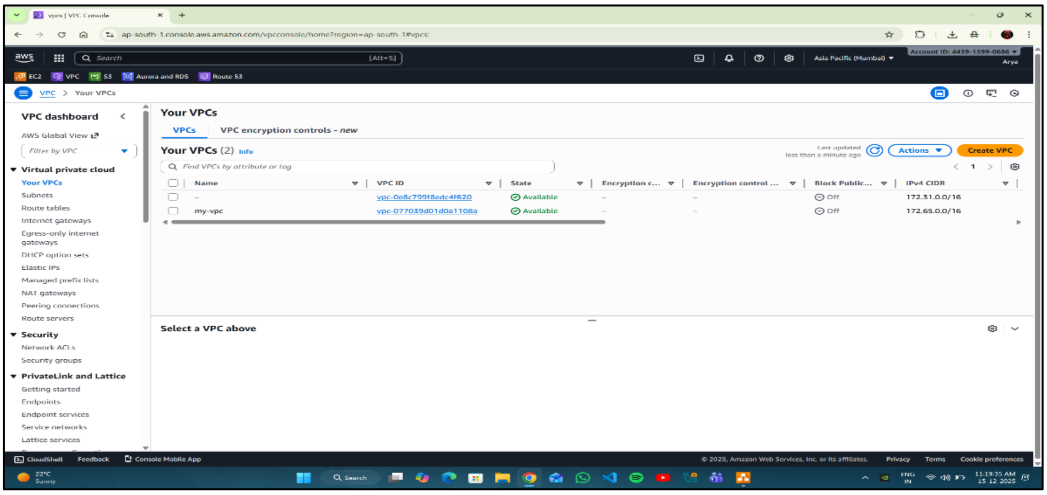
Title: 3-Tier project (Tomcat & MySQL)

Step 1: Create VPC –

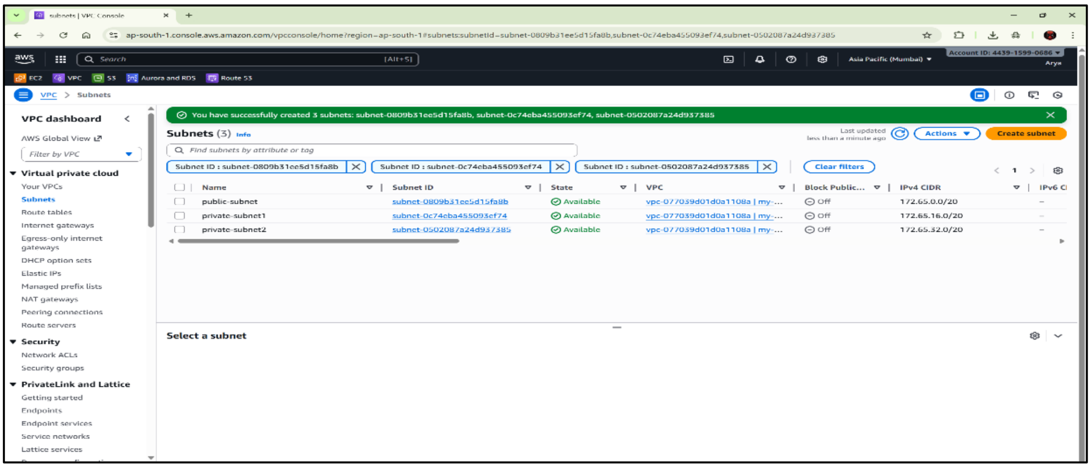


Your VPCs (2)

Name	VPC ID	State	Encryption c...	Encryption control ...	Block Public...	IPv4 CIDR
–	vpc-0e8c799f8edc4f620	Available	–	–	Off	172.31.0.0/16
my-vpc	vpc-077039d01d0a1108a	Available	–	–	Off	172.65.0.0/16

Select a VPC above

Step 2: Create 3 subnets of this VPC **(public, private, private-2)** –



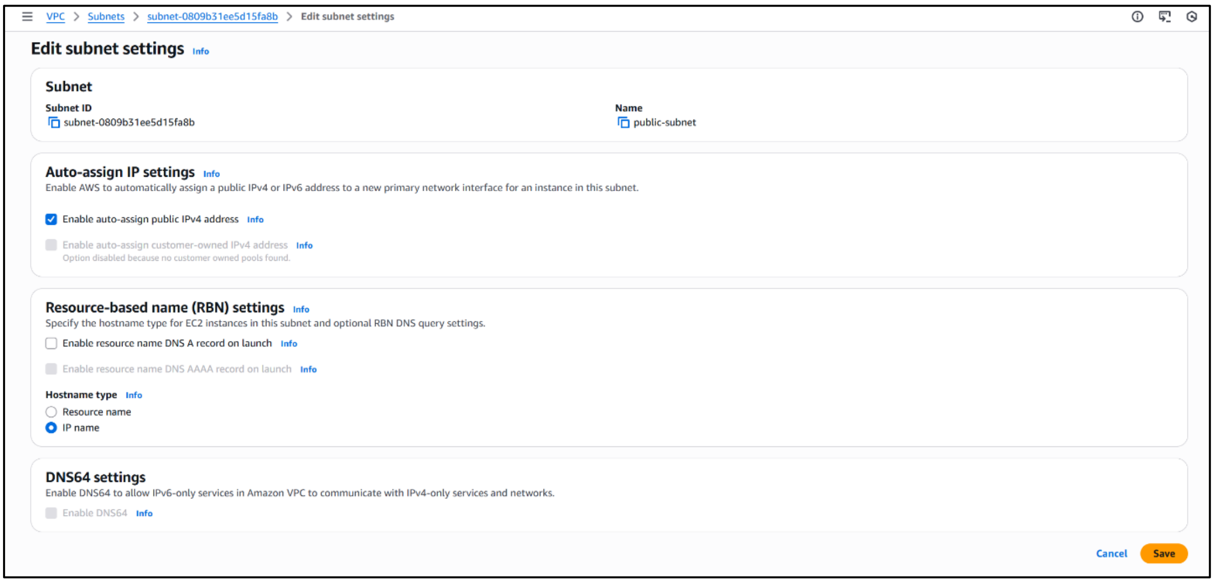
You have successfully created 3 subnets: subnet-0809b31ee5d15fa8b, subnet-0c74eba455093ef74, subnet-0502087a24d937385

Subnets (3)

Name	Subnet ID	State	VPC	Block Public...	IPv4 CIDR	IPv6 C
public-subnet	subnet-0809b31ee5d15fa8b	Available	vpc-077039d01d0a1108a | my-...	Off	172.65.0.0/20	–
private-subnet1	subnet-0c74eba455093ef74	Available	vpc-077039d01d0a1108a | my-...	Off	172.65.16.0/20	–
private-subnet2	subnet-0502087a24d937385	Available	vpc-077039d01d0a1108a | my-...	Off	172.65.32.0/20	–

Select a subnet

Step 3: Make a public subnet - In **Edit subnet setting** –



VPC > Subnets > subnet-0809b31ee5d15fa8b > Edit subnet settings

Edit subnet settings Info

Subnet

Subnet ID: subnet-0809b31ee5d15fa8b

Name: public-subnet

Auto-assign IP settings Info

Enable AWS to automatically assign a public IPv4 or IPv6 address to a new primary network interface for an instance in this subnet.

☑ Enable auto-assign public IPv4 address Info

☐ Enable auto-assign customer-owned IPv4 address Info
Option disabled because no customer owned pools found.

Resource-based name (RBN) settings Info

Specify the hostname type for EC2 instances in this subnet and optional RBN DNS query settings.

☐ Enable resource name DNS A record on launch Info

☐ Enable resource name DNS AAAA record on launch Info

Hostname type Info

○ Resource name

● IP name

DNS64 settings

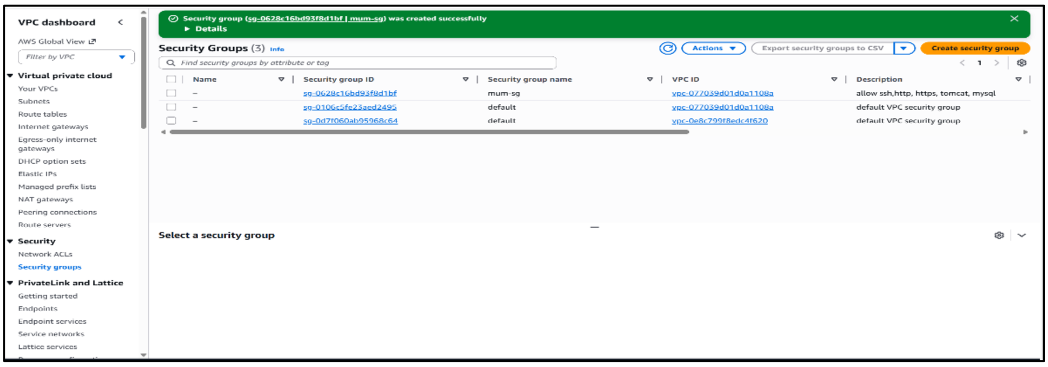
Enable DNS64 to allow IPv6-only services in Amazon VPC to communicate with IPv4-only services and networks.

☐ Enable DNS64 Info

Cancel | Save

Title: 3-Tier project (Tomcat & MySQL)

Step 4: Now create a Security Group – Allow SSH, Tomcat, MySQL, HTTP, HTTPS.

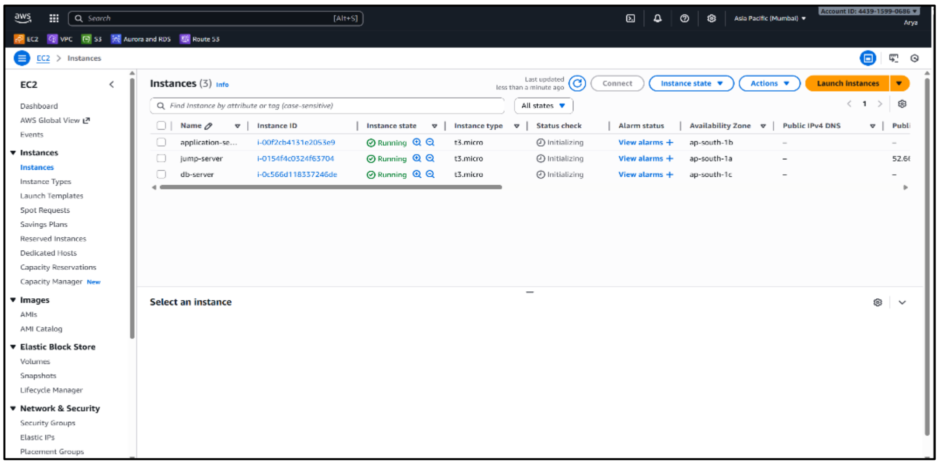


Security group (sg-0628c16bd93f8d1bf | mum-sg) was created successfully

Security Groups (3)

Name	Security group ID	Security group name	VPC ID	Description
–	sg-0628c16bd93f8d1bf	mum-sg	vpc-077039d01d0a1108a	allow ssh,http, https, tomcat, mysql
–	sg-0106c5fe23aed2495	default	vpc-077039d01d0a1108a	default VPC security group
–	sg-0d7f060ab95968c64	default	vpc-0e8c799f8edc4f620	default VPC security group

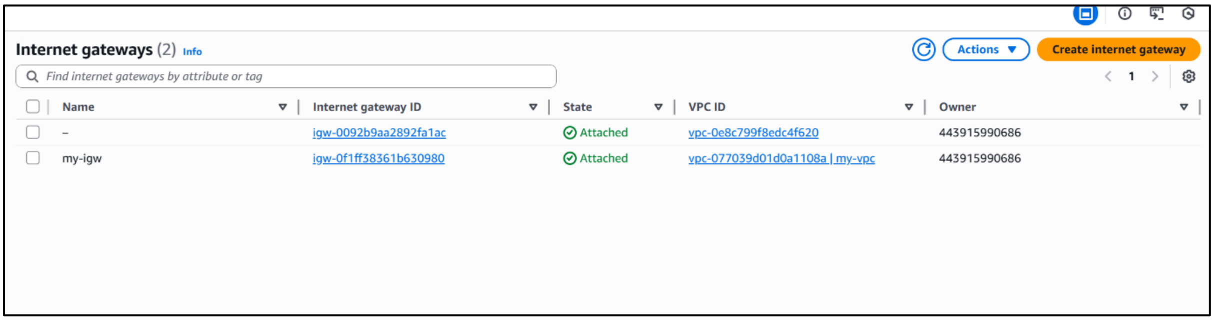
Step 5: Create instances – jump server in Public subnet, application server in private-subnet1 and Database server in private-subnet2. And we created a Security Group.



Instances (3)

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Publi
application-se...	i-00f2cb4131e2053e9	Running	t3.micro	Initializing	View alarms +	ap-south-1b	–	–
jump-server	i-0154f4c0324f63704	Running	t3.micro	Initializing	View alarms +	ap-south-1a	–	52.6
db-server	i-0c566d118337246de	Running	t3.micro	Initializing	View alarms +	ap-south-1c	–	–

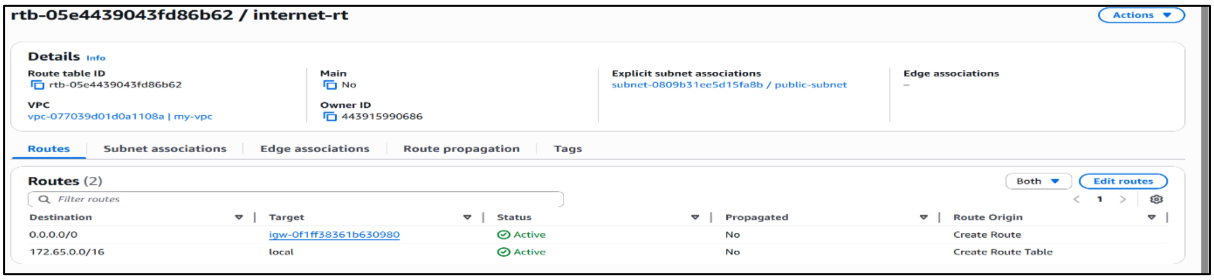
Step 6: Now create an Internet Gateway – It will give Internet to the VPC.



Internet gateways (2)

Name	Internet gateway ID	State	VPC ID	Owner
–	igw-0092b9aa2892fa1ac	Attached	vpc-0e8c799f8edc4f620	443915990686
my-igw	igw-0f1ff38361b630980	Attached	vpc-077039d01d0a1108a | my-vpc	443915990686

Step 7: Now create a Route table, and Route the Internet Gateway, and associate the Subnets with public only.



rtb-05e4439043fd86b62 / internet-rt

Details

Route table ID: rtb-05e4439043fd86b62

VPC: vpc-077039d01d0a1108a | my-vpc

Main: No

Owner ID: 443915990686

Explicit subnet associations: subnet-0809b31ee5d15fa8b / public-subnet

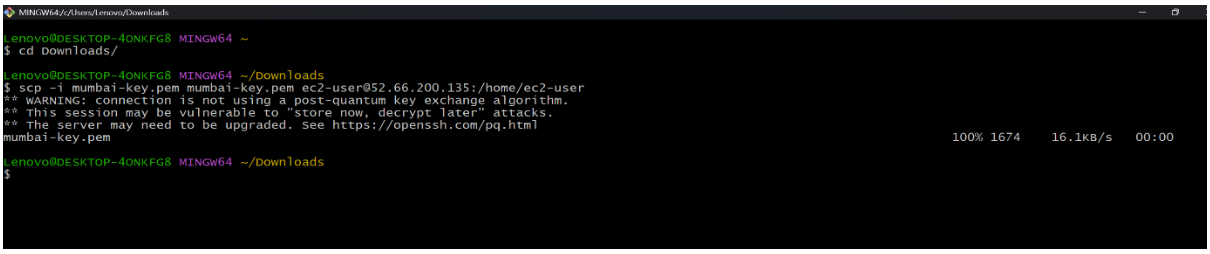
Edge associations: –

Routes (2)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	igw-0f1ff38361b630980	Active	No	Create Route
172.65.0.0/16	local	Active	No	Create Route Table

Title: 3-Tier project (Tomcat & MySQL)

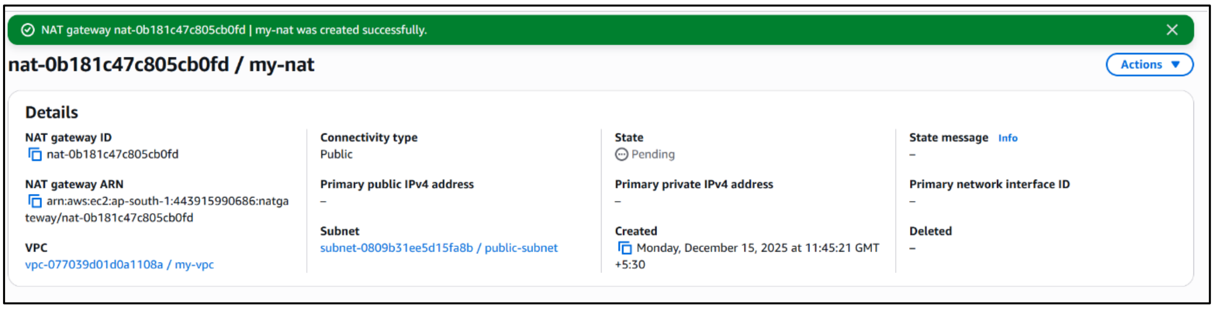
Step 8: Transfer the Key to the Jump server to get access to the application server



MINGW64:/c/Users/Lenovo/Downloads

```
Lenovo@DESKTOP-4ONKFG8 MINGW64 ~
$ cd Downloads/

Lenovo@DESKTOP-4ONKFG8 MINGW64 ~/Downloads
$ scp -i mumbai-key.pem mumbai-key.pem ec2-user@52.66.200.135:/home/ec2-user
** WARNING: connection is not using a post-quantum key exchange algorithm.
** This session may be vulnerable to "store now, decrypt later" attacks.
** The server may need to be upgraded. See https://openssh.com/pq.html
mumbai-key.pem                                                                          100% 1674    16.1KB/s   00:00

Lenovo@DESKTOP-4ONKFG8 MINGW64 ~/Downloads
$
```

Step 9: After accessing the Application server needs internet so create a NAT Gateway in the public subnet.



NAT gateway nat-0b181c47c805cb0fd | my-nat was created successfully.

nat-0b181c47c805cb0fd / my-nat

Details

NAT gateway ID: nat-0b181c47c805cb0fd

NAT gateway ARN: arn:aws:ec2:ap-south-1:443915990686:natgateway/nat-0b181c47c805cb0fd

VPC: vpc-077039d01d0a1108a / my-vpc

Connectivity type: Public

Primary public IPv4 address: –

Subnet: subnet-0809b31ee5d15fa8b / public-subnet

State: Pending

Primary private IPv4 address: –

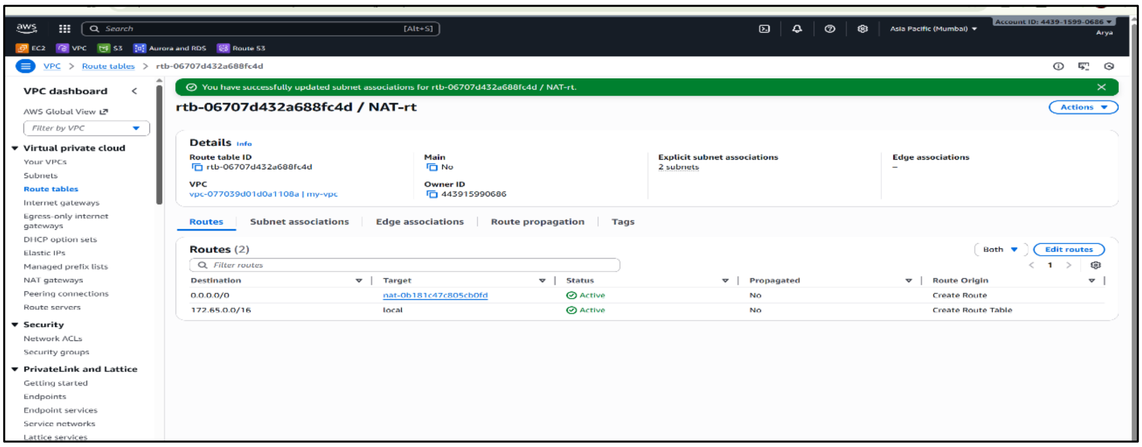
Created: Monday, December 15, 2025 at 11:45:21 GMT +5:30

State message Info: –

Primary network interface ID: –

Deleted: –

Step 10: After this, create a Route table for the NAT gateway associated with private subnets – Route NAT Gateway and Associate Subnets.



You have successfully updated subnet associations for rtb-06707d432a688fc4d / NAT-rt.

rtb-06707d432a688fc4d / NAT-rt

Details Info

Route table ID: rtb-06707d432a688fc4d

VPC: vpc-077039d01d0a1108a | my-vpc

Main: No

Owner ID: 443915990686

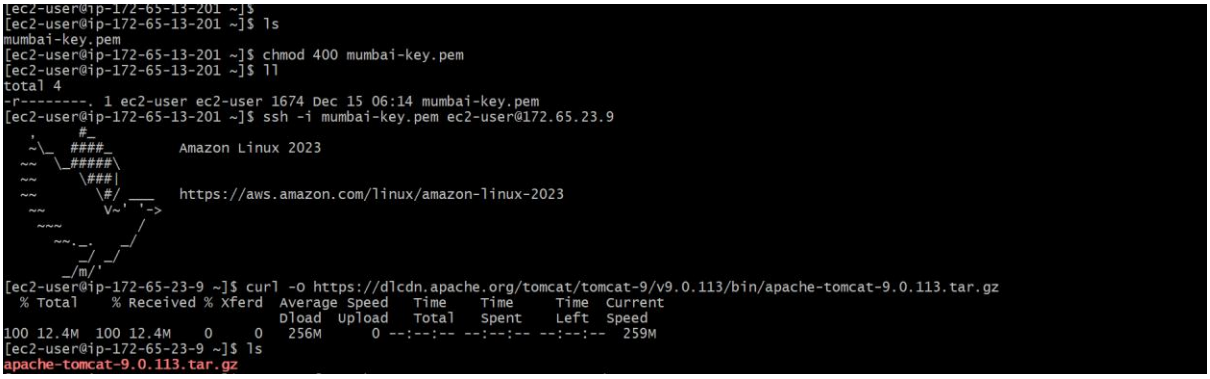
Explicit subnet associations: 2 subnets

Edge associations: –

Routes (2)

Destination	Target	Status	Propagated	Route Origin
0.0.0.0/0	nat-0b181c47c805cb0fd	Active	No	Create Route
172.65.0.0/16	local	Active	No	Create Route Table

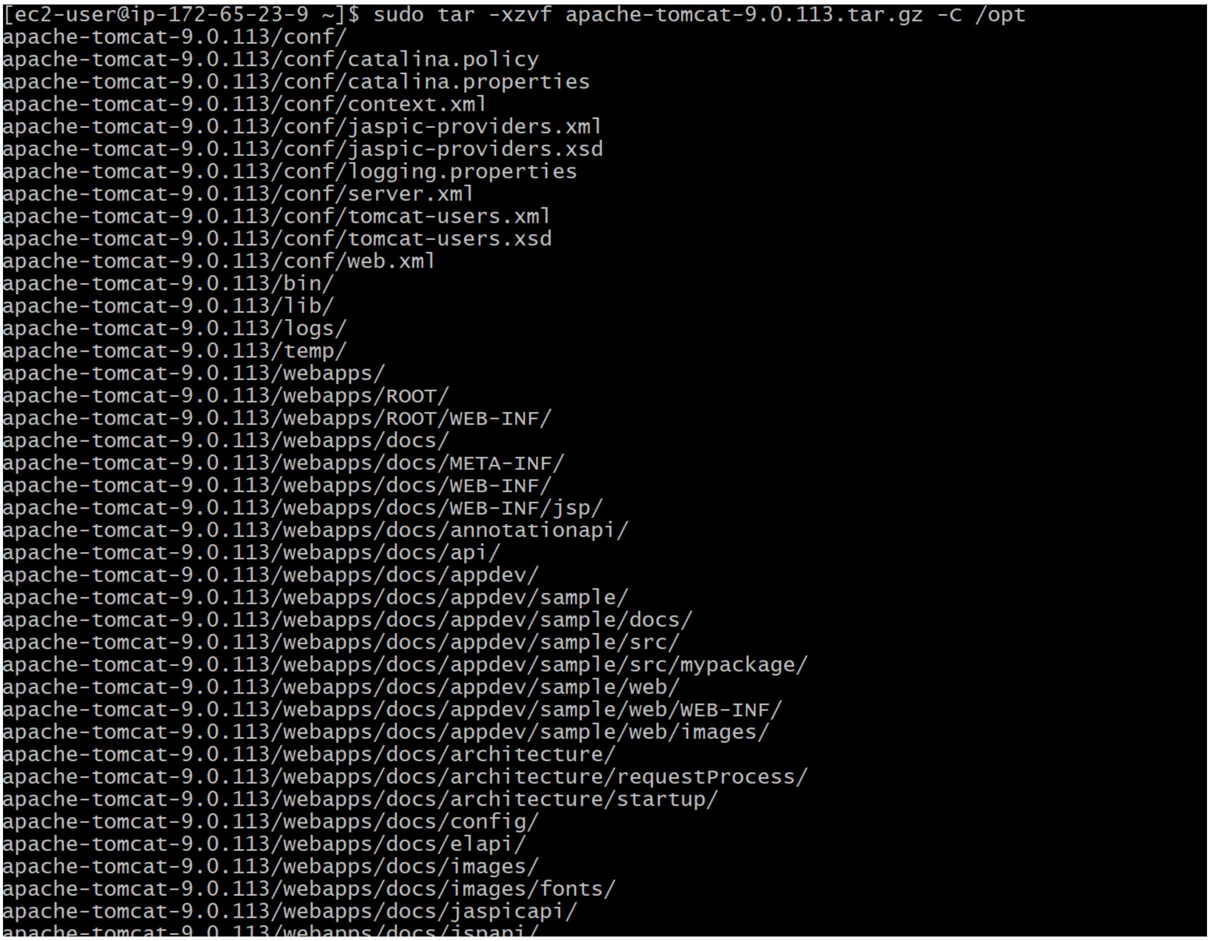
Step 11: Access the jump server and change the key permissions, and get access to the Application server and download Apache Tomcat in the application server.



```
[ec2-user@ip-172-65-13-201 ~]$
[ec2-user@ip-172-65-13-201 ~]$ ls
mumbai-key.pem
[ec2-user@ip-172-65-13-201 ~]$ chmod 400 mumbai-key.pem
[ec2-user@ip-172-65-13-201 ~]$ ll
total 4
-r--------. 1 ec2-user ec2-user 1674 Dec 15 06:14 mumbai-key.pem
[ec2-user@ip-172-65-13-201 ~]$ ssh -i mumbai-key.pem ec2-user@172.65.23.9
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-65-23-9 ~]$ curl -O https://dlcdn.apache.org/tomcat/tomcat-9/v9.0.113/bin/apache-tomcat-9.0.113.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 12.4M  100 12.4M    0     0   256M      0 --:--:-- --:--:-- --:--:--  259M
[ec2-user@ip-172-65-23-9 ~]$ ls
apache-tomcat-9.0.113.tar.gz
```

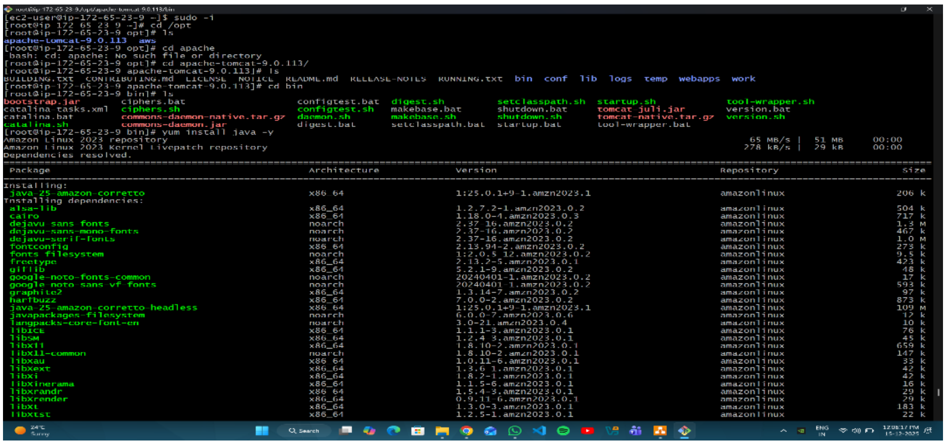
Title: 3-Tier project (Tomcat & MySQL)

Step 12: Extract this file to the/opt folder.



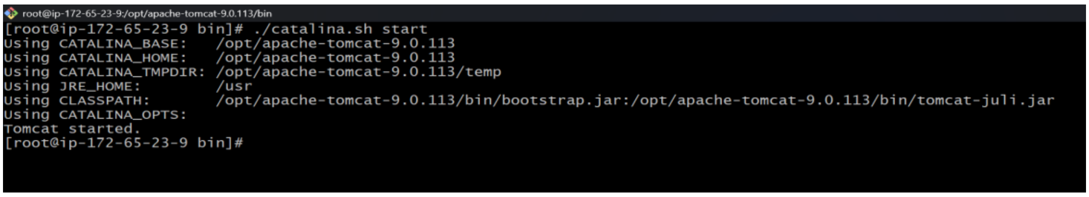
```
[ec2-user@ip-172-65-23-9 ~]$ sudo tar -xzvf apache-tomcat-9.0.113.tar.gz -C /opt
apache-tomcat-9.0.113/conf/
apache-tomcat-9.0.113/conf/catalina.policy
apache-tomcat-9.0.113/conf/catalina.properties
apache-tomcat-9.0.113/conf/context.xml
apache-tomcat-9.0.113/conf/jaspic-providers.xml
apache-tomcat-9.0.113/conf/jaspic-providers.xsd
apache-tomcat-9.0.113/conf/logging.properties
apache-tomcat-9.0.113/conf/server.xml
apache-tomcat-9.0.113/conf/tomcat-users.xml
apache-tomcat-9.0.113/conf/tomcat-users.xsd
apache-tomcat-9.0.113/conf/web.xml
apache-tomcat-9.0.113/bin/
apache-tomcat-9.0.113/lib/
apache-tomcat-9.0.113/logs/
apache-tomcat-9.0.113/temp/
apache-tomcat-9.0.113/webapps/
apache-tomcat-9.0.113/webapps/ROOT/
apache-tomcat-9.0.113/webapps/ROOT/WEB-INF/
apache-tomcat-9.0.113/webapps/docs/
apache-tomcat-9.0.113/webapps/docs/META-INF/
apache-tomcat-9.0.113/webapps/docs/WEB-INF/
apache-tomcat-9.0.113/webapps/docs/WEB-INF/jsp/
apache-tomcat-9.0.113/webapps/docs/annotationapi/
apache-tomcat-9.0.113/webapps/docs/api/
apache-tomcat-9.0.113/webapps/docs/appdev/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/docs/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/src/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/src/mypackage/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/web/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/web/WEB-INF/
apache-tomcat-9.0.113/webapps/docs/appdev/sample/web/images/
apache-tomcat-9.0.113/webapps/docs/architecture/
apache-tomcat-9.0.113/webapps/docs/architecture/requestProcess/
apache-tomcat-9.0.113/webapps/docs/architecture/startup/
apache-tomcat-9.0.113/webapps/docs/config/
apache-tomcat-9.0.113/webapps/docs/elapi/
apache-tomcat-9.0.113/webapps/docs/images/
apache-tomcat-9.0.113/webapps/docs/images/fonts/
apache-tomcat-9.0.113/webapps/docs/jaspicapi/
apache-tomcat-9.0.113/webapps/docs/jspapi/
```

Step 13: Go to **/opt/apache-tomcat-9.0.113/** and in that **bin/** here you see catalina.sh, so we have to start Tomcat with this file, **but Tomcat needs Java to start, so first install Java.**



```
[ec2-user@ip-172-65-23-9 ~]$ sudo -i
[root@ip-172-65-23-9 ~]# cd /opt
[root@ip-172-65-23-9 opt]# ls
apache-tomcat-9.0.113  aws
[root@ip-172-65-23-9 opt]# cd apache
-bash: cd: apache: No such file or directory
[root@ip-172-65-23-9 opt]# cd apache-tomcat-9.0.113/
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# ls
BUILDING.txt  CONTRIBUTING.md  LICENSE  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  bin  conf  lib  logs  temp  webapps  work
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# cd bin
[root@ip-172-65-23-9 bin]# ls
bootstrap.jar        ciphers.bat                    configtest.bat  digest.sh          setclasspath.sh  startup.sh            tool-wrapper.sh
catalina-tasks.xml   ciphers.sh                     configtest.sh   makebase.bat       shutdown.bat     tomcat-juli.jar       version.bat
catalina.bat         commons-daemon-native.tar.gz   daemon.sh       makebase.sh        shutdown.sh      tomcat-native.tar.gz  version.sh
catalina.sh          commons-daemon.jar             digest.bat      setclasspath.bat   startup.bat      tool-wrapper.bat
[root@ip-172-65-23-9 bin]# yum install java -y
Amazon Linux 2023 repository                                                          65 MB/s |  51 MB     00:00
Amazon Linux 2023 Kernel Livepatch repository                                        278 kB/s |  29 kB     00:00
Dependencies resolved.
=====================================================================================================================
 Package                                 Architecture       Version                           Repository        Size
=====================================================================================================================
Installing:
 java-25-amazon-corretto                 x86_64             1:25.0.1+9-1.amzn2023.1           amazonlinux      206 k
Installing dependencies:
 alsa-lib                                x86_64             1.2.7.2-1.amzn2023.0.2            amazonlinux      504 k
 cairo                                   x86_64             1.18.0-4.amzn2023.0.3             amazonlinux      717 k
 dejavu-sans-fonts                       noarch             2.37-16.amzn2023.0.2              amazonlinux      1.3 M
 dejavu-sans-mono-fonts                  noarch             2.37-16.amzn2023.0.2              amazonlinux      467 k
 dejavu-serif-fonts                      noarch             2.37-16.amzn2023.0.2              amazonlinux      1.0 M
 fontconfig                              x86_64             2.13.94-2.amzn2023.0.2            amazonlinux      273 k
 fonts-filesystem                        noarch             1:2.0.5-12.amzn2023.0.2           amazonlinux      9.5 k
 freetype                                x86_64             2.13.2-5.amzn2023.0.1             amazonlinux      423 k
 giflib                                  x86_64             5.2.1-9.amzn2023.0.2              amazonlinux       48 k
 google-noto-fonts-common                noarch             20240401-1.amzn2023.0.2           amazonlinux       17 k
 google-noto-sans-vf-fonts               noarch             20240401-1.amzn2023.0.2           amazonlinux      593 k
 graphite2                               x86_64             1.3.14-7.amzn2023.0.2             amazonlinux       97 k
 harfbuzz                                x86_64             7.0.0-2.amzn2023.0.2              amazonlinux      873 k
 java-25-amazon-corretto-headless        x86_64             1:25.0.1+9-1.amzn2023.1           amazonlinux      109 M
 javapackages-filesystem                 noarch             6.0.0-7.amzn2023.0.6              amazonlinux       12 k
 langpacks-core-font-en                  noarch             3.0-21.amzn2023.0.4               amazonlinux       10 k
 libICE                                  x86_64             1.1.1-3.amzn2023.0.1              amazonlinux       76 k
 libSM                                   x86_64             1.2.4-3.amzn2023.0.1              amazonlinux       45 k
 libX11                                  x86_64             1.8.10-2.amzn2023.0.1             amazonlinux      659 k
 libX11-common                           noarch             1.8.10-2.amzn2023.0.1             amazonlinux      147 k
 libXau                                  x86_64             1.0.11-6.amzn2023.0.1             amazonlinux       33 k
 libXext                                 x86_64             1.3.6-1.amzn2023.0.1              amazonlinux       42 k
 libXi                                   x86_64             1.8.2-1.amzn2023.0.1              amazonlinux       42 k
 libXinerama                             x86_64             1.1.5-6.amzn2023.0.1              amazonlinux       16 k
 libXrandr                               x86_64             1.5.4-3.amzn2023.0.1              amazonlinux       29 k
 libXrender                              x86_64             0.9.11-6.amzn2023.0.1             amazonlinux       29 k
 libXt                                   x86_64             1.3.0-3.amzn2023.0.1              amazonlinux      183 k
 libXtst                                 x86_64             1.2.5-1.amzn2023.0.1              amazonlinux       22 k
```

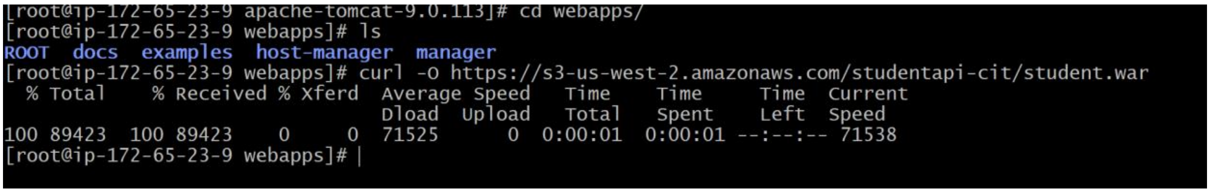
Step 14: Now start Tomcat - **./catalina.sh start**



```
[root@ip-172-65-23-9 bin]# ./catalina.sh start
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.113
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.113
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.113/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.113/bin/bootstrap.jar:/opt/apache-tomcat-9.0.113/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
[root@ip-172-65-23-9 bin]#
```

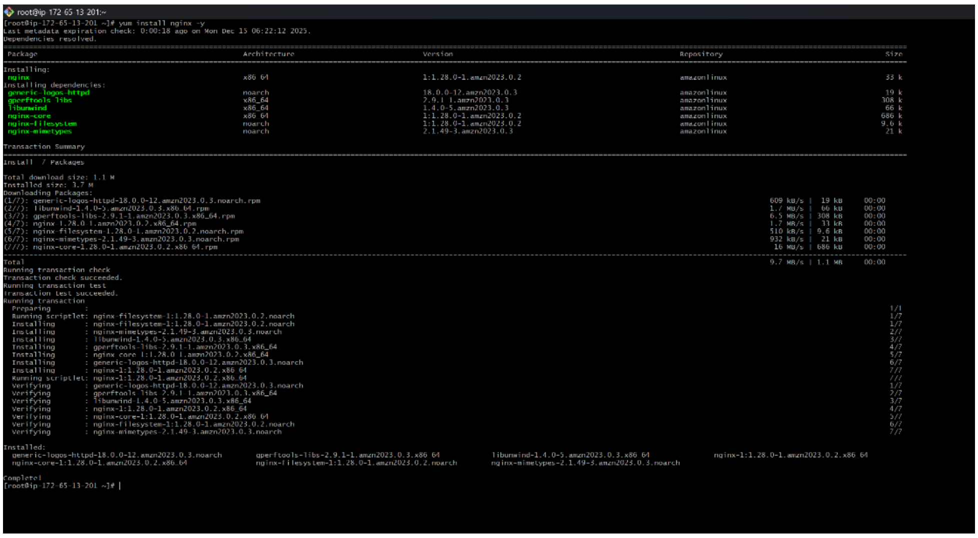
Step 15: Now we have to dump the website in the application server –

so go to - /opt/apache-tomcat-9.0.113/webapps/



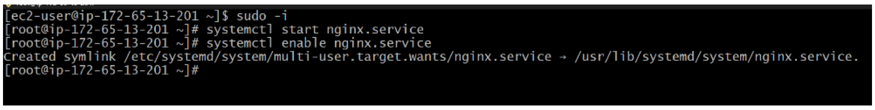
```
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# cd webapps/
[root@ip-172-65-23-9 webapps]# ls
ROOT  docs  examples  host-manager  manager
[root@ip-172-65-23-9 webapps]# curl -O https://s3-us-west-2.amazonaws.com/studentapi-cit/student.war
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 89423  100 89423    0     0  71525      0  0:00:01  0:00:01 --:--:-- 71538
[root@ip-172-65-23-9 webapps]# |
```

Step 16: Now everything is set, but we have to give the location to the jump server. We need nginx, so install nginx in the jump server.



```
[root@ip-172-65-13-201 ~]# yum install nginx -y
Last metadata expiration check: 0:00:18 ago on Mon Dec 15 06:22:12 2025.
Dependencies resolved.
================================================================================================================
 Package                      Architecture          Version                          Repository           Size
================================================================================================================
Installing:
 nginx                        x86_64                1:1.28.0-1.amzn2023.0.2          amazonlinux          33 k
Installing dependencies:
 generic-logos-httpd          noarch                18.0.0-12.amzn2023.0.3           amazonlinux          19 k
 gperftools-libs              x86_64                2.9.1-1.amzn2023.0.3             amazonlinux         308 k
 libunwind                    x86_64                1.4.0-5.amzn2023.0.3             amazonlinux          66 k
 nginx-core                   x86_64                1:1.28.0-1.amzn2023.0.2          amazonlinux         686 k
 nginx-filesystem             noarch                1:1.28.0-1.amzn2023.0.2          amazonlinux         9.6 k
 nginx-mimetypes              noarch                2.1.49-3.amzn2023.0.3            amazonlinux          21 k

Transaction Summary
================================================================================================================
Install  7 Packages

Total download size: 1.1 M
Installed size: 3.7 M
Downloading Packages:
(1/7): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm                       609 kB/s |  19 kB     00:00
(2/7): libunwind-1.4.0-5.amzn2023.0.3.x86_64.rpm                                   1.7 MB/s |  66 kB     00:00
(3/7): gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64.rpm                             6.5 MB/s | 308 kB     00:00
(4/7): nginx-1.28.0-1.amzn2023.0.2.x86_64.rpm                                      1.7 MB/s |  33 kB     00:00
(5/7): nginx-filesystem-1.28.0-1.amzn2023.0.2.noarch.rpm                           510 kB/s | 9.6 kB     00:00
(6/7): nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch.rpm                            932 kB/s |  21 kB     00:00
(7/7): nginx-core-1.28.0-1.amzn2023.0.2.x86_64.rpm                                  16 MB/s | 686 kB     00:00
----------------------------------------------------------------------------------------------------------------
Total                                                                              9.7 MB/s | 1.1 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                         1/1
  Running scriptlet: nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch                                         1/7
  Installing       : nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch                                         1/7
  Installing       : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch                                            2/7
  Installing       : libunwind-1.4.0-5.amzn2023.0.3.x86_64                                                   3/7
  Installing       : gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64                                             4/7
  Installing       : nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64                                               5/7
  Installing       : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                       6/7
  Installing       : nginx-1:1.28.0-1.amzn2023.0.2.x86_64                                                    7/7
  Running scriptlet: nginx-1:1.28.0-1.amzn2023.0.2.x86_64                                                    7/7
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                       1/7
  Verifying        : gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64                                             2/7
  Verifying        : libunwind-1.4.0-5.amzn2023.0.3.x86_64                                                   3/7
  Verifying        : nginx-1:1.28.0-1.amzn2023.0.2.x86_64                                                    4/7
  Verifying        : nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64                                               5/7
  Verifying        : nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch                                         6/7
  Verifying        : nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch                                            7/7

Installed:
  generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch      gperftools-libs-2.9.1-1.amzn2023.0.3.x86_64      libunwind-1.4.0-5.amzn2023.0.3.x86_64      nginx-1:1.28.0-1.amzn2023.0.2.x86_64
  nginx-core-1:1.28.0-1.amzn2023.0.2.x86_64              nginx-filesystem-1:1.28.0-1.amzn2023.0.2.noarch  nginx-mimetypes-2.1.49-3.amzn2023.0.3.noarch

Complete!
[root@ip-172-65-13-201 ~]# |
```

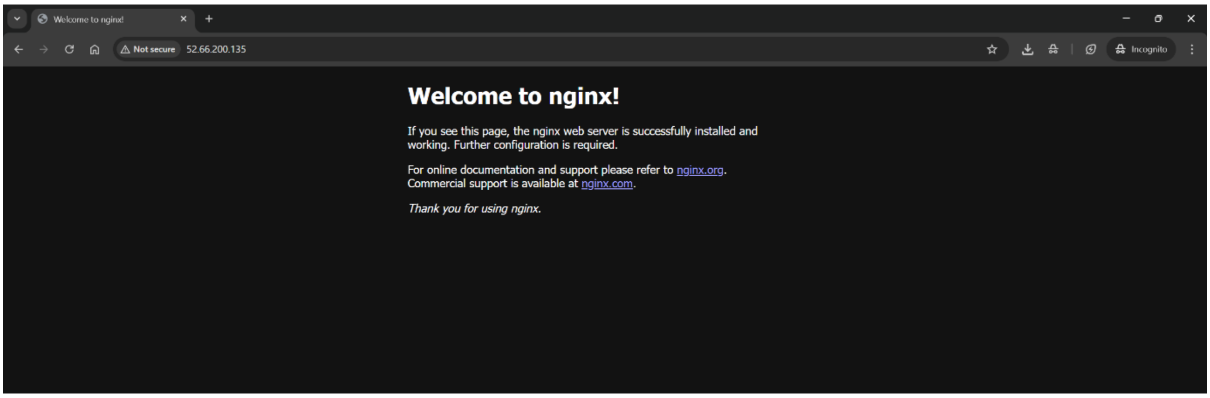
Step 17: Start nginx



```
[ec2-user@ip-172-65-13-201 ~]$ sudo -i
[root@ip-172-65-13-201 ~]# systemctl start nginx.service
[root@ip-172-65-13-201 ~]# systemctl enable nginx.service
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
[root@ip-172-65-13-201 ~]#
```

Step 18: After this, if you hit the public IP Address of the jump server, you see the nginx page.

Title: 3-Tier project (Tomcat & MySQL)



Welcome to nginx!

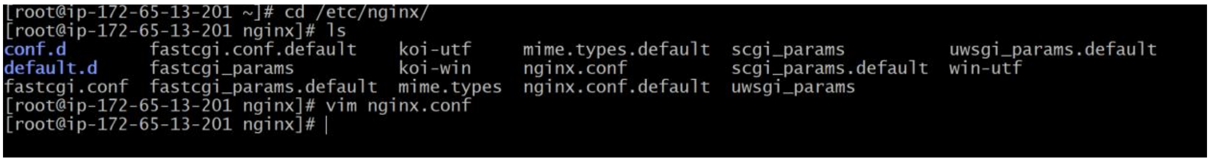
If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

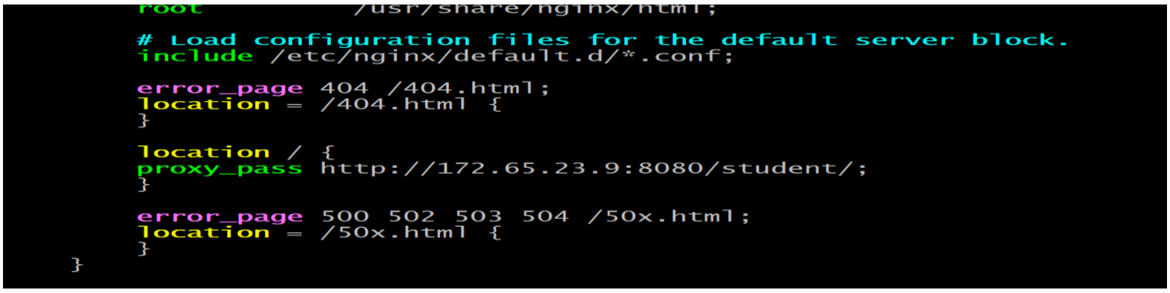
Thank you for using nginx.

So after this, we have to add the location of the application server to see the website.

Step 19: First, log in root of the application server and go to - **/etc/nginx/.** Here, you see nginx.conf, so in this file we have to add the location.



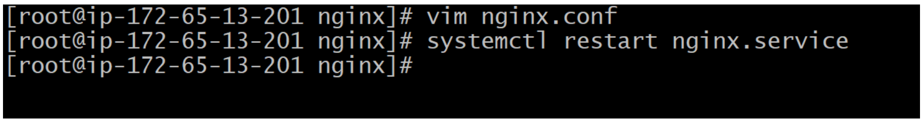
```
[root@ip-172-65-13-201 ~]# cd /etc/nginx/
[root@ip-172-65-13-201 nginx]# ls
conf.d          fastcgi.conf.default    koi-utf     mime.types.default  scgi_params          uwsgi_params.default
default.d       fastcgi_params          koi-win     nginx.conf          scgi_params.default  win-utf
fastcgi.conf    fastcgi_params.default  mime.types  nginx.conf.default  uwsgi_params
[root@ip-172-65-13-201 nginx]# vim nginx.conf
[root@ip-172-65-13-201 nginx]# |
```



```
        root         /usr/share/nginx/html;

        # Load configuration files for the default server block.
        include /etc/nginx/default.d/*.conf;

        error_page 404 /404.html;
        location = /404.html {
        }

        location / {
        proxy_pass http://172.65.23.9:8080/student/;
        }

        error_page 500 502 503 504 /50x.html;
        location = /50x.html {
        }
    }
```

In the location, we need to give the private IP Address of the Application server with the folder name to identify the folder and website.

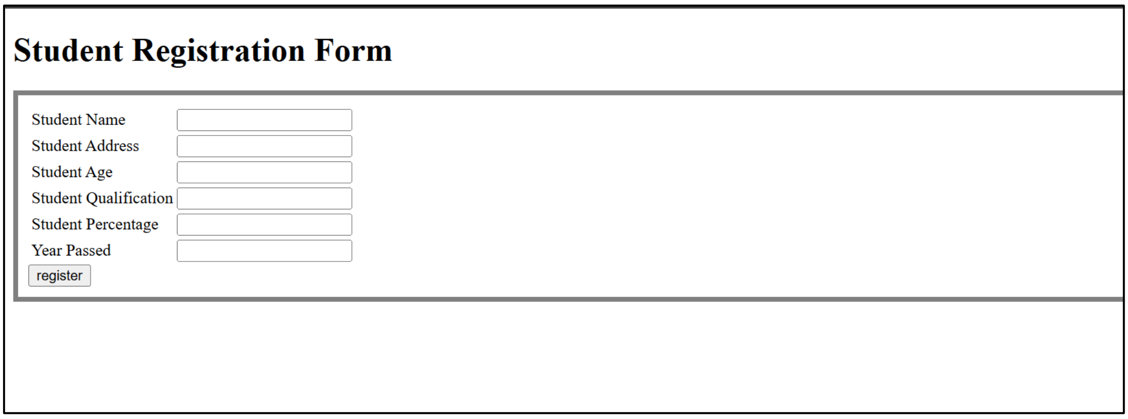
Step 20: After this, restart the nginx.



```
[root@ip-172-65-13-201 nginx]# vim nginx.conf
[root@ip-172-65-13-201 nginx]# systemctl restart nginx.service
[root@ip-172-65-13-201 nginx]#
```

Step 21: Then you can see the website after hitting the public IP address

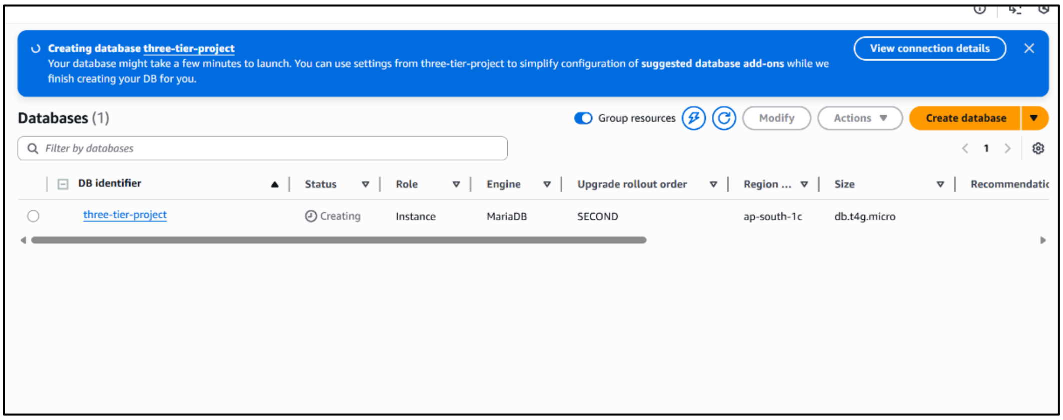
Title: 3-Tier project (Tomcat & MySQL)



Student Registration Form

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

Step 22: So, to store the data, create an RDS Database



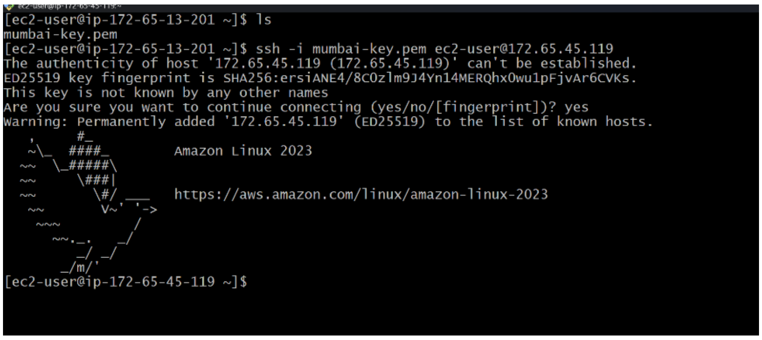
Creating database three-tier-project
Your database might take a few minutes to launch. You can use settings from three-tier-project to simplify configuration of **suggested database add-ons** while we finish creating your DB for you.

Databases (1)

DB identifier	Status	Role	Engine	Upgrade rollout order	Region ...	Size	Recommendatic
three-tier-project	Creating	Instance	MariaDB	SECOND	ap-south-1c	db.t4g.micro	

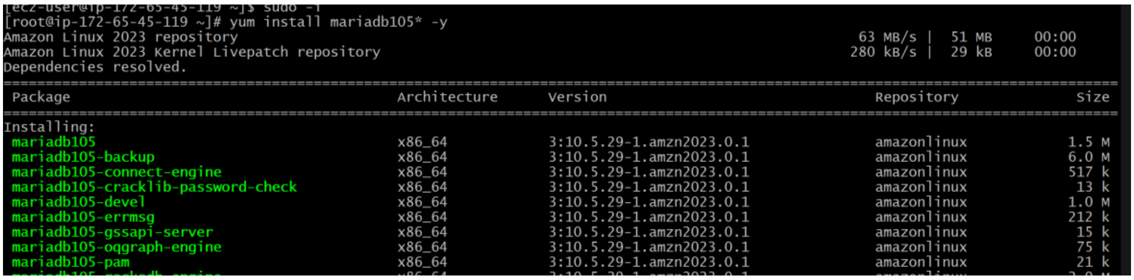
RDS creation select Mariadb – user name and manage password, and also select our VPC and security group. But to select the Availability zone, select the zone where you created db server and create an RDS DB instance there.

Step 23: Get access to the DB instance first with the key and db instance private IP address



```
[ec2-user@ip-172-65-13-201 ~]$ ls
mumbai-key.pem
[ec2-user@ip-172-65-13-201 ~]$ ssh -i mumbai-key.pem ec2-user@172.65.45.119
The authenticity of host '172.65.45.119 (172.65.45.119)' can't be established.
ED25519 key fingerprint is SHA256:ersiANE4/8COzlm9J4Yn14MERQhxOwu1pFjvAr6CVKs.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.65.45.119' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
[ec2-user@ip-172-65-45-119 ~]$
```

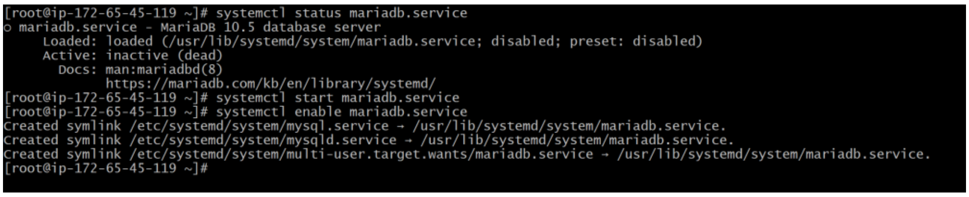
Step 24: To access the RDS Database, we need to install mariadb105*.



```
[ec2-user@ip-172-65-45-119 ~]$ sudo -i
[root@ip-172-65-45-119 ~]# yum install mariadb105* -y
Amazon Linux 2023 repository                                          63 MB/s |  51 MB     00:00
Amazon Linux 2023 Kernel Livepatch repository                        280 kB/s |  29 kB     00:00
Dependencies resolved.
=====================================================================================================
 Package                              Architecture   Version                     Repository      Size
=====================================================================================================
Installing:
 mariadb105                           x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    1.5 M
 mariadb105-backup                    x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    6.0 M
 mariadb105-connect-engine            x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    517 k
 mariadb105-cracklib-password-check   x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux     13 k
 mariadb105-devel                     x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    1.0 M
 mariadb105-errmsg                    x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    212 k
 mariadb105-gssapi-server             x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux     15 k
 mariadb105-oqgraph-engine            x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux     75 k
 mariadb105-pam                       x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux     21 k
 mariadb105-rocksdb-engine            x86_64         3:10.5.29-1.amzn2023.0.1    amazonlinux    2.9 M
```

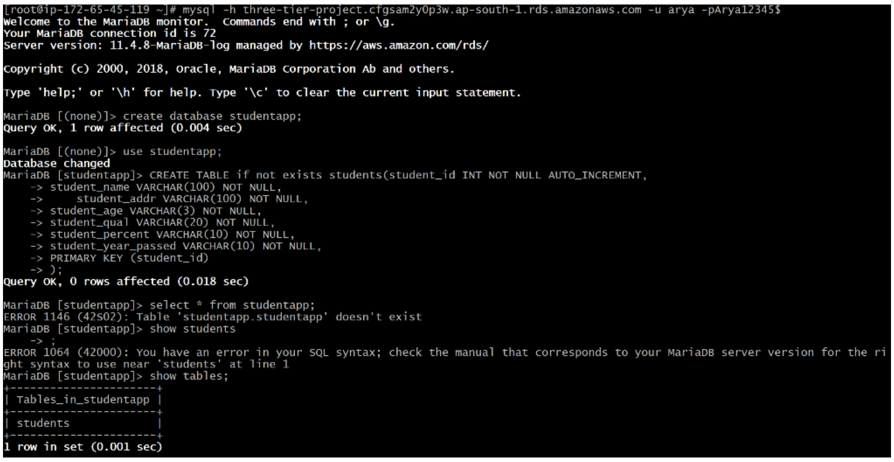
Title: 3-Tier project (Tomcat & MySQL)

Now start it –



```
[root@ip-172-65-45-119 ~]# systemctl status mariadb.service
○ mariadb.service - MariaDB 10.5 database server
     Loaded: loaded (/usr/lib/systemd/system/mariadb.service; disabled; preset: disabled)
     Active: inactive (dead)
       Docs: man:mariadbd(8)
             https://mariadb.com/kb/en/library/systemd/
[root@ip-172-65-45-119 ~]# systemctl start mariadb.service
[root@ip-172-65-45-119 ~]# systemctl enable mariadb.service
Created symlink /etc/systemd/system/mysql.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/mysqld.service → /usr/lib/systemd/system/mariadb.service.
Created symlink /etc/systemd/system/multi-user.target.wants/mariadb.service → /usr/lib/systemd/system/mariadb.service.
[root@ip-172-65-45-119 ~]#
```

Step 24: After the creation of the RDS Database instance, you get an Endpoint with this, and you can access the RDS instance on the DB server. – mysql -h endpoint -u USERNAME -pPASSWORD



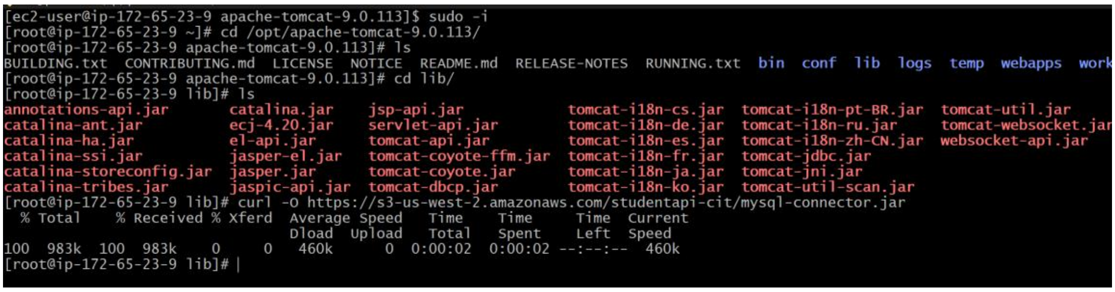
```
[root@ip-172-65-45-119 ~]# mysql -h three-tier-project.cfgsam2y0p3w.ap-south-1.rds.amazonaws.com -u arya -pArya12345$
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 72
Server version: 11.4.8-MariaDB-log managed by https://aws.amazon.com/rds/

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 (0.004 sec)

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 (0.018 sec)

MariaDB [studentapp]> select * from studentapp;
ERROR 1146 (42S02): Table 'studentapp.studentapp' doesn't exist
MariaDB [studentapp]> show students
    -> ;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the ri
ght syntax to use near 'students' at line 1
MariaDB [studentapp]> show tables;
+----------------------+
| Tables_in_studentapp |
+----------------------+
| students             |
+----------------------+
1 row in set (0.001 sec)
```

In this, also configure the Database and table for the website with commands.

Step 25: Now log out of the DB server and log in application server to connect application database.

After logging root of the app, go to - /opt/apache-tomcat-9.0.113/lib/

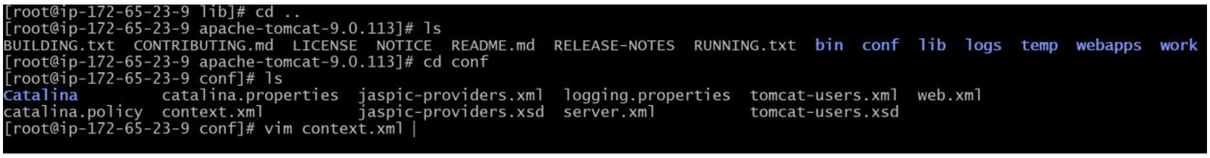
Here, we have to download **mysql connector** to connect to the Database.



```
[ec2-user@ip-172-65-23-9 apache-tomcat-9.0.113]$ sudo -i
[root@ip-172-65-23-9 ~]# cd /opt/apache-tomcat-9.0.113/
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# ls
BUILDING.txt  CONTRIBUTING.md  LICENSE  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  bin  conf  lib  logs  temp  webapps  work
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# cd lib/
[root@ip-172-65-23-9 lib]# ls
annotations-api.jar       catalina.jar    jsp-api.jar            tomcat-i18n-cs.jar  tomcat-i18n-pt-BR.jar  tomcat-util.jar
catalina-ant.jar          ecj-4.20.jar    servlet-api.jar        tomcat-i18n-de.jar  tomcat-i18n-ru.jar     tomcat-websocket.jar
catalina-ha.jar           el-api.jar      tomcat-api.jar         tomcat-i18n-es.jar  tomcat-i18n-zh-CN.jar  websocket-api.jar
catalina-ssi.jar          jasper-el.jar   tomcat-coyote-ffm.jar  tomcat-i18n-fr.jar  tomcat-jdbc.jar
catalina-storeconfig.jar  jasper.jar      tomcat-coyote.jar      tomcat-i18n-ja.jar  tomcat-jni.jar
catalina-tribes.jar       jaspic-api.jar  tomcat-dbcp.jar        tomcat-i18n-ko.jar  tomcat-util-scan.jar
[root@ip-172-65-23-9 lib]# curl -O https://s3-us-west-2.amazonaws.com/studentapi-cit/mysql-connector.jar
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  983k  100  983k    0     0   460k      0  0:00:02  0:00:02 --:--:--  460k
[root@ip-172-65-23-9 lib]# |
```

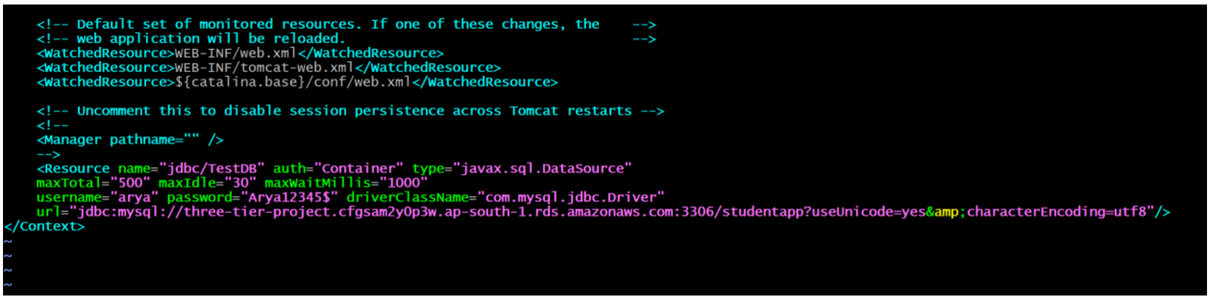
Step 26: Now we have to add the location of db in the application server, so go to - /opt/apache-tomcat-9.0.113/conf/

Here, you see **content.xml,** so in this, we add the location and save it.



```
[root@ip-172-65-23-9 lib]# cd ..
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# ls
BUILDING.txt  CONTRIBUTING.md  LICENSE  NOTICE  README.md  RELEASE-NOTES  RUNNING.txt  bin  conf  lib  logs  temp  webapps  work
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# cd conf
[root@ip-172-65-23-9 conf]# ls
Catalina         catalina.properties  jaspic-providers.xml  logging.properties  tomcat-users.xml  web.xml
catalina.policy  context.xml          jaspic-providers.xsd  server.xml          tomcat-users.xsd
[root@ip-172-65-23-9 conf]# vim context.xml |
```

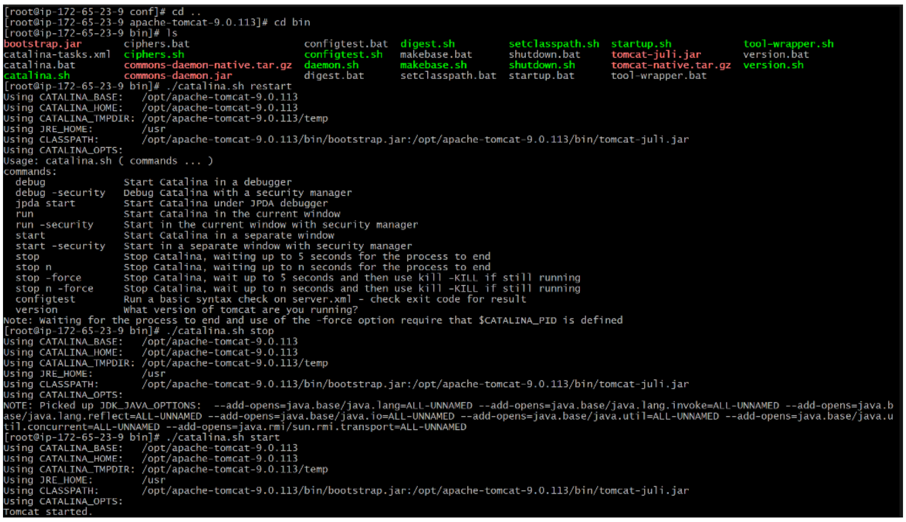
Title: 3-Tier project (Tomcat & MySQL)



```
    <!-- 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>

    <!-- Uncomment this to disable session persistence across Tomcat restarts -->
    <!--
    <Manager pathname="" />
    -->
    <Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource"
    maxTotal="500" maxIdle="30" maxWaitMillis="1000"
    username="arya" password="Arya12345$" driverClassName="com.mysql.jdbc.Driver"
    url="jdbc:mysql://three-tier-project.cfgsam2y0p3w.ap-south-1.rds.amazonaws.com:3306/studentapp?useUnicode=yes&amp;characterEncoding=utf8"/>
</Context>
~
~
~
~
```

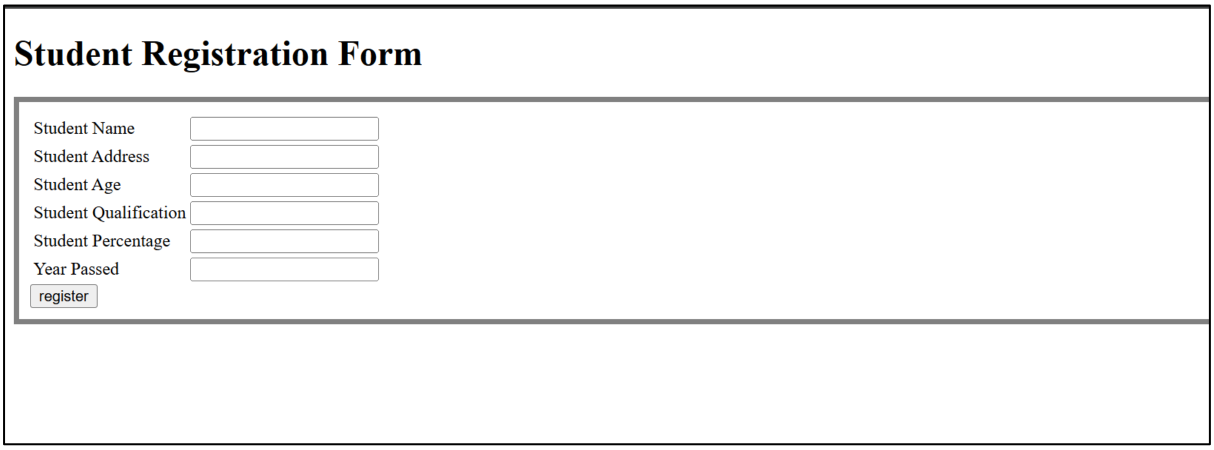
In this, we give the username and password of the database and also the endpoint of the RDS Instance.

Step 27: Then stop the Tomcat and start it again.



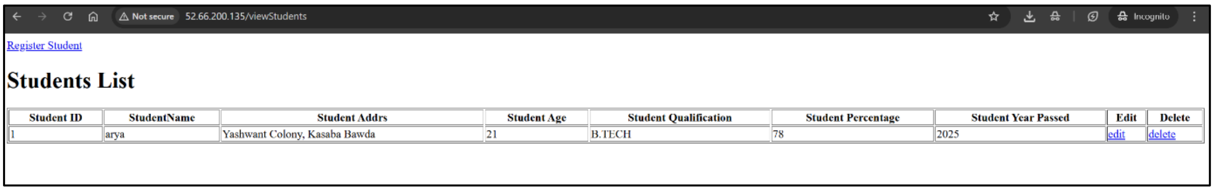
```
[root@ip-172-65-23-9 conf]# cd ..
[root@ip-172-65-23-9 apache-tomcat-9.0.113]# cd bin
[root@ip-172-65-23-9 bin]# ls
bootstrap.jar        ciphers.bat                   configtest.bat  digest.sh          setclasspath.sh  startup.sh              tool-wrapper.sh
catalina-tasks.xml   ciphers.sh                    configtest.sh   makebase.bat       shutdown.bat     tomcat-juli.jar         version.bat
catalina.bat         commons-daemon-native.tar.gz  daemon.sh       makebase.sh        shutdown.sh      tomcat-native.tar.gz    version.sh
catalina.sh          commons-daemon.jar            digest.bat      setclasspath.bat   startup.bat      tool-wrapper.bat
[root@ip-172-65-23-9 bin]# ./catalina.sh restart
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.113
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.113
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.113/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.113/bin/bootstrap.jar:/opt/apache-tomcat-9.0.113/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Usage: catalina.sh ( commands ... )
commands:
  debug             Start Catalina in a debugger
  debug -security   Debug Catalina with a security manager
  jpda start        Start Catalina under JPDA debugger
  run               Start Catalina in the current window
  run -security     Start in the current window with security manager
  start             Start Catalina in a separate window
  start -security   Start in a separate window with security manager
  stop              Stop Catalina, waiting up to 5 seconds for the process to end
  stop n            Stop Catalina, waiting up to n seconds for the process to end
  stop -force       Stop Catalina, wait up to 5 seconds and then use kill -KILL if still running
  stop n -force     Stop Catalina, wait up to n seconds and then use kill -KILL if still running
  configtest        Run a basic syntax check on server.xml - check exit code for result
  version           What version of tomcat are you running?
Note: Waiting for the process to end and use of the -force option require that $CATALINA_PID is defined
[root@ip-172-65-23-9 bin]# ./catalina.sh stop
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.113
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.113
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.113/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.113/bin/bootstrap.jar:/opt/apache-tomcat-9.0.113/bin/tomcat-juli.jar
Using CATALINA_OPTS:
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --add-opens=java.b
ase/java.lang.reflect=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.base/java.u
til.concurrent=ALL-UNNAMED --add-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
[root@ip-172-65-23-9 bin]# ./catalina.sh start
Using CATALINA_BASE:   /opt/apache-tomcat-9.0.113
Using CATALINA_HOME:   /opt/apache-tomcat-9.0.113
Using CATALINA_TMPDIR: /opt/apache-tomcat-9.0.113/temp
Using JRE_HOME:        /usr
Using CLASSPATH:       /opt/apache-tomcat-9.0.113/bin/bootstrap.jar:/opt/apache-tomcat-9.0.113/bin/tomcat-juli.jar
Using CATALINA_OPTS:
Tomcat started.
```

After all these steps, we create a 3-Tier project.



Student Registration Form

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



Register Student

Students List

Student ID	StudentName	Student Addrs	Student Age	Student Qualification	Student Percentage	Student Year Passed	Edit	Delete
1	arya	Yashwant Colony, Kasaba Bawda	21	B.TECH	78	2025	edit	delete

Here you can see where all the data will be stored.