

Deploying 3 Tier Application

EC2-Github-RDS_MariaDB

Goal:

- A **frontend** (user interface)
- A **backend** (API/server logic)
- A **database** (MariaDB)

We are using:

- **EC2 (AWS)** – to *host app code* (frontend + backend)
- **GitHub** – to *store and version code*
- **RDS MariaDB (AWS)** – managed database (no need to install MariaDB on EC2)

First we will create database MariaDB using RDS service

The screenshot shows the AWS Aurora and RDS Databases page. On the left, there's a sidebar with options like Dashboard, Databases, Query editor, and Performance insights. The main area is titled 'Databases (0)' and has a 'Create database' button at the top right. A red box highlights this button. Below it, there's a search bar and a table header with columns for DB identifier, Status, Role, Engine, Region, and Size. A cartoon robot icon is in the center, and the message 'No resources' is displayed.

The screenshot shows the 'Create database' configuration page. It has two tabs: 'Standard create' (selected) and 'Easy create'. The 'Standard create' tab includes a note: 'You set all of the configuration options, including ones for availability, security, backups, and maintenance.' The 'Easy create' tab includes a note: 'Use recommended best-practice configurations. Some configuration options can be changed after the database is created.' Below these, there's a section for 'Engine options' with several engine type buttons. The 'MariaDB' button is selected and highlighted with a blue border. Other engines shown include Aurora (MySQL Compatible), Aurora (PostgreSQL Compatible), MySQL, PostgreSQL, Oracle, Microsoft SQL Server, and IBM Db2. Each engine has its respective logo next to it.

AWS | Search [Alt+S] United States (N. Virginia) Ganraj

EC2 IAM VPC Billing and Cost Manage... Aurora and RDS S3

Aurora and RDS > Create database

▼ Hide filters

Show only 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
MariaDB 11.4.5

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](#)

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.
database-1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

AWS | Search [Alt+S] United States (N. Virginia) Ganraj

EC2 IAM VPC Billing and Cost Manage... Aurora and RDS S3

Aurora and RDS > Create database

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.
mariadb

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 63 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ 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.

Credentials management
You can use AWS Secrets Manager or manage your master user credentials.

Managed in AWS Secrets Manager - most secure
RDS generates a password for you and manages it throughout its lifecycle using AWS Secrets Manager.

Self managed
Create your own password or have RDS create a password that you manage.

Auto generate password
Amazon RDS can generate a password for you, or you can specify your own password.

Set password whatever you want

AWS | Search [Alt+S] United States (N. Virginia) Ganraj

EC2 IAM VPC Billing and Cost Manage... Aurora and RDS S3

Aurora and RDS > Create database

Memory optimized classes (includes r and x classes)

Burstable classes (includes t classes)

db.t3.micro
2 vCPUs 1 GiB RAM Network: Up to 2,085 Mbps

Storage

Storage type [Info](#)
Provisioned IOPS SSD (io2) storage volumes are now available.
General Purpose SSD (gp2)
Baseline performance determined by volume size

Allocated storage [Info](#)
20 GiB

Allocated storage value must be 20 GiB to 6,144 GiB

► Additional storage configuration

Availability & durability

The screenshot shows the 'Create database' wizard in the AWS Management Console. The 'Compute resource' section is highlighted with a red box around the 'Connect to an EC2 compute resource' option, which is selected. Below it, the 'Network type' section shows 'IPv4' selected. The 'Virtual private cloud (VPC)' section lists 'Default VPC (vpc-0dbbb05820b87165c)' with 6 Subnets and 6 Availability Zones. A note at the bottom says 'After a database is created, you can't change its VPC.'

Here you can we have selected don't connect to Ec2 you choose connect to Ec2 compute If you already created Ec2 and VPC also your own created or Default one. You can decide your database is need to public or private access.

The screenshot shows the 'Create database' wizard with several sections highlighted with red boxes. The 'Virtual private cloud (VPC)' section shows 'Default VPC (vpc-0dbbb05820b87165c)' selected. The 'DB subnet group' section shows 'default-vpc-0dbbb05820b87165c' selected. The 'Public access' section shows 'Yes' selected. The 'VPC security group (firewall)' section shows 'One or more VPC security groups' selected.

Security Groups need to select which enabled with specific ports to manage security over it and you can select availability zones also .

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](#)

Existing VPC security groups
[Choose one or more options](#)

[default](#) [FortLinux](#)

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](#).

Certificate authority - optional [Info](#)
 Using a server certificate provides an extra layer of security by validating that the connection is being made to an Amazon database. It does so by checking the server certificate that is automatically installed on all databases that you provision.

rds-ca-rsa2048-g1 (default)

Then just click create database. Once its shows active status you can use it

Aurora and RDS > Databases

Databases (1)

Creating database mariadb
 Your database might take a few minutes to launch. You can use settings from mariadb to simplify configuration of [suggested database add-ons](#) while we finish creating your DB for you.

DB identifier	Status	Role	Engine	Region ...	Size
mariadb	Creating	Instance	MariaDB	us-east-1f	db.t3.micro

Aurora and RDS > Databases

Databases (1)

Successfully created database mariadb
 You can use settings from mariadb to simplify configuration of [suggested database add-ons](#) while we finish creating your DB for you.

DB identifier	Status	Role	Engine	Region ...	Size
mariadb	Available	Instance	MariaDB	us-east-1f	db.t3.micro

Now we will create EC2 for our project

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Application and OS Images (Amazon Machine Image)'. A search bar at the top right contains the text 'Webserver'. Below it, a section titled 'Recent' lists various AMI categories: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux, and Debian. An 'ubuntu' icon is highlighted. To the right, there's a link to 'Browse more AMIs'. On the far right, there are 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', 'Storage (volumes)', and a large orange 'Launch instance' button.

IMP :-Here you can select whatever about instance type .but I have used t2 because of free tier. Recommended is higher it will use high usage because of dependencies.

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Instance type'. It shows the selected instance type as 't2.micro'. Below it, there's a detailed description of its specifications: Family: t2, 1 vCPU, 1 GiB Memory, Current generation: true, On-Demand Windows base pricing: 0.0162 USD per Hour, On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour, On-Demand SUSE base pricing: 0.0116 USD per Hour, On-Demand RHEL base pricing: 0.026 USD per Hour, and On-Demand Linux base pricing: 0.0116 USD per Hour. A note below states 'Additional costs apply for AMIs with pre-installed software'. To the right, there are sections for 'Summary', 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', 'Storage (volumes)', and a large orange 'Launch instance' button.

Twist is here you need select security group which has allowed proper ports like [ssh,http,https,tcp].

The screenshot shows the 'Launch an instance' wizard in the AWS Management Console. The current step is 'Configure storage'. It shows a single volume of 8 GiB. To the right, there are sections for 'Summary', 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', 'Storage (volumes)', and a large orange 'Launch instance' button.

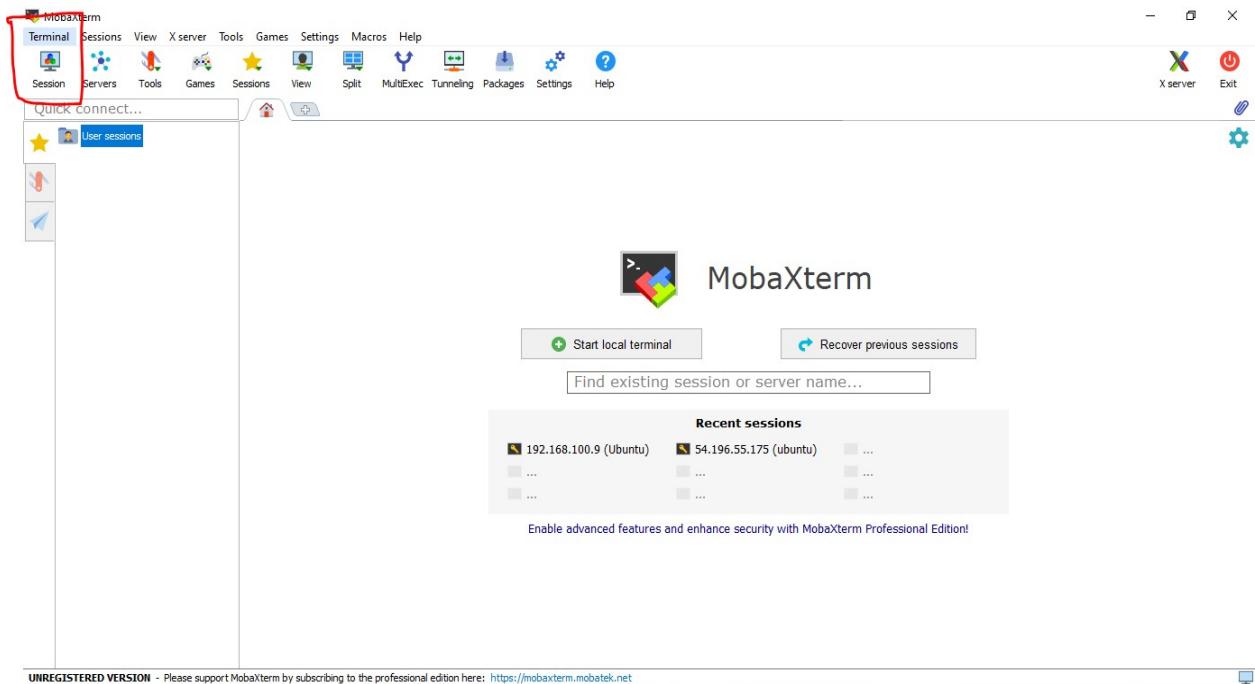
The screenshot shows the AWS Management Console with the EC2 service selected. The main pane displays a table of instances. There is one instance listed: 'Webserver' (Instance ID: i-04815e6815691337a), which is 'Running'. The instance type is 't2.micro', and it is in the 'Initializing' status check. It is located in the 'us-east-1b' availability zone and has a public IPv4 address 'ec2-54-158'. The left sidebar shows the navigation path 'EC2 > Instances' and lists various EC2-related options like Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, and Savings Plans.

See you can directly use EC2 for ubuntu os and other process. But we need two more terminals for this project to work so that why we are using MobaXterm for that process , So we will use SSH key to access our EC2 to MobaXterm. See you have already downloaded your SSH key When you are created EC2 or you have to create new SSH key go create one .

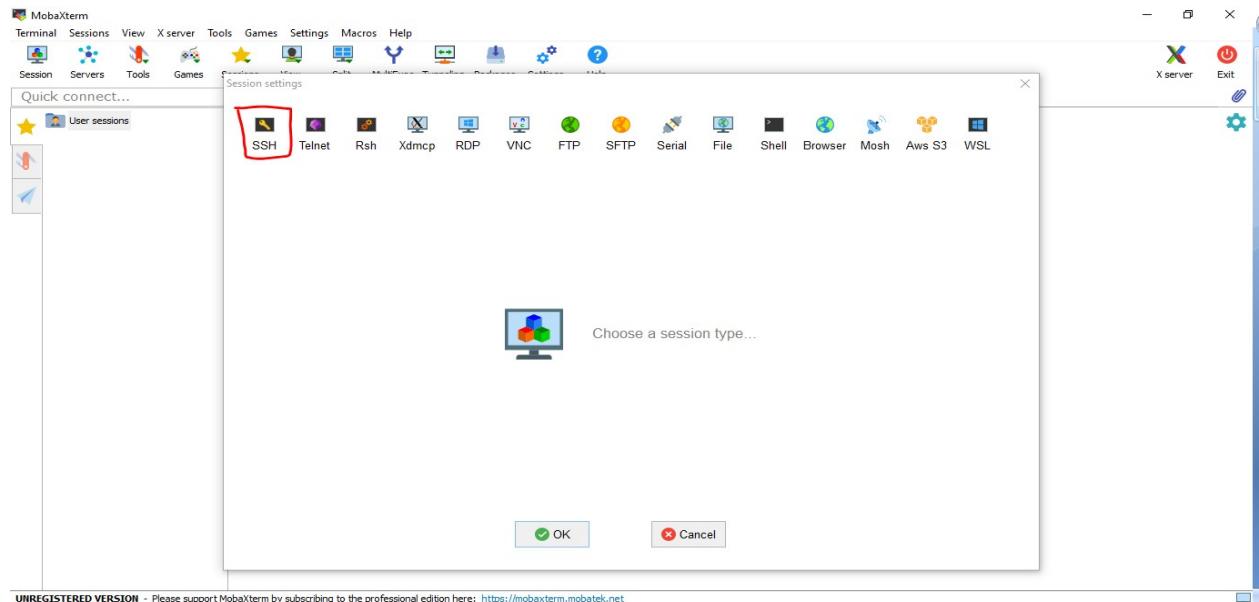
First Download Mobaxterm from Internet (web) from its Official Site

<https://mobaxterm.mobatek.net/>

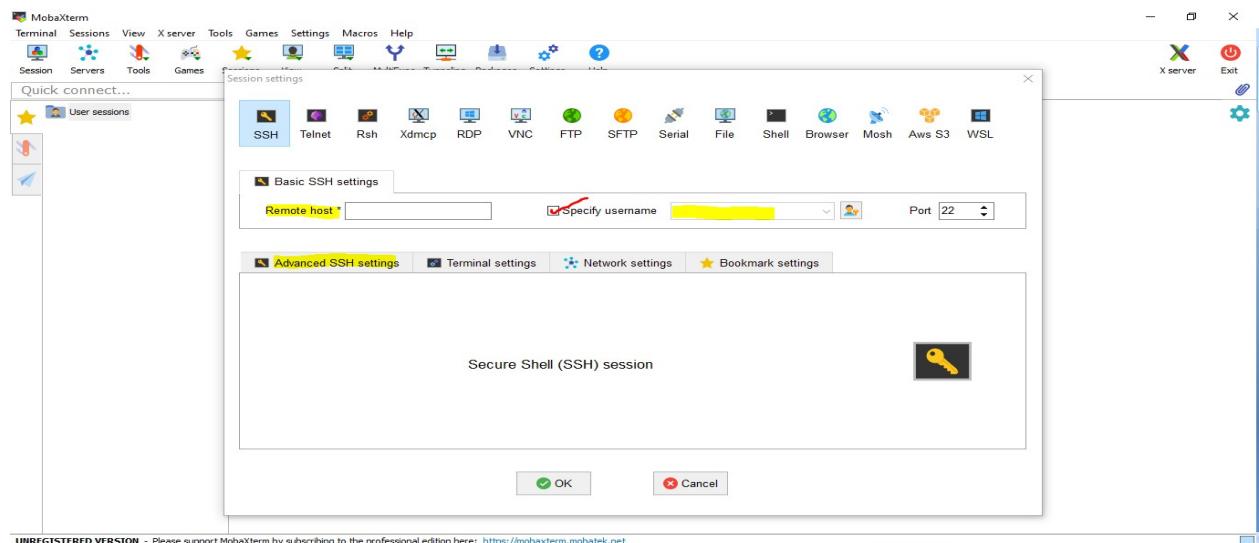
After installation open MobaXterm



Go to terminals select SSH option

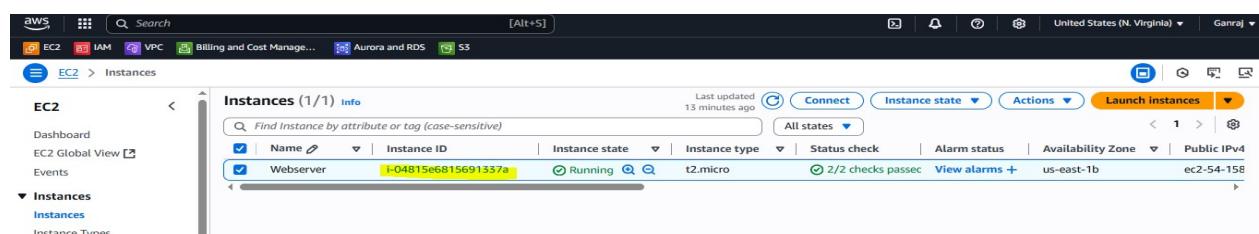


On the below you see marked and highlighted parts need to fill-up. I will show one by one



Remote Host – your Ec2 instance public IP to connect or Private if given access to it

You can find your instance IP just click on ec2 instance ID



Instance summary for i-04815e6815691337a (Webserver) [Info](#)

Instance ID: i-04815e6815691337a

IPv6 address: -

Hostname type: IP name: ip-172-31-91-183.ec2.internal

Answer private resource DNS name: IPv4 (A)

Auto-assigned IP address: 54.158.83.135 [Public IP]

IAM Role: -

IMDSv2: Required

Public IP4 address: [\\$4.158.83.135 | open address](#)

Instance state: Running

Private IP DNS name (IPv4 only): ip-172-31-91-183.ec2.internal

Instance type: t2.micro

VPC ID: vpc-0dbb05820b87165c

Subnet ID: subnet-0b3b28230b801c487

Instance ARN: arn:aws:ec2:us-east-1:351552951218:instance/i-04815e6815691337a

Public DNS Name: ec2-54-158-83-135.compute-1.amazonaws.com

Elastic IP addresses: -

AWS Compute Optimizer finding: -

Auto Scaling Group name: -

Managed: false

Specify username : there lots name mentioned for linux os distributions

Environment	Username
Ubuntu Server VM / cloud	ubuntu
Amazon Linux	ec2-user
CentOS cloud image	centos
WSL Ubuntu	Your chosen name
Kali Linux (recent)	kali
Alpine Linux	root

SSH Telnet Rsh Xdmcp RDP VNC FTP SFTP Serial File Shell Browser Mosh Aws S3 WSL

Basic SSH settings

Remote host * [54.158.83.135](#) Specify username [ubuntu](#) Port [22](#)

Advanced SSH settings

X11-Forwarding Compression Remote environment: Interactive shell

Execute command: Do not exit after command ends

SSH-browser type: SFTP protocol Follow SSH path (experimental)

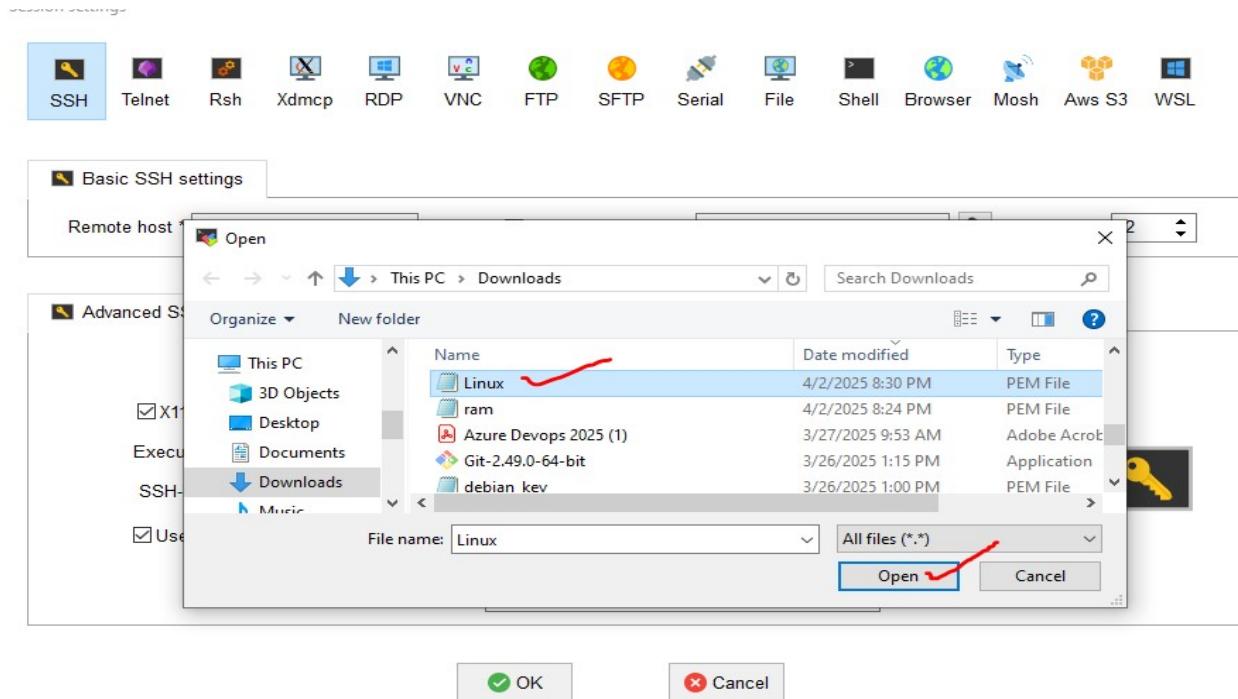
Use private key

Expert SSH settings

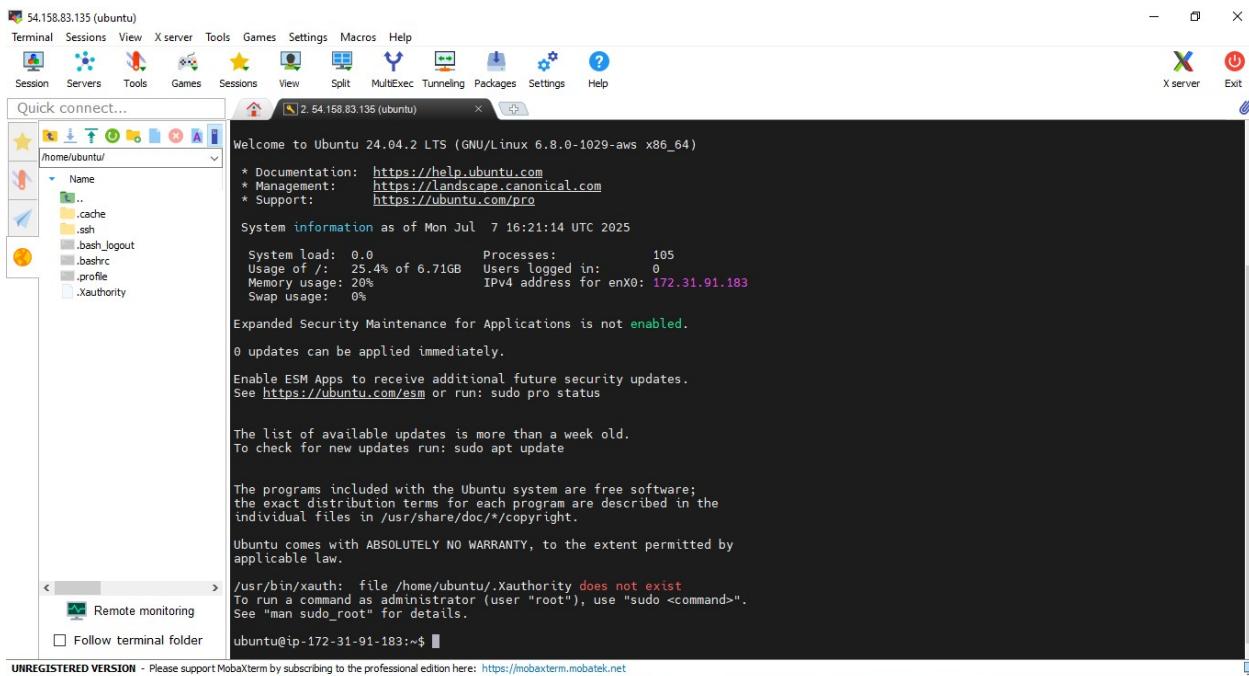
Execute macro at session start: <none>

OK Cancel

You have click check box use private key add it from where you saved in a .pem format and click OK.



Here you can see we have successfully connected to our EC2 instance using ssh key to Mobaxterm.



First check security group opened ports

The screenshot shows the AWS Management Console with the URL [https://console.aws.amazon.com/ec2/v2/home?#SecurityGroups:group/sg-01e3b75e9bb582d6d](#). The left sidebar is collapsed. The main content area displays the 'Inbound rules' tab for a security group named 'sg-01e3b75e9bb582d6d - ForLinux'. The table lists nine inbound rules:

Name	Security group rule ID	IP version	Type	Protocol	Port range
-	sgr-058577c933a8adc42	IPv4	MySQL/Aurora	TCP	3306
-	sgr-0a5c548336190bf4c	IPv4	All TCP	TCP	0 - 65535
-	sgr-0bd6507b3d095d2	IPv4	HTTPS	TCP	443
-	sgr-0c9a86abd1cf04e84	IPv4	HTTP	TCP	80
-	sgr-0ada94f7740605028	IPv4	Custom TCP	TCP	8080
-	sgr-02cedb4e09f45f765	IPv4	SSH	TCP	22
-	sgr-037a11ee25071bdb6	IPv4	NFS	TCP	2049
-	sgr-0c0475b85b9d9565f	IPv4	All traffic	All	All
-	sgr-032646cab41a11a6	IPv4	Custom TCP	TCP	27017

Now we need install MariaDB dependencies or you can say installation on our Ec2 instance Ubuntu.

Note : - we take reference of two repositories one is EasyCurd where installation part mentioned about MAriaDB and the main one [cloudblitz-student-app](#)

I have added how install MariaDB on github repo .md file you can copy it from there

<https://github.com/ganrajdol99215/EasyCRUD/blob/main/README.md>

The screenshot shows a GitHub repository page for 'EasyCRUD' at <https://github.com/ganrajdol99215/EasyCRUD>. The page displays the 'README' file content:

MariaDB Setup and Configuration Guide for Windows

This guide explains how to set up MariaDB, create a database, and Create Database User

1. Installing MariaDB

Installing MariaDB on Ubuntu

```
apt update && apt install mariadb-server -y
```

2. Securing MariaDB

Open the Command Prompt as Administrator and run the following command to secure your installation:

```
mysql_secure_installation
```

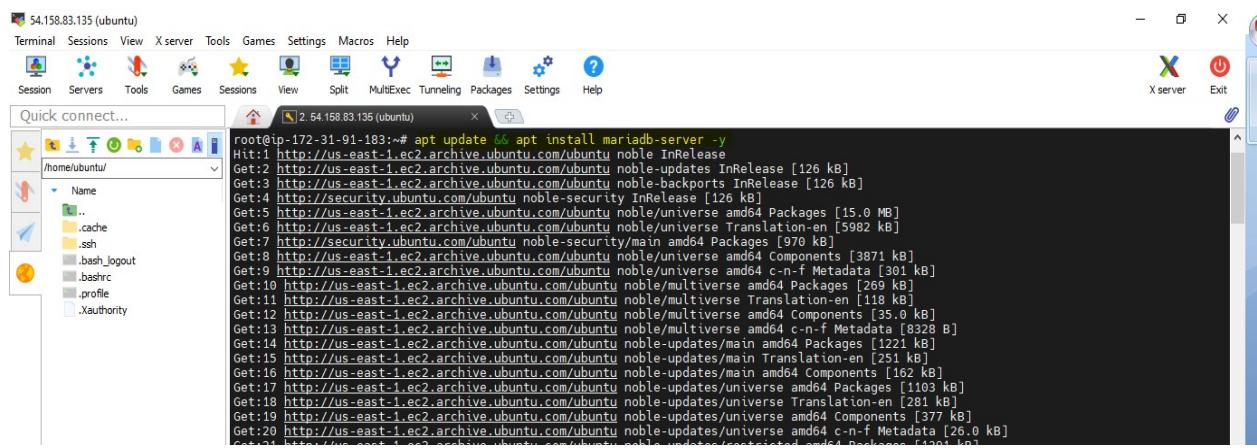
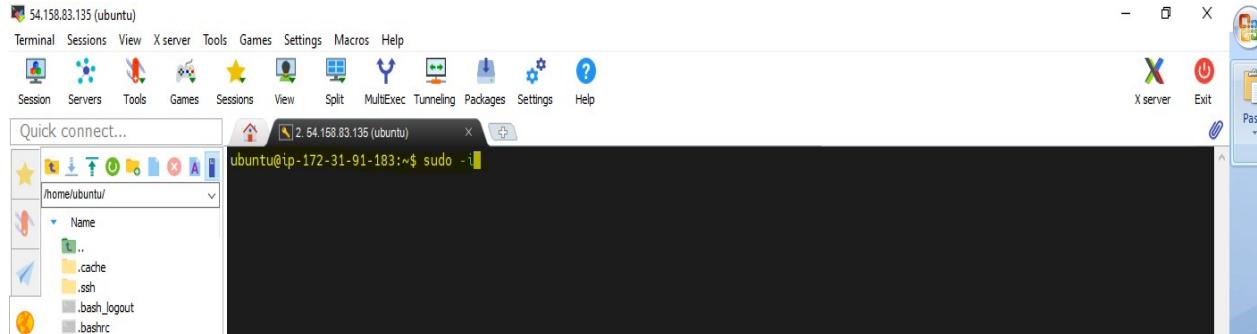
Follow the prompts to: Set a root password. Remove insecure default users and test databases. Disable remote root login.

On the right side of the page, there are sections for Releases, Packages, Languages, and Suggested workflows. The Languages section shows a bar chart with the following data:

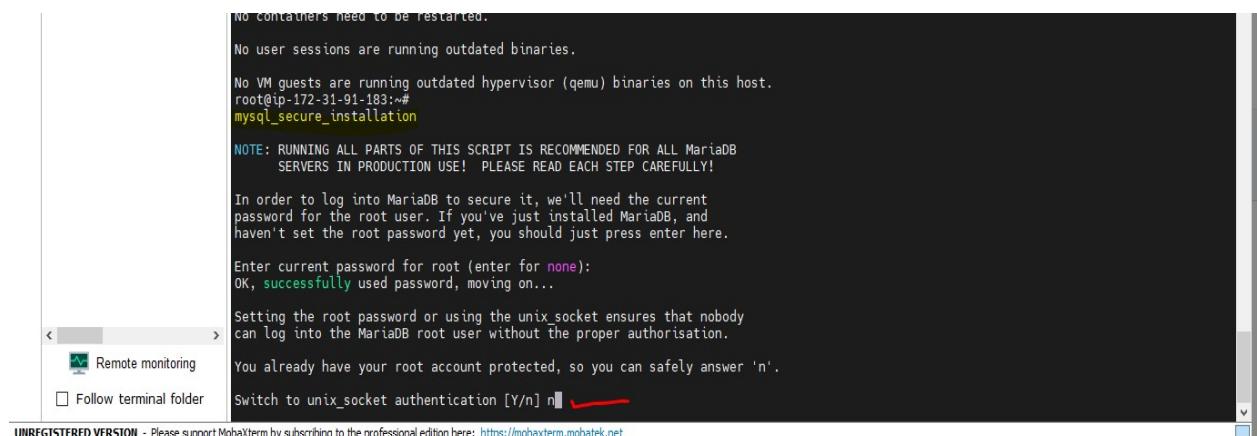
Language	Percentage
JavaScript	56.6%
Java	21.1%
CSS	20.0%
HTML	2.3%

The Suggested workflows section includes options for 'Publish Node.js Package' and 'SLSA Generic generator'.

First make you sudo user or root user mean you can run this installation without any restrictions



Now we need one more command to process config



Read before press Y or n

```

haven't set the root password yet, you should just press enter here.
Enter current password for root (enter for none):
OK, successfully used password, moving on...
Setting the root password or using the unix_socket ensures that nobody
can log into the MariaDB root user without the proper authorisation.
You already have your root account protected, so you can safely answer 'n'.
Switch to unix_socket authentication [Y/n] n
... skipping.
You already have your root account protected, so you can safely answer 'n'.
Change the root password? [Y/n] n
... skipping.
By default, a MariaDB installation has an anonymous user, allowing anyone
to log in to MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.
Remove anonymous users? [Y/n] n
... skipping.
Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.
Disallow root login remotely? [Y/n] n
... skipping.
By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.
Remove test database and access to it? [Y/n] n

```

UNREGISTERED VERSION - Please support Mobaxterm by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

MariaDB installation done in or Vm
We need to use following command

```
mysql -u root -p
mysql -u admin -p***** -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
```

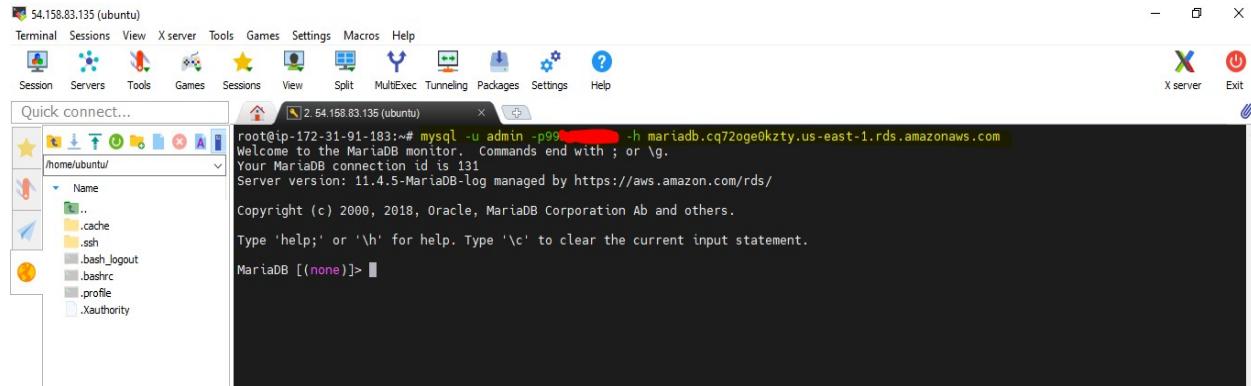
Explanation

Part	Meaning
mysql	The MySQL client command-line tool.
-u admin	Use the username admin to log in.
-p*****	The password (the asterisks here are just placeholders for security).
-h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com	The host (server) you are connecting to—in this case, an Amazon RDS instance running MariaDB.

Now we have to connect to our RDS MariaDB Database for that go to AWS RDS Database
MariaDB Copy Endpoint.

Connectivity & security		
Endpoint mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com	Networking Availability Zone: us-east-1f VPC: vpc-0dbbb05820b87165c Subnet group: default-vpc-0dbbb05820b87165c Subnets: subnet-0203806e5692cba, subnet-0b3b28230b801c487, subnet-04829216fe64f61b6, subnet-0331ba118986b9a4f, subnet-09e0679ef0ffea96, subnet-0f976eab839104d9e	Security VPC security groups: ForLinux (sg-01e5b75e9bb582d6d), Active: default (sg-0438e8ee0b74e6cc0), Active: rds-ca-rsa2048-g1 Publicly accessible: Yes Certificate authority: Info: rds-ca-rsa2048-g1 Certificate authority date: May 26, 2061, 05:04 (UTC+05:30) DB instance certificate expiration date: July 07, 2026, 20:57 (UTC+05:30)

Here you can see we are in the MariaDB database now we have to create one database and Tables for storing our data .



```
root@ip-172-31-91-183:~# mysql -u admin -p99... -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 131
Server version: 11.4.5-MariaDB-log managed by https://aws.amazon.com/rds/

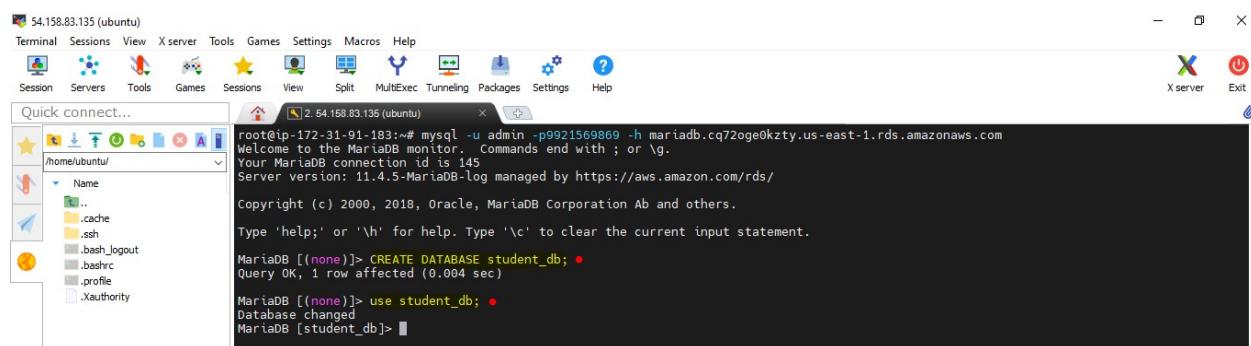
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> 
```

You can see readme.md file for details

<https://github.com/ganrajdol99215/cloudblitz-student-app/blob/main/Readme.md>



```
root@ip-172-31-91-183:~# mysql -u admin -p9921569869 -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 145
Server version: 11.4.5-MariaDB-log managed by https://aws.amazon.com/rds/

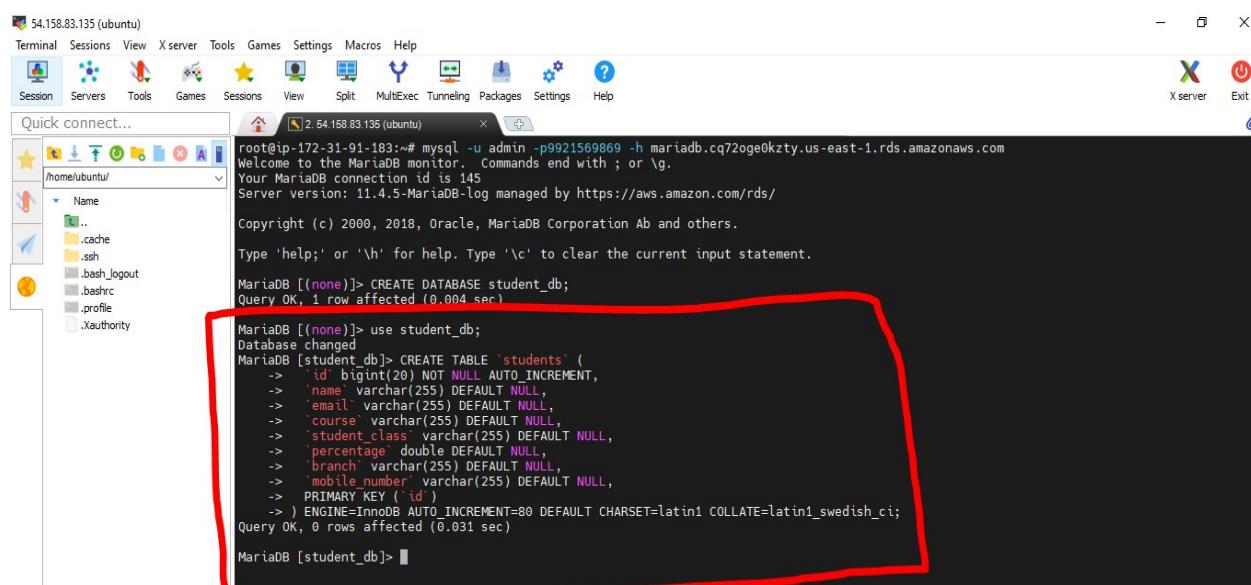
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

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

MariaDB [(none)]> CREATE DATABASE student_db;
Query OK, 1 row affected (0.004 sec)

MariaDB [(none)]> use student_db;
Database changed
MariaDB [student_db]> 
```

Now we will create table



```
root@ip-172-31-91-183:~# mysql -u admin -p9921569869 -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 145
Server version: 11.4.5-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 student_db;
Query OK, 1 row affected (0.004 sec)

MariaDB [(none)]> use student_db;
Database changed
MariaDB [student_db]> CREATE TABLE `students` (
    -> `id` bigint(20) NOT NULL AUTO_INCREMENT,
    -> `name` varchar(255) DEFAULT NULL,
    -> `email` varchar(255) DEFAULT NULL,
    -> `course` varchar(255) DEFAULT NULL,
    -> `student_class` varchar(255) DEFAULT NULL,
    -> `percentage` double DEFAULT NULL,
    -> `branch` varchar(255) DEFAULT NULL,
    -> `mobile_number` varchar(255) DEFAULT NULL,
    -> PRIMARY KEY (`id`),
    -> ) ENGINE=InnoDB AUTO_INCREMENT=80 DEFAULT CHARSET=latin1 COLLATE=latin1_swedish_ci;
Query OK, 0 rows affected (0.031 sec)

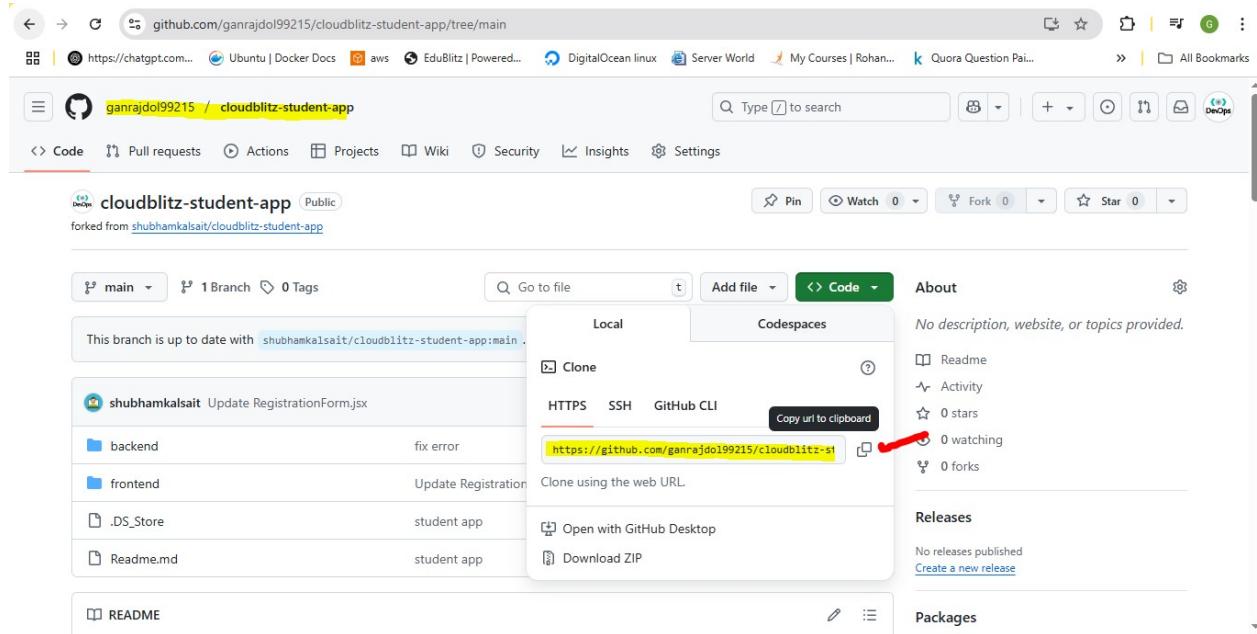
MariaDB [student_db]> 
```

See we created database also table to store specific keyword data of our project application

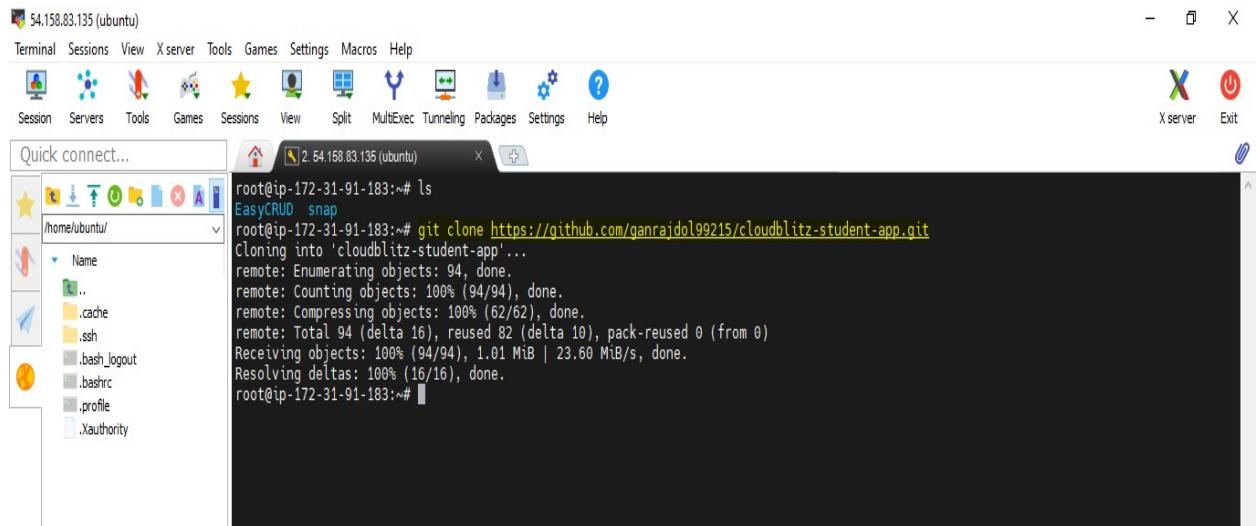
Now moving to backend process

For that we need our github repo – and we will clone it to our VM

<https://github.com/ganrajdol99215/cloudblitz-student-app/tree/main>



The screenshot shows a web browser window with the GitHub repository page for 'cloudblitz-student-app'. The 'Code' tab is active. A tooltip for the 'Clone' button displays the URL 'https://github.com/ganrajdol99215/cloudblitz-student-app.git'. A red arrow points to the 'Copy url to clipboard' button.

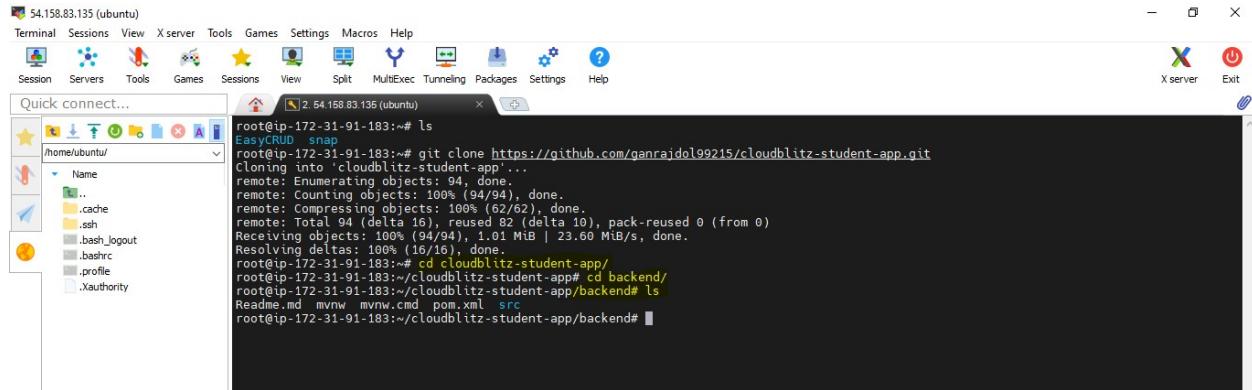


The screenshot shows the Xfce desktop environment on a VM. A terminal window titled '2.54.158.83.135 (ubuntu)' displays the output of the 'git clone' command:

```
root@ip-172-31-91-183:~# ls
EasyCRUD snap
root@ip-172-31-91-183:~# git clone https://github.com/ganrajdol99215/cloudblitz-student-app.git
Cloning into 'cloudblitz-student-app'...
remote: Enumerating objects: 94, done.
remote: Counting objects: 100% (94/94), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 94 (delta 16), reused 82 (delta 10), pack-reused 0 (from 0)
Receiving objects: 100% (94/94), 1.01 MiB | 23.60 MiB/s, done.
Resolving deltas: 100% (16/16), done.
root@ip-172-31-91-183:~#
```

We working on our Backend process

Our Backend is on JAVA

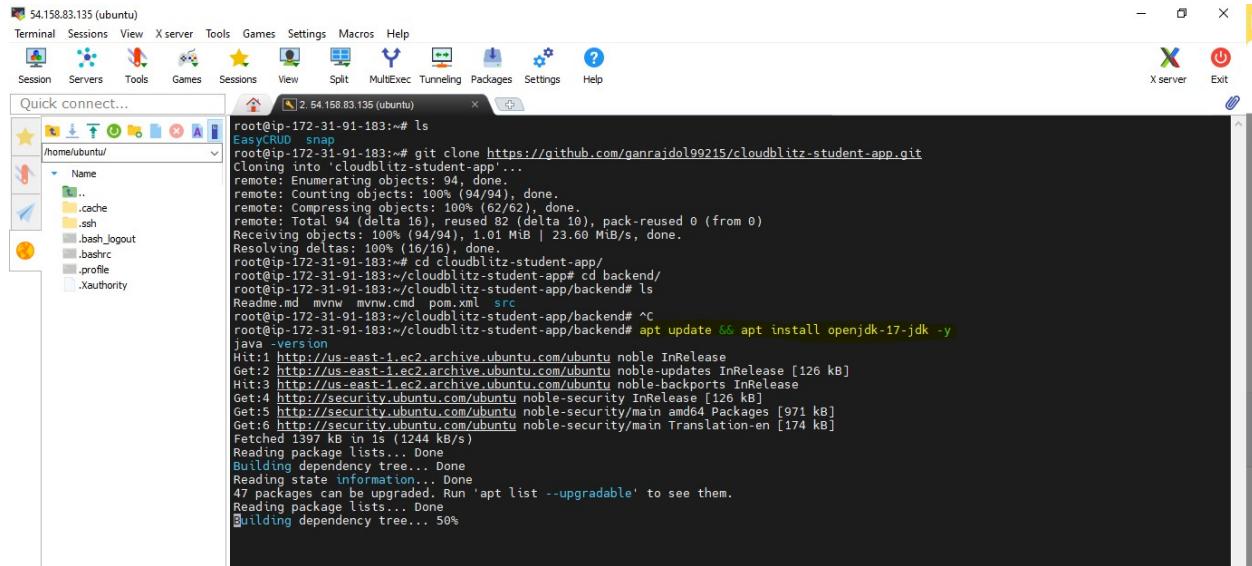


```
root@ip-172-31-91-183:~# git clone https://github.com/ganrajdol99215/cloudblitz-student-app.git
Cloning into 'cloudblitz-student-app'...
remote: Enumerating objects: 94, done.
remote: Counting objects: 100% (94/94), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 94 (delta 16), reused 82 (delta 10), pack-reused 0 (from 0)
Receiving objects: 100% (94/94), 1.01 MiB | 23.60 MiB/s, done.
Resolving deltas: 100% (16/16), done.
root@ip-172-31-91-183:~# cd cloudblitz-student-app/
root@ip-172-31-91-183:~/cloudblitz-student-app# cd backend/
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ls
Readme.md mvnw mvnw.cmd pom.xml src
root@ip-172-31-91-183:~/cloudblitz-student-app/backend#
```

Now we are in the /cloudblitz-student-app/backend# we have install dependencies that need for project like java's , maven , jdk ,npm etc.

here you can find all dependencies need for backend and as I already said we are referencing two git repositories **Easycrud** and **cloudblitz-student-app**

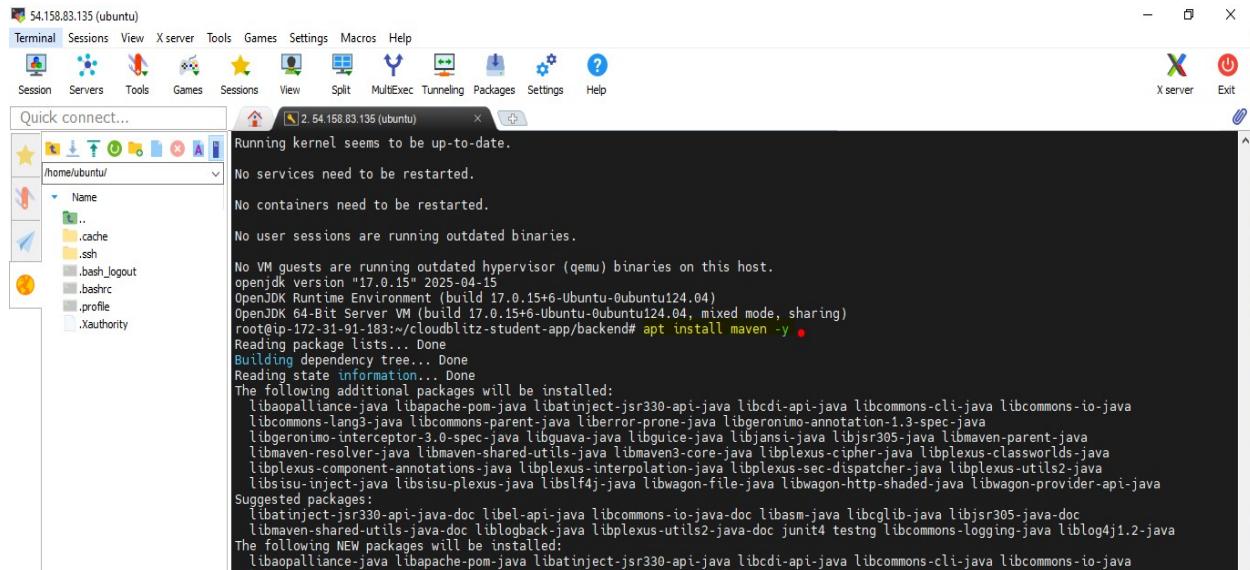
<https://github.com/ganrajdol99215/EasyCRUD/tree/main/backend>



```
root@ip-172-31-91-183:~# ls
root@ip-172-31-91-183:~# git clone https://github.com/ganrajdol99215/cloudblitz-student-app.git
Cloning into 'cloudblitz-student-app'...
remote: Enumerating objects: 94, done.
remote: Counting objects: 100% (94/94), done.
remote: Compressing objects: 100% (62/62), done.
remote: Total 94 (delta 16), reused 82 (delta 10), pack-reused 0 (from 0)
Receiving objects: 100% (94/94), 1.01 MiB | 23.60 MiB/s, done.
Resolving deltas: 100% (16/16), done.
root@ip-172-31-91-183:~# cd cloudblitz-student-app/
root@ip-172-31-91-183:~/cloudblitz-student-app# cd backend/
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ls
Readme.md mvnw mvnw.cmd pom.xml src
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ^C
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# apt update && apt install openjdk-17-jdk -y
java -version
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [971 kB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [174 kB]
Fetched 1397 kB in (1244 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
47 packages can be upgraded. Run 'apt list --upgradable' to see them.
Reading package lists... Done
Building dependency tree... 50%
```

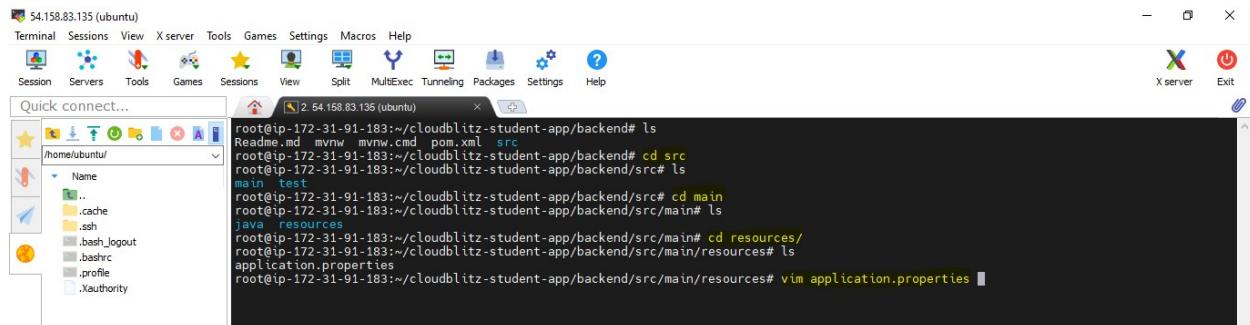
Always remember each application depends up on specific version .

Now install Maven



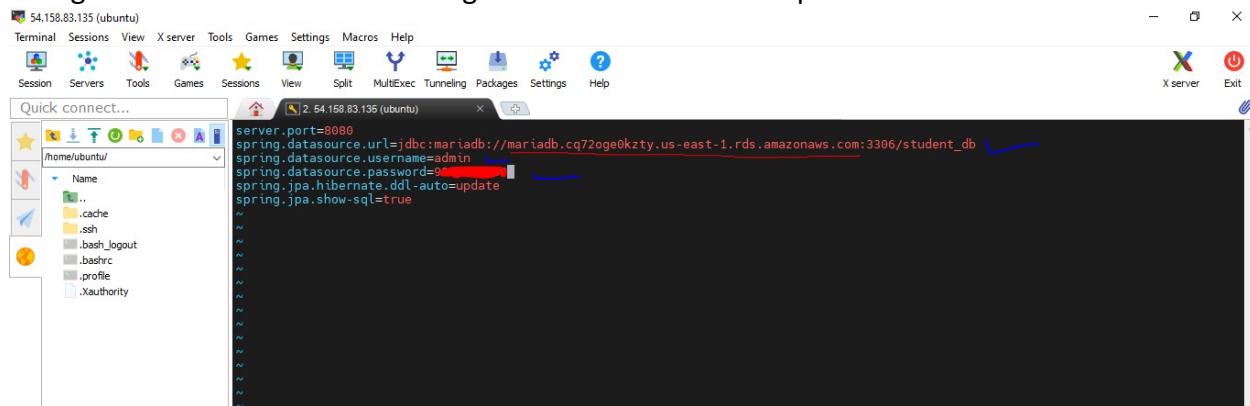
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
openjdk version "17.0.15" 2025-04-15
OpenJDK Runtime Environment (build 17.0.15+6-Ubuntu-0ubuntu124.04)
OpenJDK 64-Bit Server VM (build 17.0.15+6-Ubuntu-0ubuntu124.04, mixed mode, sharing)
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# apt install maven -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
libappliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java
libcommons-lang3-java libcommons-parent-java liberror-prone-java libgeronimo-annotation-1.3-spec-java
libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java libjansi-java libjsr305-java libmaven-parent-java
libmaven-resolver-java libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java libplexus-classworlds-java
libplexus-component-annotations-java libplexus-interpolation-java libplexus-sec-dispatcher-java libplexus-util2-java
libsisu-inject-java libsisu-plexus-java libsisu-fj java libwagon-file-java libwagon-http-shaded-java libwagon-provider-api-java
Suggested packages:
libatinject-jsr330-api-java-doc libel-apt-java libcommons-io-java-doc libasm-java libcglib-java libjsr305-java-doc
libmaven-shared-utils-java-doc liblogback-java libplexus-util2-java-doc junit4 testng libcommons-logging-java liblog4j1.2-java
The following NEW packages will be installed:
libappliance-java libapache-pom-java libatinject-jsr330-api-java libcdi-api-java libcommons-cli-java libcommons-io-java

Now you have change a file details for you have to enter in a specific in backend which **application.properties** open using vim editor .



root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ls
Readme.md mvnw mvnw.cmd pom.xml src
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# cd src
main test
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src# cd main
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main# ls
java resources
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main# cd resources/
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main/resources# ls
application.properties
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main/resources# vim application.properties

Changes need to made for accessing backend to database as per in RDS of MariaDB



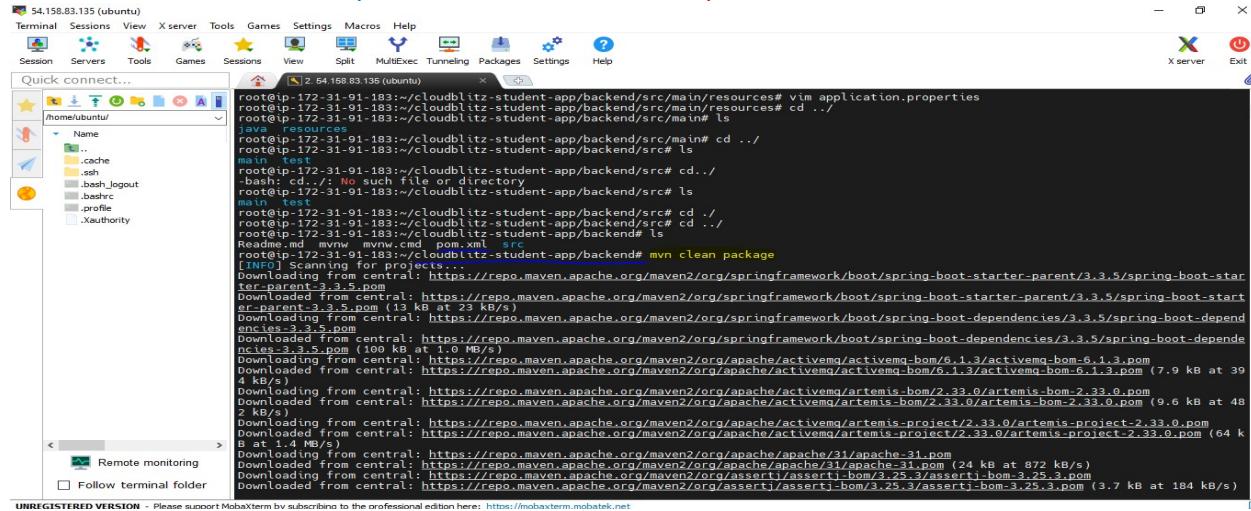
server.port=8080
spring.datasource.url=jdbc:mariadb://mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com:3306/student_db
spring.datasource.username=admin
spring.datasource.password=*****
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true

And save it :wq

Now we will build our api which is backend using maven command

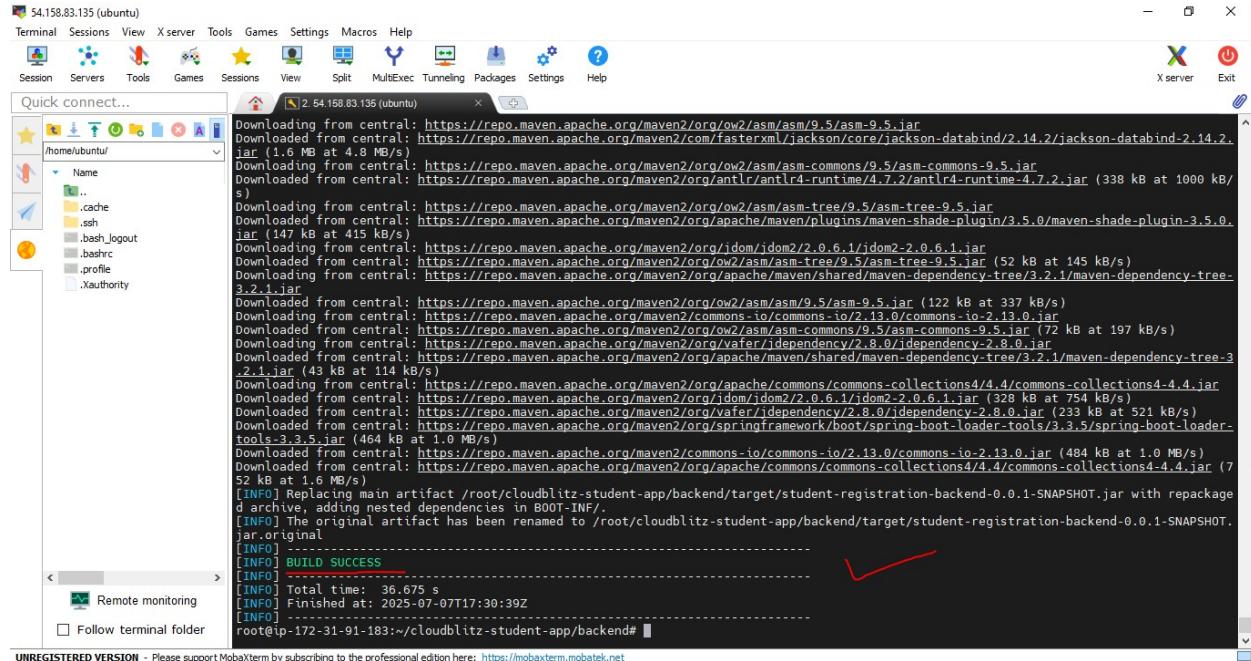
mvn clean package

but it will run whereas the pom.xml file so select that path



```
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# vim application.properties
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main/resources# cd ..
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main# ls
main  test
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src# cd ..
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main# ls
main  test
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src# cd ..
root@ip-172-31-91-183:~/cloudblitz-student-app/backend/src/main# ls
Readme.md  mvnw  mvnw.cmd  pom.xml
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# mvn clean package
[INFO] Scanning for projects...
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-starter-parent/3.3.5/spring-boot-starter-parent-3.3.5.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-starter-parent/3.3.5/spring-boot-starter-parent-3.3.5.pom (13 kB at 23 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/3.3.5/spring-boot-dependencies-3.3.5.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/3.3.5/spring-boot-dependencies-3.3.5.pom (13 kB at 23 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/activemq/activemq-bom/6.1.3/activemq-bom-6.1.3.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/activemq/activemq-bom/6.1.3/activemq-bom-6.1.3.pom (7.9 kB at 39 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/activemq/activemq-bom/2.33.0/activemq-bom-2.33.0.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/activemq/activemq-bom/2.33.0/activemq-bom-2.33.0.pom (9.6 kB at 48 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/activemq/artemis-bom/2.32.0/artemis-bom-2.32.0.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/activemq/artemis-bom/2.32.0/artemis-bom-2.32.0.pom (64 kB at 1.4 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/31/apache-31.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/apache/31/apache-31.pom (4 kB at 872 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/maven/plugins/maven-surefire-report-plugin/3.25.3/maven-surefire-report-plugin-3.25.3.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/assertj/assertj-bom/3.25.3/assertj-bom-3.25.3.pom (3.7 kB at 184 kB/s)
```

We have get Build Success Status it will shows our api ready



```
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# mvn clean package
[INFO] Scanning for projects...
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem/9.5/asem-9.5.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/com/fasterxml/jackson/core/jackson-databind/2.14.2/jackson-databind-2.14.2.jar (1.6 MB at 4.8 MB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-commons/9.5/asem-commons-9.5.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/antlr/antlr4-runtime/4.7.2/antlr4-runtime-4.7.2.jar (338 kB at 1000 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-tree/9.5/asem-tree-9.5.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/antlr/antlr4-runtime/4.7.2/antlr4-runtime-4.7.2.jar (338 kB at 1000 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-tree/9.5/asem-tree-9.5.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-tree/9.5/asem-tree-9.5.jar (52 kB at 145 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-shade-plugin/3.5.0/maven-shade-plugin-3.5.0.jar (147 kB at 419 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/dromara/IDOM/2.0.6.1/IDOM2-2.0.6.1.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-tree/9.5/asem-tree-9.5.jar (52 kB at 145 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/maven/shared/maven-dependency-tree/3.2.1/maven-dependency-tree-3.2.1.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem/9.5/asem-9.5.jar (122 kB at 337 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/commons-io/2.13.0/commons-io-2.13.0.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/commons-io/2.13.0/commons-io-2.13.0.jar (72 kB at 197 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-commons/9.5/asem-commons-9.5.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/ow2/asem/asem-commons/9.5/asem-commons-9.5.jar (72 kB at 197 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/yafei/dependency/2.8.0/dependency-2.8.0.jar
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/yafei/dependency/2.8.0/dependency-2.8.0.jar (233 kB at 521 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-loader-tools/3.3.5/spring-boot-loader-tools-3.3.5.jar (464 kB at 1.0 MB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/commons-io/2.13.0/commons-io-2.13.0.jar (484 kB at 1.0 MB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-collections4/4.4/commons-collections4-4.4.jar (752 kB at 1.6 MB/s)
[INFO] [INFO] Replacing main artifact /root/cloudblitz-student-app/backend/target/student-registration-backend-0.0.1-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF.
[INFO] [INFO] The original artifact has been renamed to /root/cloudblitz-student-app/backend/target/student-registration-backend-0.0.1-SNAPSHOT.jar.original
[INFO] [INFO] [INFO] BUILD SUCCESS
[INFO] [INFO] Total time: 36.675 s
[INFO] [INFO] Finished at: 2025-07-07T17:30:39Z
[INFO] [INFO] root@ip-172-31-91-183:~/cloudblitz-student-app/backend#
```

Then automatically target folder will be created in the same directory and in that we have snapshot.jar file which is our build file .

The screenshot shows a terminal window titled "2. 54.158.83.135 (ubuntu)". The terminal is displaying Maven dependency download logs from central repositories. It then shows the execution of a Maven command to build the application, resulting in a successful build. The final command run is "java -jar student-registration-backend-0.0.1-SNAPSHOT.jar".

```

[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-shade-plugin/3.5.0/maven-shade-plugin-3.5.0.jar (147 kB at 415 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/jdom/jdom2/2.0.6.1/jdom2-2.0.6.1.jar (52 kB at 145 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/lazymutt/asym-tree/9.5/asym-tree-9.5.jar (52 kB at 145 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-dependency-tree/3.2.1/maven-dependency-tree-3.2.1.jar (43 kB at 114 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-collections4/4.4/commons-collections4-4.4.jar (328 kB at 754 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/waterfalldependency/2.8.0/dependency-2.8.0.jar (233 kB at 521 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-loader-tools/1.3.5/spring-boot-loader-tools-1.3.5.jar (464 kB at 1.0 MB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/commons-io/2.13.0/commons-io-2.13.0.jar (484 kB at 1.0 MB/s)
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/commons/collections4/4.4/commons-collections4-4.4.jar (52 kB at 1.6 MB/s)
[INFO] [INFO] Replacing main artifact /root/cloudblitz-student-app/backend/target/student-registration-backend-0.0.1-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF/
[INFO] [INFO] The original artifact has been renamed to /root/cloudblitz-student-app/backend/target/student-registration-backend-0.0.1-SNAPSHOT.jar.original
[INFO] [INFO] -----
[INFO] [INFO] BUILD SUCCESS
[INFO] [INFO] -----
[INFO] [INFO] Total time: 36.675 s
[INFO] [INFO] Finished at: 2025-07-07T17:30:39Z
[INFO] [INFO] -----
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ls
Readme.md mvnw mvnw.cmd pom.xml src target/
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# cd target/
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# ls
classes          maven-archiver
generated-sources  maven-status
generated-test-sources student-registration-backend-0.0.1-SNAPSHOT.jar
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# test-classes
student-registration-backend-0.0.1-SNAPSHOT.jar.original
surefire-reports
test-classes
root@ip-172-31-91-183:~/cloudblitz-student-app/backend# target#

```

Now we have run our application using command and our jar file so we can run our application

java -jar student-registration-backend-0.0.1-SNAPSHOT.jar

Note : - with this command we only running on foreground which affect once we close the terminal or close Ec2 it will not run background for that you can use **nohup** command

nohup java -jar your-backend.jar > backend.log 2>&1 &

nohup java -jar student-registration-backend-0.0.1-SNAPSHOT.jar > backend.log 2>&1 &

You can use on situation need

Here we can see our backend running.



Whitelabel Error Page

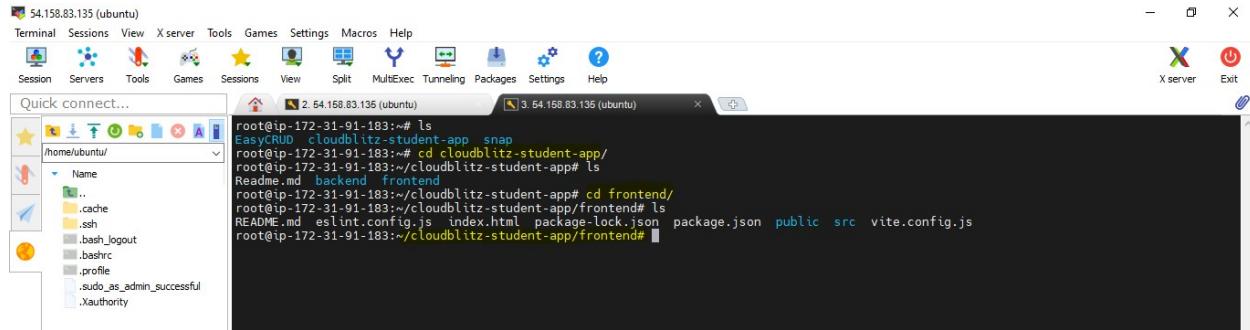
This application has no explicit mapping for /error, so you are seeing this as a fallback.

Mon Jul 07 17:46:47 UTC 2025

There was an unexpected error (type=NotFound, status=404).

Now will do the frontend process

Our frontend is on react.js

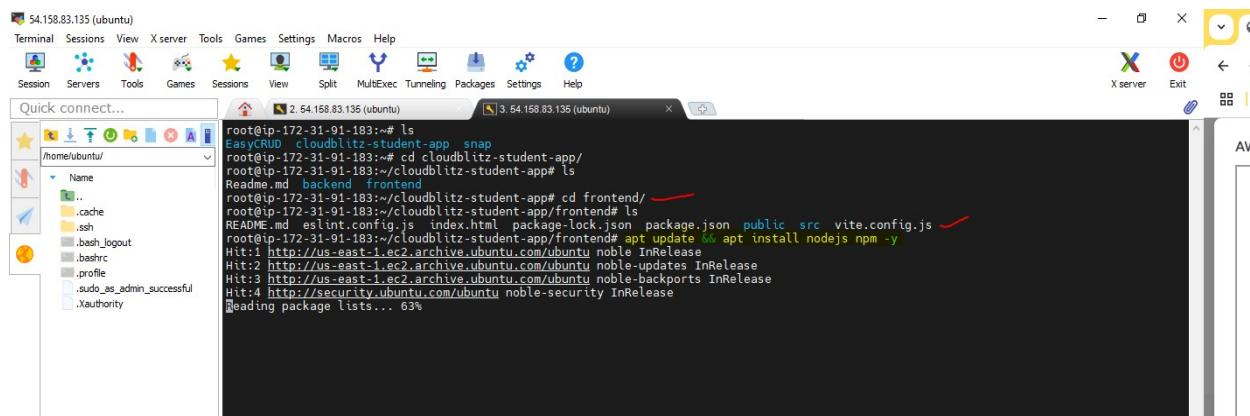


```
root@ip-172-31-91-183:~# ls
EasyCRUD  cloudblitz-student-app  snap
root@ip-172-31-91-183:~# cd cloudblitz-student-app/
root@ip-172-31-91-183:~/cloudblitz-student-app# ls
Readme.md  backend  frontend
root@ip-172-31-91-183:~/cloudblitz-student-app# cd frontend/
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# ls
README.md  eslint.config.js  index.html  package-lock.json  package.json  public  src  vite.config.js
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend#
```

Now we will install dependencies of react.js

From git repo's Readme file

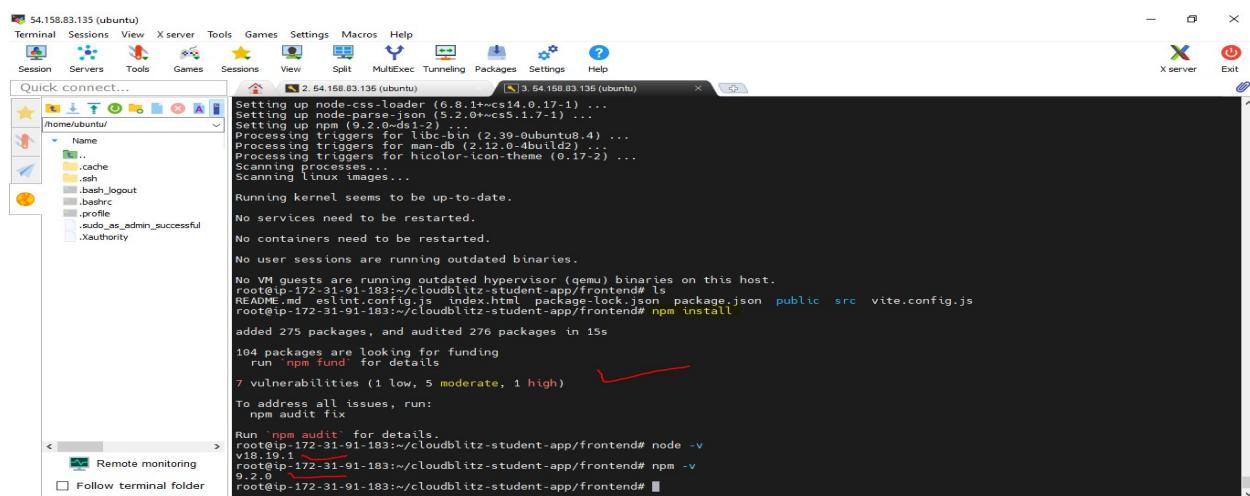
<https://github.com/ganrajdol99215/EasyCRUD/blob/main/frontend/README.md>



```
root@ip-172-31-91-183:~# ls
EasyCRUD  cloudblitz-student-app  snap
root@ip-172-31-91-183:~# cd cloudblitz-student-app/
root@ip-172-31-91-183:~/cloudblitz-student-app# ls
Readme.md  backend  frontend
root@ip-172-31-91-183:~/cloudblitz-student-app# cd frontend/
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# ls
README.md  eslint.config.js  index.html  package-lock.json  package.json  public  src  vite.config.js
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# apt update && apt install nodejs npm -y
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Hit:4 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... 63%
```

Now we have install npm important for frontend

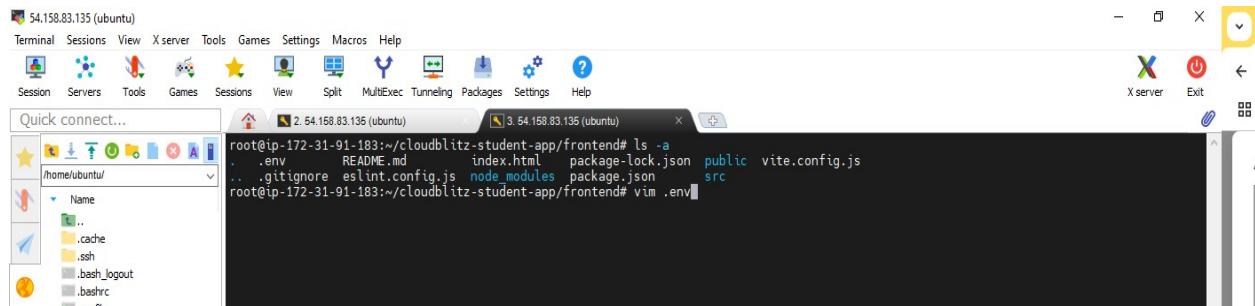
npm install



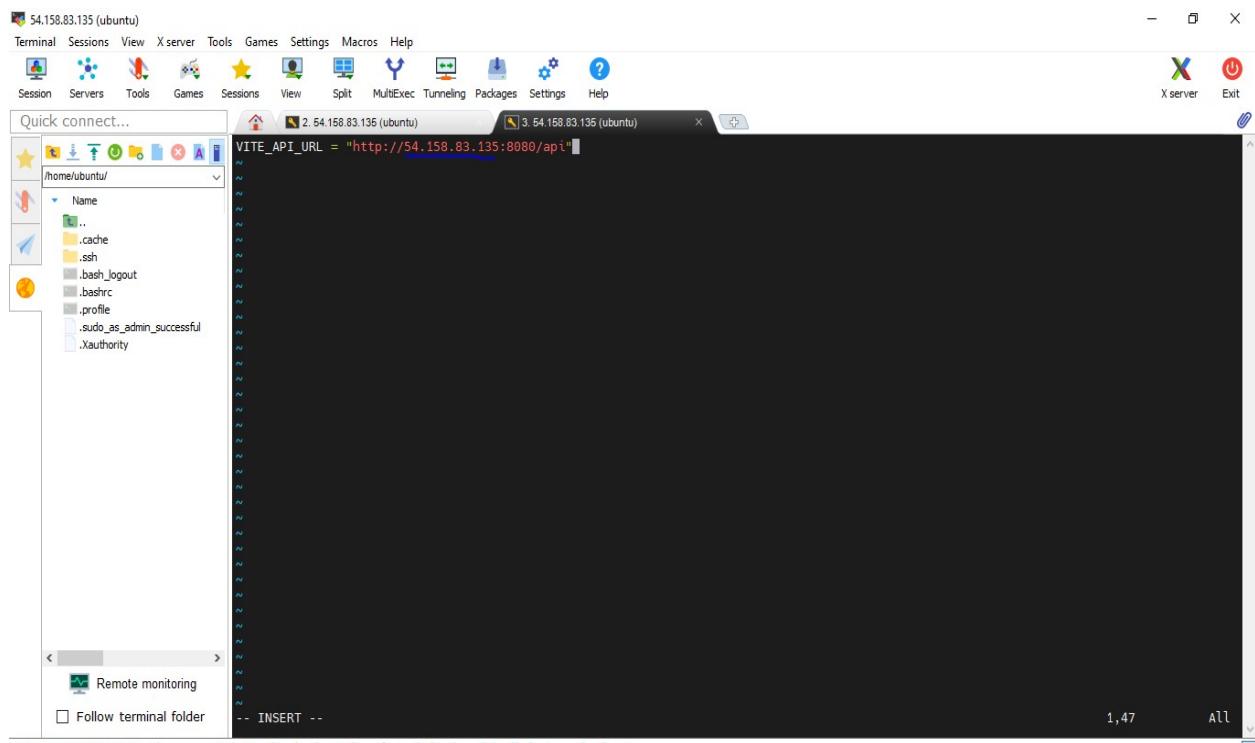
```
Setting up node-cs5-loader (6.6.1+~cs514.0.17-1) ...
Setting up node-parse-json (5.2.0+~cs5.1.7-1) ...
Setting up npm (9.2.0-ds1-2) ...
Processing triggers for libc-bin (2.39-0ubuntu8.4) ...
Processing triggers for man-db (2.12.0-4ubuntu2.1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Scanning processes...
Scanning linux images...
Running kernel seems to be up-to-date.
No services need to be restarted.
No containers need to be restarted.
No user sessions are running outdated binaries.
No VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# ls
README.md  eslint.config.js  index.html  package-lock.json  package.json  public  src  vite.config.js
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# npm install
added 275 packages, and audited 276 packages in 15s
104 packages are looking for funding
  run npm fund for details
7 vulnerabilities (1 low, 5 moderate, 1 high)
To address all issues, run:
  npm audit fix
Run 'npm audit' for details.
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# node -v
v16.14.2
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# npm -v
9.2.0
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend#
```

Now we have edit .env file in frontend

In that we have to mention details for our frontend which will host .



Replace with our EC2's public IP which host our Frontend on that public ip when we hit that URL

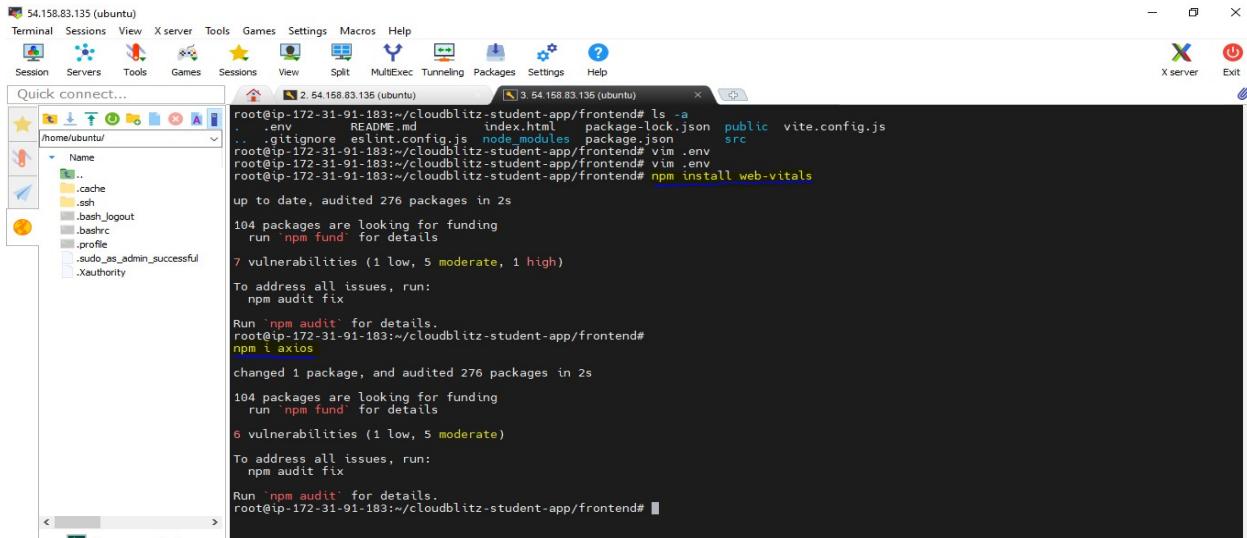


We need install some remaining dependencies

npm install web-vitals

npm i axios

This dependencies you can installed before editing .env file also



```
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# ls -a
. env  README.md  index.html  package-lock.json  public  vite.config.js
..  .gitignore  eslint.config.js  node_modules  package.json  src
.bash_logout
.bashrc
.profile
.sudo_as_admin_successful
.Xauthority

root@ip-172-31-91-183:/cloudblitz-student-app/frontend# vim .env
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# vim .env
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# npm install web-vitals

up to date, audited 276 packages in 2s
104 packages are looking for funding
  run `npm fund` for details
7 vulnerabilities (1 low, 5 moderate, 1 high)

To address all issues, run:
  npm audit fix

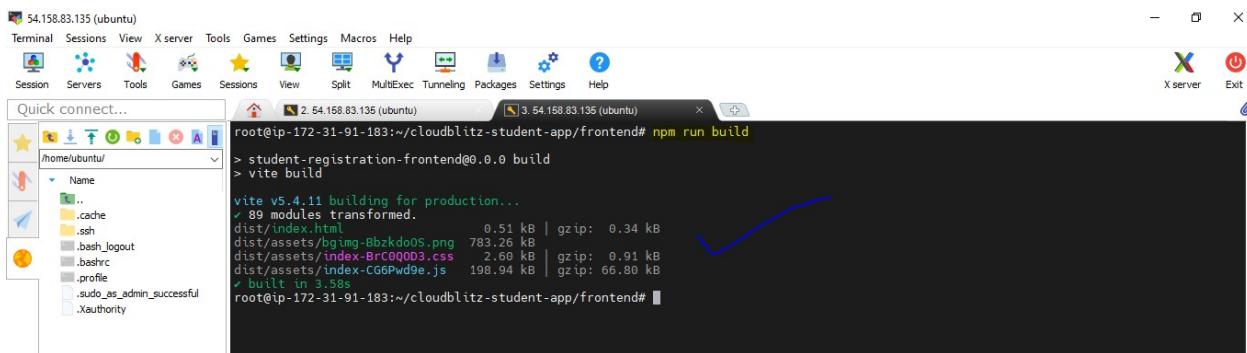
Run `npm audit` for details.
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# npm i axios
changed 1 package, and audited 276 packages in 2s
104 packages are looking for funding
  run `npm fund` for details
6 vulnerabilities (1 low, 5 moderate)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
root@ip-172-31-91-183:/cloudblitz-student-app/frontend#
```

Now we will build our frontend using command

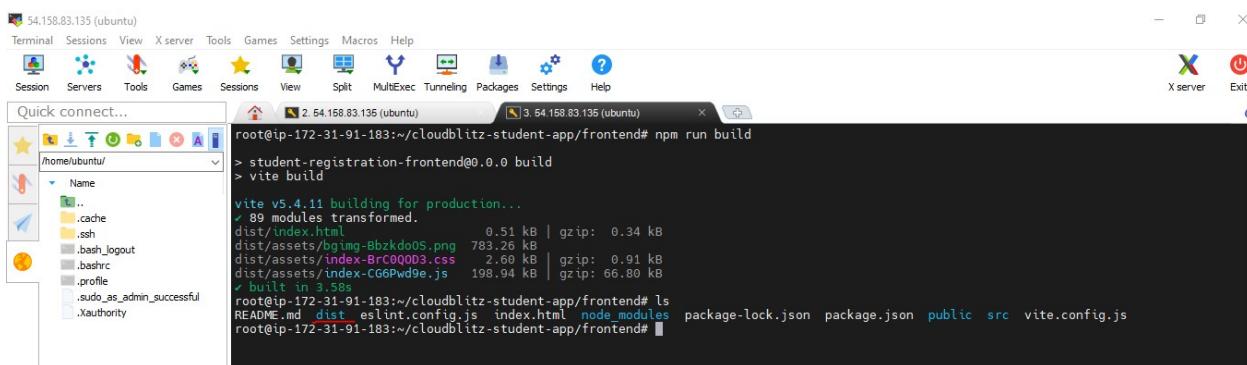
npm run build



```
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# npm run build
> student-registration-frontend@0.0.0 build
> vite build

vite v5.4.11 building for production...
✓ 89 modules transformed.
dist/index.html          0.51 kB | gzip:  0.34 kB
dist/assets/bimg-Bbzkd0o5.png 783.26 kB
dist/assets/index-Brc00003.css 2.60 kB | gzip:  0.91 kB
dist/assets/index-CG6Pwde.js 198.94 kB | gzip: 66.80 kB
✓ built in 3.58s
root@ip-172-31-91-183:/cloudblitz-student-app/frontend#
```

In the **dist** our build file generated we will run that file using **apache server**



```
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# npm run build
> student-registration-frontend@0.0.0 build
> vite build

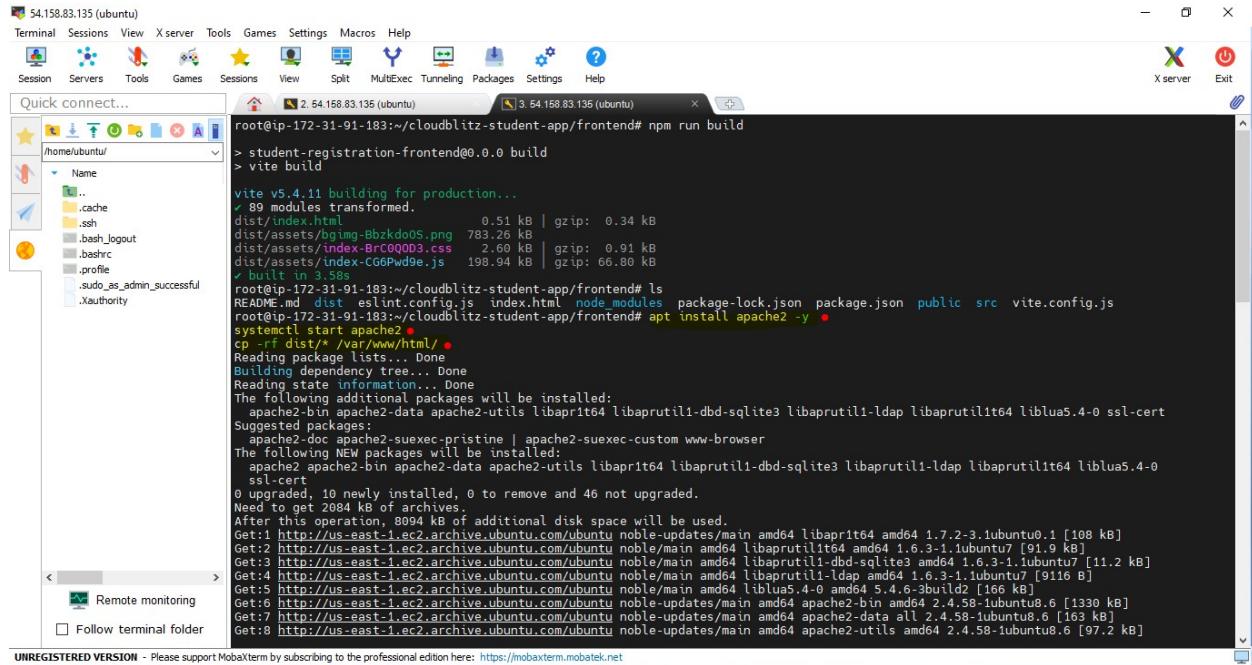
vite v5.4.11 building for production...
✓ 89 modules transformed.
dist/index.html          0.51 kB | gzip:  0.34 kB
dist/assets/bimg-Bbzkd0o5.png 783.26 kB
dist/assets/index-Brc00003.css 2.60 kB | gzip:  0.91 kB
dist/assets/index-CG6Pwde.js 198.94 kB | gzip: 66.80 kB
✓ built in 3.58s
root@ip-172-31-91-183:/cloudblitz-student-app/frontend# ls
README.md  dist  eslint.config.js  index.html  node_modules  package-lock.json  package.json  public  src  vite.config.js
root@ip-172-31-91-183:/cloudblitz-student-app/frontend#
```

<https://github.com/ganrajdol99215/EasyCRUD/blob/main/frontend/README.md>

from here you get command to run install apache server

which will do the process copying data build file from **dist** to specific directory in vm which is **/var/www/html/**

```
apt install apache2 -y  
systemctl start apache2  
cp -rf dist/* /var/www/html/
```



```
54.158.83.135 (ubuntu)  
Terminal Sessions View Xserver Tools Games Settings Macros Help  
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help  
Quick connect... 2. 54.158.83.135 (ubuntu) 3. 54.158.83.135 (ubuntu)  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# npm run build  
> student-registration-frontend@0.0.0 build  
> vite build  
  
vite v5.4.11 building for production...  
✓ 89 modules transformed.  
dist/index.html 0.51 kB | gzip: 0.34 kB  
dist/assets/bimg-BBzKd005.png 783.26 kB  
dist/assets/index-BrcQ00D3.css 2.60 kB | gzip: 0.91 kB  
dist/assets/index-CG6Pwde.js 198.94 kB | gzip: 66.80 kB  
✓ built in 3.58s  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# ls  
README.md dist eslint.config.js index.html node_modules package-lock.json package.json public src vite.config.js  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# apt install apache2 -y •  
systemctl start apache2•  
cp -rf dist/* /var/www/html/•  
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 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert  
Suggested packages:  
apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser  
The following NEW packages will be installed:  
apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0  
ssl-cert  
0 upgraded, 10 newly installed, 0 to remove and 46 not upgraded.  
Need to get 2084 kB of additional disk space.  
After this operation, 8094 kB of additional disk space will be used.  
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libapr1t64 amd64 1.7.2-3.1ubuntu0.1 [108 kB]  
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]  
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]  
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]  
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]  
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu0.6 [1330 kB]  
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu0.6 [163 kB]  
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu0.6 [97.2 kB]  
UNREGISTERED VERSION - Please support MobaTerm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

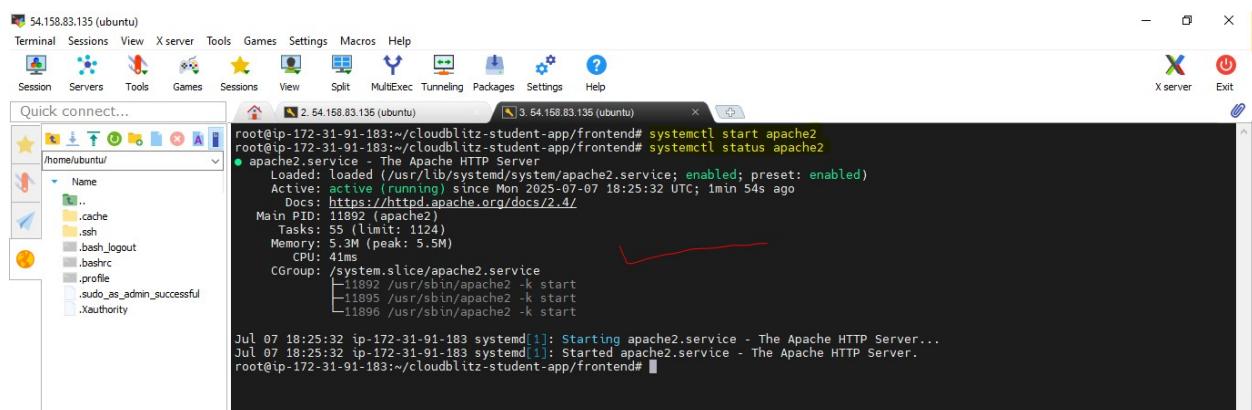
We will start apache server now to run our web application

Using command

```
systemctl start apache2
```

and check status

```
systemctl status apache2
```



```
54.158.83.135 (ubuntu)  
Terminal Sessions View Xserver Tools Games Settings Macros Help  
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help  
Quick connect... 2. 54.158.83.135 (ubuntu) 3. 54.158.83.135 (ubuntu)  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# systemctl start apache2  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend# systemctl status apache2  
● apache2.service - The Apache HTTP Server  
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)  
   Active: active (running) since Mon 2025-07-07 18:25:32 UTC; 1min 54s ago  
     Docs: https://httpd.apache.org/docs/2.4/  
   Main PID: 11892 (apache2)  
      Tasks: 55 (limit: 1124)  
        Memory: 5.3M (peak: 5.5M)  
         CPU: 41ms  
        CGroup: /system.slice/apache2.service  
                └─11892 /usr/sbin/apache2 -k start  
                    ├─11895 /usr/sbin/apache2 -k start  
                    ├─11896 /usr/sbin/apache2 -k start  
Jul 07 18:25:32 ip-172-31-91-183 systemd[1]: Starting apache2.service - The Apache HTTP Server...  
Jul 07 18:25:32 ip-172-31-91-183 systemd[1]: Started apache2.service - The Apache HTTP Server.  
root@ip-172-31-91-183:~/cloudblitz-student-app/frontend#
```

Now we will check our application by visiting the host IP (**ec2's public IP**)

CLOUBLITZ Student Registration

Name	Email	Course	Class	Percentage	Branch	Mobile Number	Actions
anonymous	xyx@gmail.com	DevOps	MCA	90	Computer App	984	Delete

Fill-up the data and click register it will store your data

CLOUBLITZ Student Registration

Name	Email	Course	Class	Percentage	Branch	Mobile Number	Actions
anonymous	xyx@gmail.com	DevOps	MCA	90	Computer App	984	Delete

Your information has been saved!

Name	Email	Course	Class	Percentage	Branch	Mobile Number	Actions
anonymous	xyx@gmail.com	DevOps	MCA	90	Computer App	984	Delete

Now we will check to our database is data stores over there or not for that we have connect with our database and check tables

```
mysql -u admin -p9921569869 -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
```

The screenshot shows a terminal window with two tabs. The left tab is titled '2. 54.158.83.135 (ubuntu)' and the right tab is titled '3. 54.158.83.135 (ubuntu)'. The right tab contains the following terminal session:

```
root@ip-172-31-91-183:~# cd /cloudblitz-student-app# cd /cloudblitz-student-app# mysql -u admin -p9921569869 -h mariadb.cq72oge0kzty.us-east-1.rds.amazonaws.com
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 337
Server version: 11.4.5-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)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| innodb |
| mysql |
| performance_schema |
| student_db |
| sys |
+-----+
6 rows in set (0.003 sec)

MariaDB [(none)]> use student_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [student_db]> select * from user;
+----+----+----+----+----+----+----+
| id | branch | course | email | mobile_number | name | percentage | student_class |
+----+----+----+----+----+----+----+
| 1 | Computer App | DevOps | xyz@gmail.com | 9849455655 | anonymous | 90 | MCA |
+----+----+----+----+----+----+----+
1 row in set (0.004 sec)

MariaDB [student_db]>
```

In the Screenshot we can clearly see we inserted data registered data successfully store in our database so this an 3 tier application using RDS(MariaDB)—GitHub has a repo--EC2(ubuntu) as a host Database –Backend(API)—Frontend

So we have successfully deployed 3 tier Web Application.