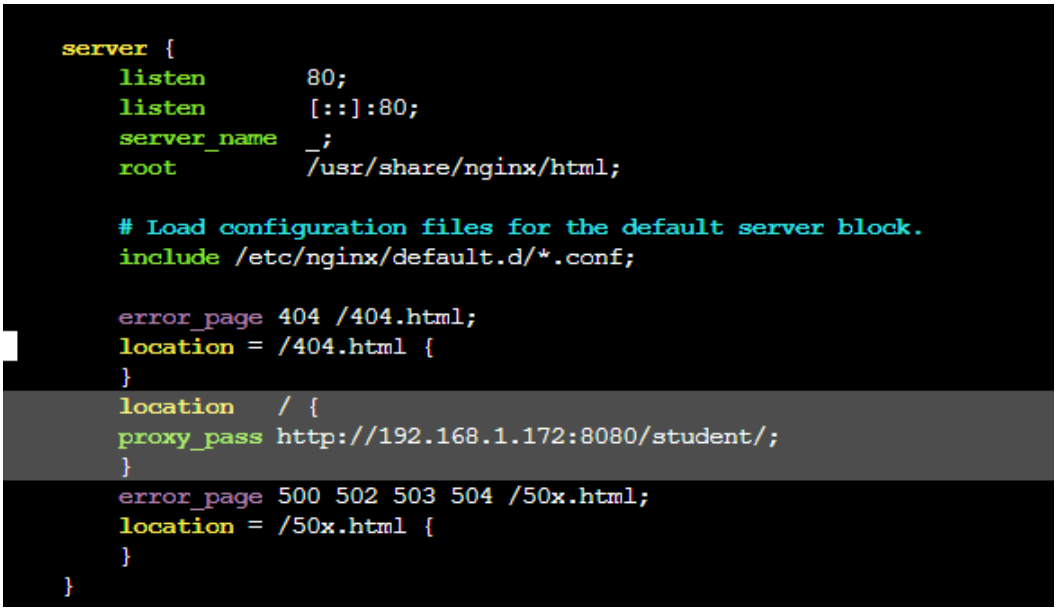
```
Vim /etc/nginx/nginx.conf
```

Add this code in nginx.conf file…

```
location / {
proxy_pass http://192.168.1.172:8080/student/;
}
```

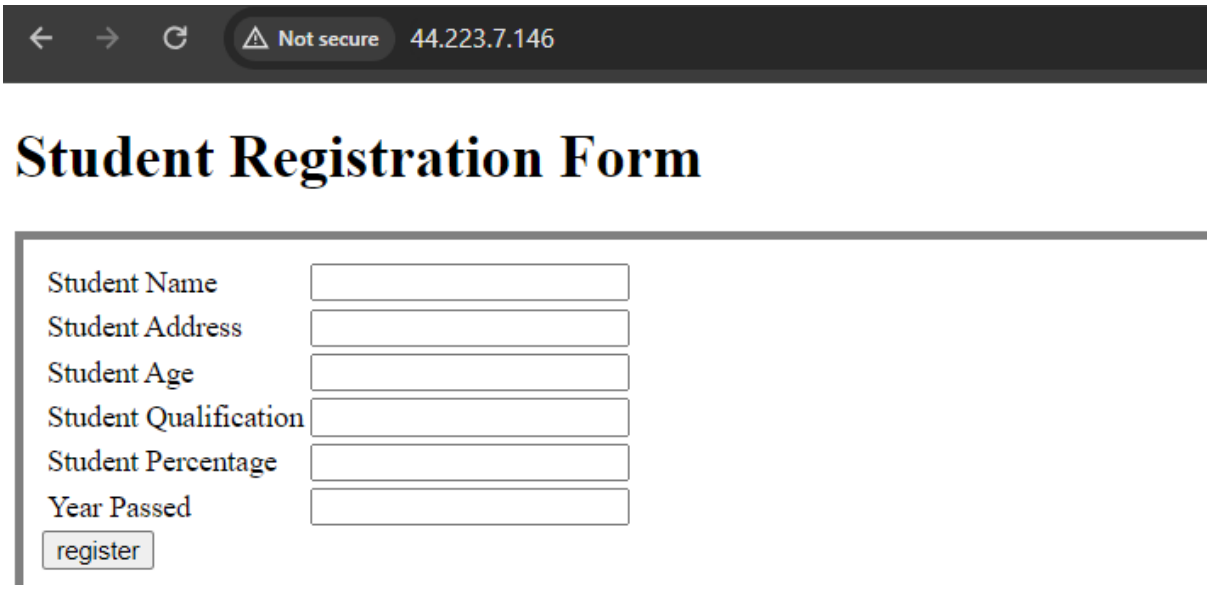
Note:- Here we are putting private ip of 1st instance. Because we Hosted our Application in private subnet…..

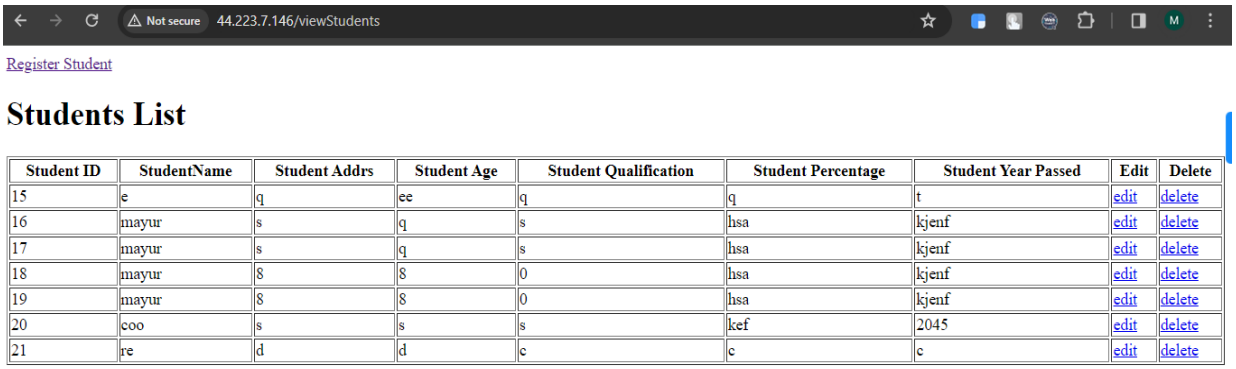
- The Actual image of the configuration file



❖ **Step9:-** Access the website using public IP of NGINX Server

Public IP = 44.223.7.146





Student ID	StudentName	Student Addrs	Student Age	Student Qualification	Student Percentage	Student Year Passed	Edit	Delete
15	e	q	ee	q	q	t	edit	delete
16	mayur	s	q	s	hsa	kjenf	edit	delete
17	mayur	s	q	s	hsa	kjenf	edit	delete
18	mayur	8	8	0	hsa	kjenf	edit	delete
19	mayur	8	8	0	hsa	kjenf	edit	delete
20	coo	s	s	s	kef	2045	edit	delete
21	re	d	d	c	c	c	edit	delete

Overall Architecture Diagram:-