

Task -1

For Monolithic

1) Launched a single EC2 instance with the desired instance type (t2.micro) and AMI (Ubuntu) and enable all traffic.

The screenshot displays the AWS Management Console interface for the 'us-east-1' region. The left-hand navigation pane shows various services, with 'Instances' selected under the 'EC2' category. The main content area, titled 'Instances (1/1)', shows a table with one instance: 'wordpress' (ID: i-09c810c812cf3a479), which is in the 'Running' state. Below the table, the 'Instance: i-09c810c812cf3a479 (wordpress)' details are expanded, showing a summary of its configuration. The instance is running on a 't2.micro' instance type with the 'Ubuntu' AMI. It has a public IPv4 address of 54.167.50.6 and a private IP of 172.31.31.148. The instance is located in the 'us-east-1c' availability zone. The 'Instance summary' section provides a detailed overview of the instance's configuration, including its ID, name, state, type, and various addresses and DNS names.

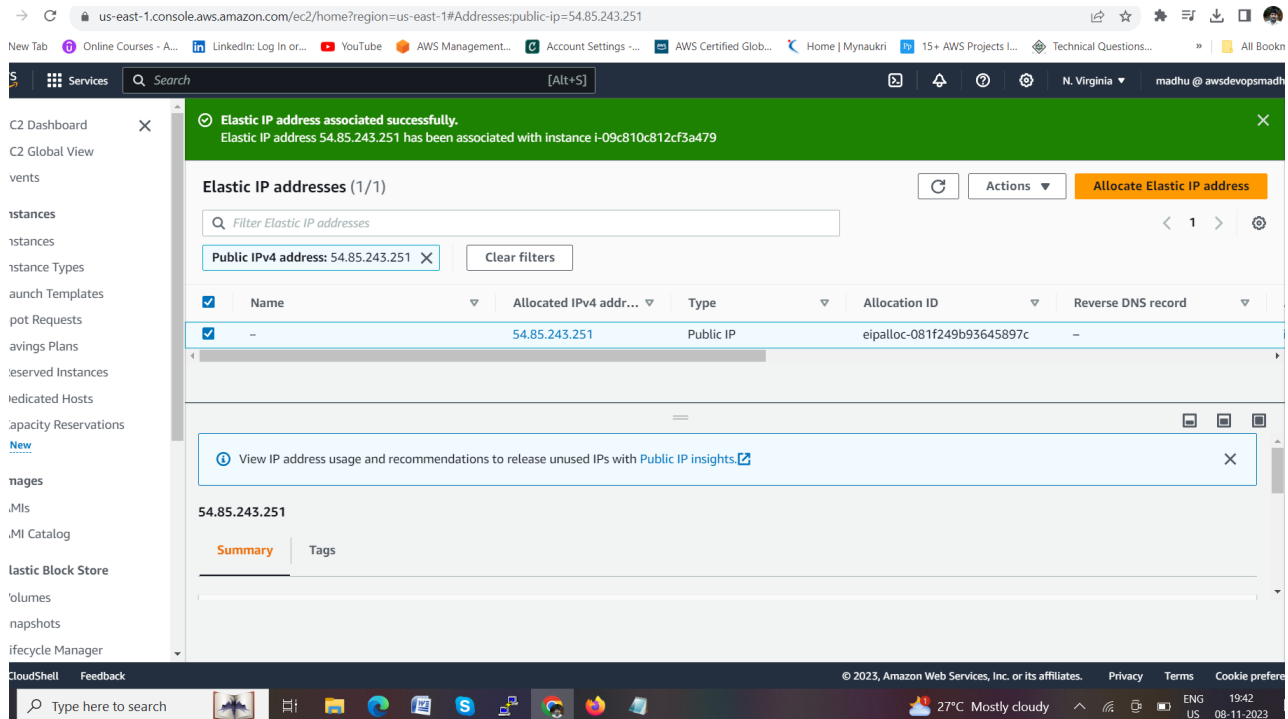
Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Pub
wordpress	i-09c810c812cf3a479	Running	t2.micro	2/2 checks passed	No alarms	us-east-1c	ec2

Instance: i-09c810c812cf3a479 (wordpress)

Instance summary

Instance ID	Public IPv4 address	Private IPv4 addresses
i-09c810c812cf3a479 (wordpress)	54.167.50.6 open address	172.31.31.148
IPv6 address	Instance state	Public IPv4 DNS
-	Running	ec2-54-167-50-6.compute-1.amazonaws.com lope address
Hostname type	Private IP DNS name (IPv4 only)	Elastic IP addresses
IP name: ip-172-31-31-148.ec2.internal	ip-172-31-31-148.ec2.internal	-
Answer private resource DNS name	Instance type	
IPv4 (A)	t2.micro	

2) To keep a static IP address, we will create an Elastic IP address and associate it



3) Use MobaXterm to sign into the instance using the key created during instance.

4) Update and install my sql along with apache 2 web server.

5) sudo apt update
sudo apt install apache2 php lib apache2-mod-php php-mysqmysql-server-y

6)Login to MySQL server
sudo mysql-u root

7) Create a new database.
CREATE DATA BASE word
press;

8)Create a user and set a password:
CREATE USER 'admin'@'localhost'IDENTIFIEDBY'wpadminpass';

9)Grant the user permissions on the data base and apply the changes. GRANT ALL PRIVILEGES ON wordpress.* TO 'admin'@'localhost'; FLUSH PRIVILEGES;

10) Select the word press data base to confirm if the data base is created USE word press;
SELECTDATABASE();

exit

```
mysql> SELECT DATABASE();
+-----+
| DATABASE() |
+-----+
| wordpress  |
+-----+
1 row in set (0.00 sec)
```

11) Install word press(download and extract)

Download Word Press : `wget https://wordpress.org/latest.tar.gz`

Extract: `tar -xvzf latest.tar.gz`

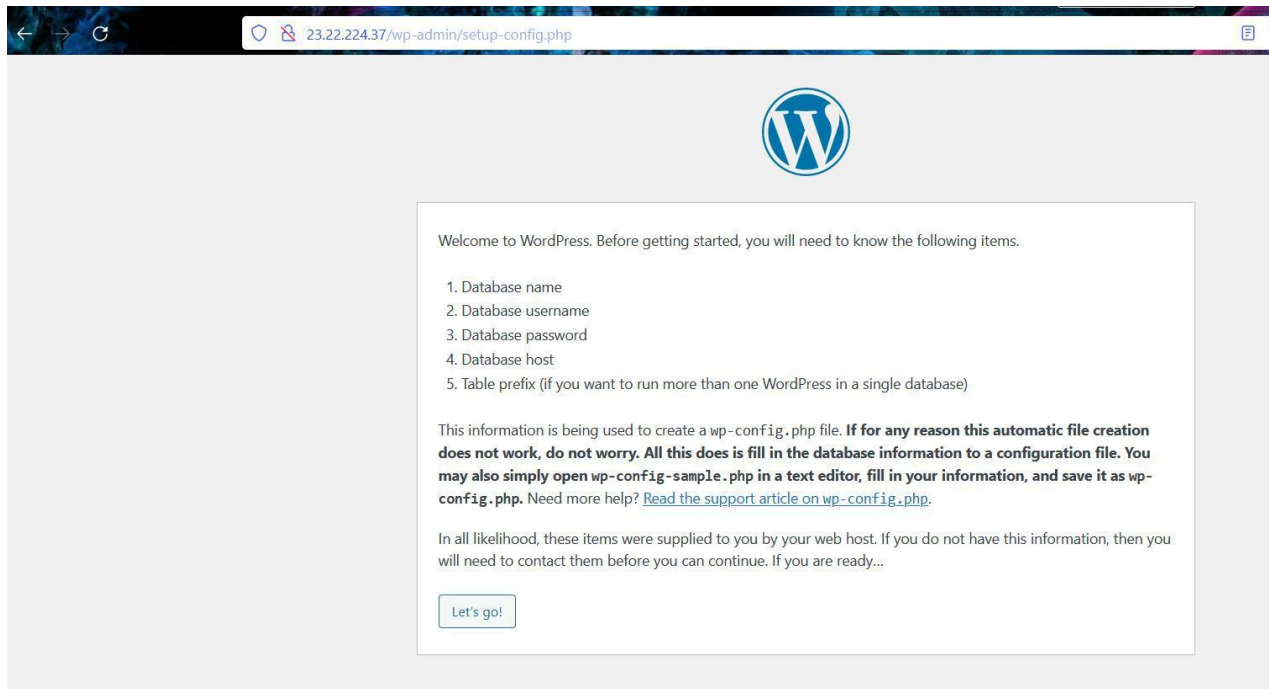
12) Move files to the web server directory

`sudo mv wordpress/* /var/www/html/`

13) remove the index.html file as that would open the apache2 default page

`sudo rm -rf index.html`

Enter the public IP in the URL of the browser to get the word press site



14) You can enter the same details that you have used earlier to create the data base or you can directly edit the wp-config.php file

```
GNU nano 6.2 wp-config.php *
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the installation.
 * You don't have to use the web site, you can copy this file to "wp-config.php"
 * and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * Database settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/documentation/article/editing-wp-config-php/
 *
 * @package WordPress
 */

/** Database settings - You can get this info from your web host */
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** Database username */
define( 'DB_USER', 'ubuntu' );

/** Database password */
define( 'DB_PASSWORD', 'Intern@1' );

/** Database hostname */
define( 'DB_HOST', 'localhost' );

/** Database charset to use in creating database tables. */
define( 'DB_CHARSET', 'utf8mb4' );

/** The database collate type. Don't change this if in doubt. */
define( 'DB_COLLATE', '' );

/**#@+
 *
 *
 *
 */

^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute   ^C Location  M-U Und
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify   ^/_ Go To Line M-E Red
```

15) Once done you can enter the Site title and the user name to set up the word press site.

WordPress Setup Configuration

Below you should enter your database connection details. If you are not sure about these, contact your host.

Database Name:
The name of the database you want to use with WordPress.

Username:
Your database username.

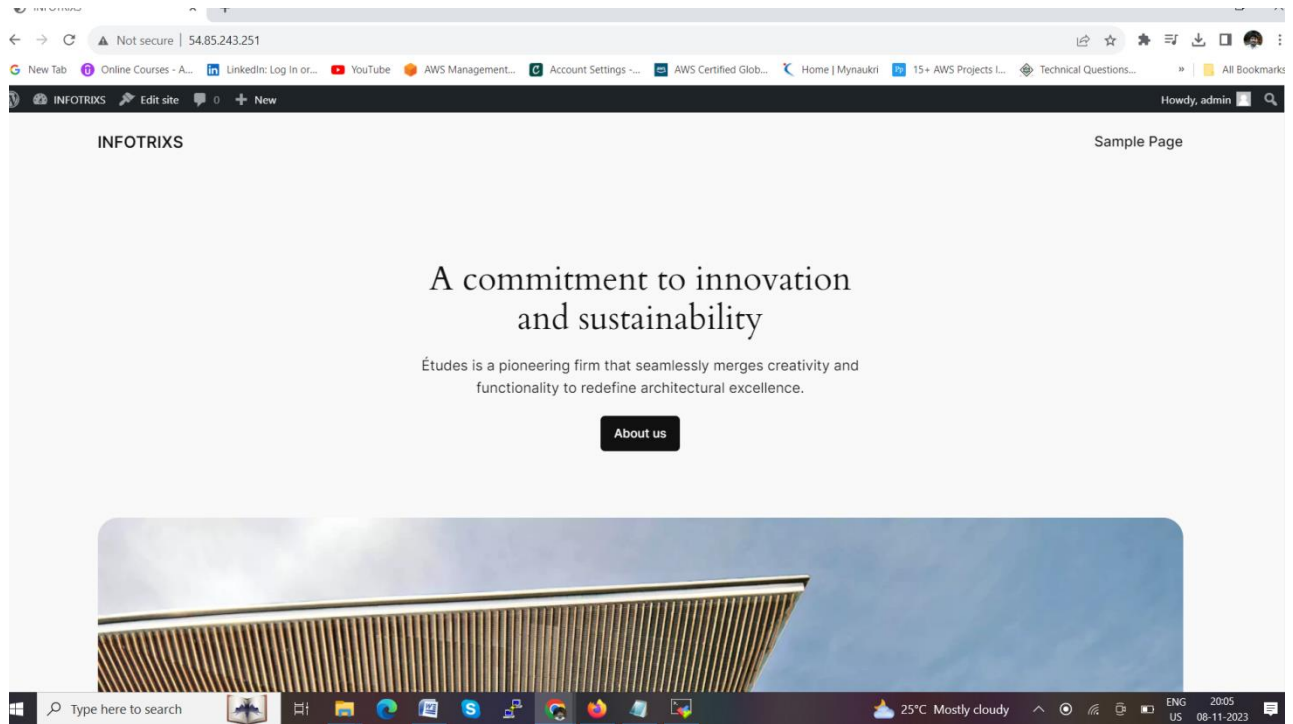
Password: [Hide](#)
Your database password.

Database Host:
You should be able to get this info from your web host, if localhost does not work.

Table Prefix:
If you want to run multiple WordPress installations in a single database, change this.

[Submit](#)

16) Sample word press website

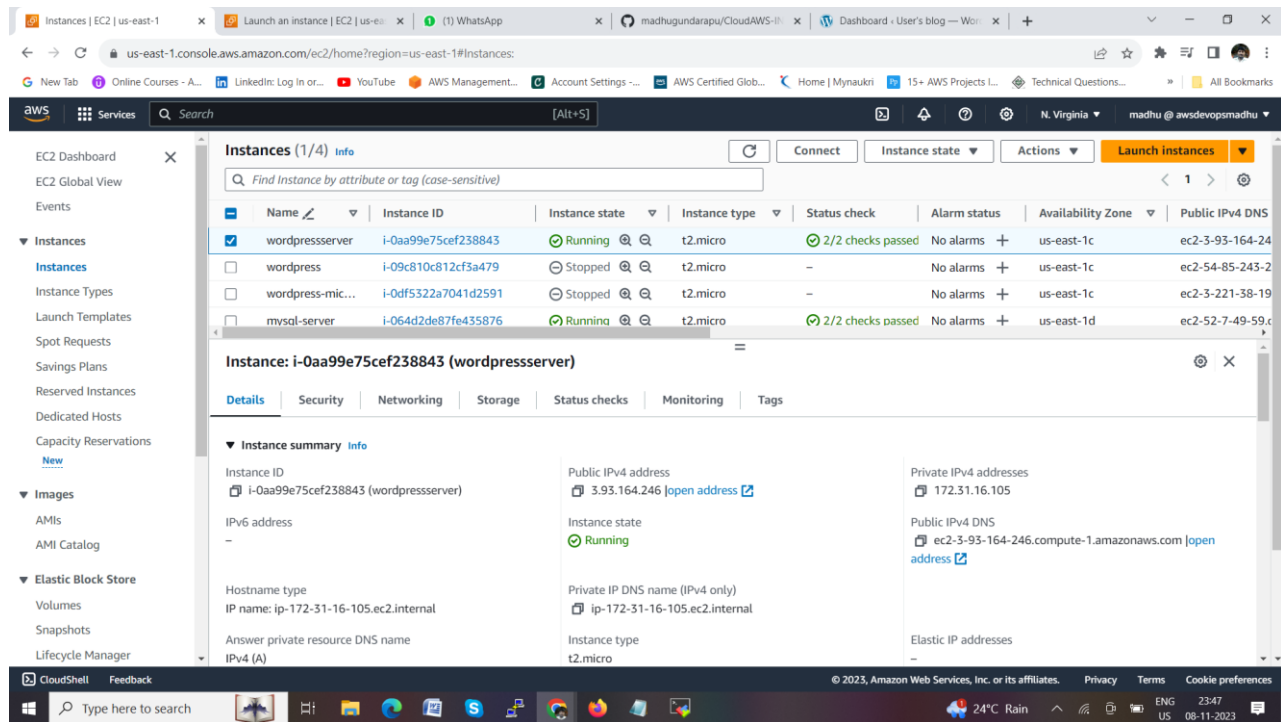


Task 2

For Micro services: 2 EC2 instances , 1 for Wordpress and 1 EC2 instances, 1 for MYSQL.

1 EC2 instance for Wordpress

Create a Wordpress instance launch



The screenshot displays the AWS Management Console interface for EC2 instances in the us-east-1 region. The left sidebar shows the navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, Snapshots, and Lifecycle Manager. The main content area shows a table of instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. The 'wordpressserver' instance is selected, and its details are shown below the table.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
wordpressserver	i-0aa99e75cef238843	Running	t2.micro	2/2 checks passed	No alarms	us-east-1c	ec2-3-93-164-24
wordpress	i-09c810c812cf3a479	Stopped	t2.micro	-	No alarms	us-east-1c	ec2-54-85-243-2
wordpress-mic...	i-0df5322a7041d2591	Stopped	t2.micro	-	No alarms	us-east-1c	ec2-3-221-38-19
mysql-server	i-064d2de87fe435876	Running	t2.micro	2/2 checks passed	No alarms	us-east-1d	ec2-52-7-49-59.c

Instance: i-0aa99e75cef238843 (wordpressserver)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary Info

Instance ID: i-0aa99e75cef238843 (wordpressserver)

Public IPv4 address: 3.93.164.246 [open address](#)

Private IPv4 addresses: 172.31.16.105

Public IPv4 DNS: ec2-3-93-164-246.compute-1.amazonaws.com [open address](#)

Private IP DNS name (IPv4 only): ip-172-31-16-105.ec2.internal

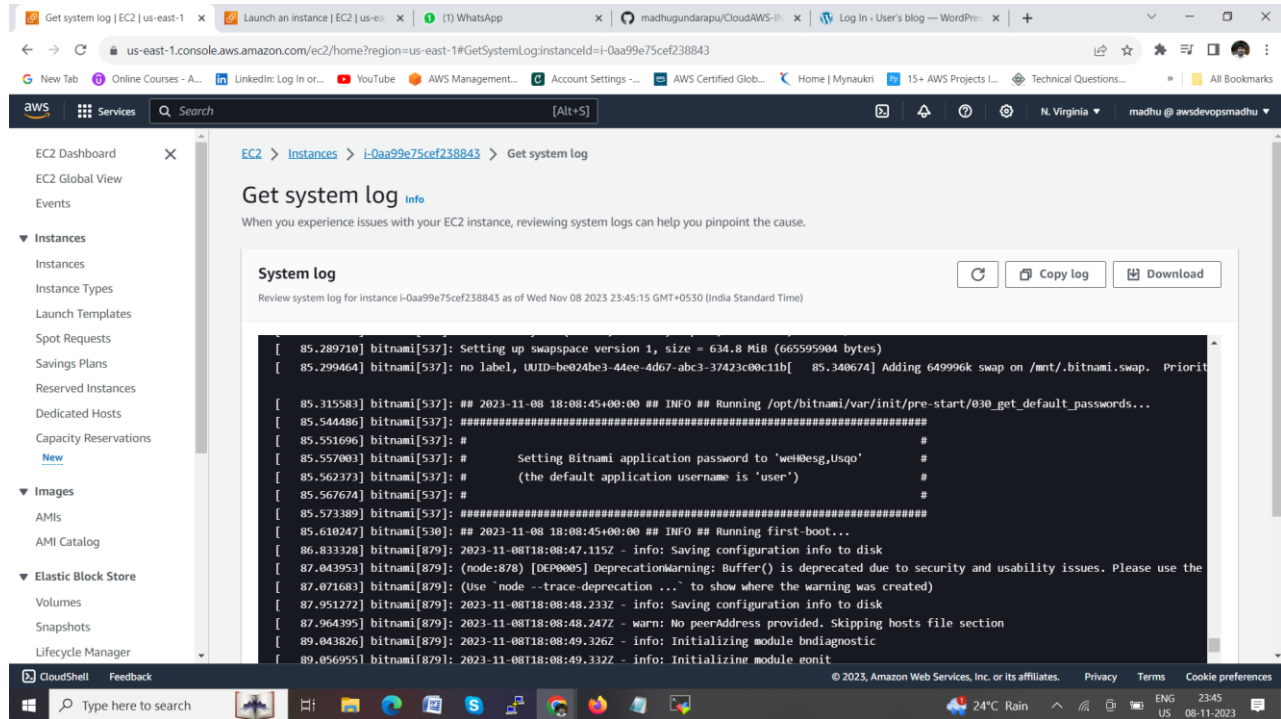
Instance state: Running

Instance type: t2.micro

Hostname type: IP name: ip-172-31-16-105.ec2.internal

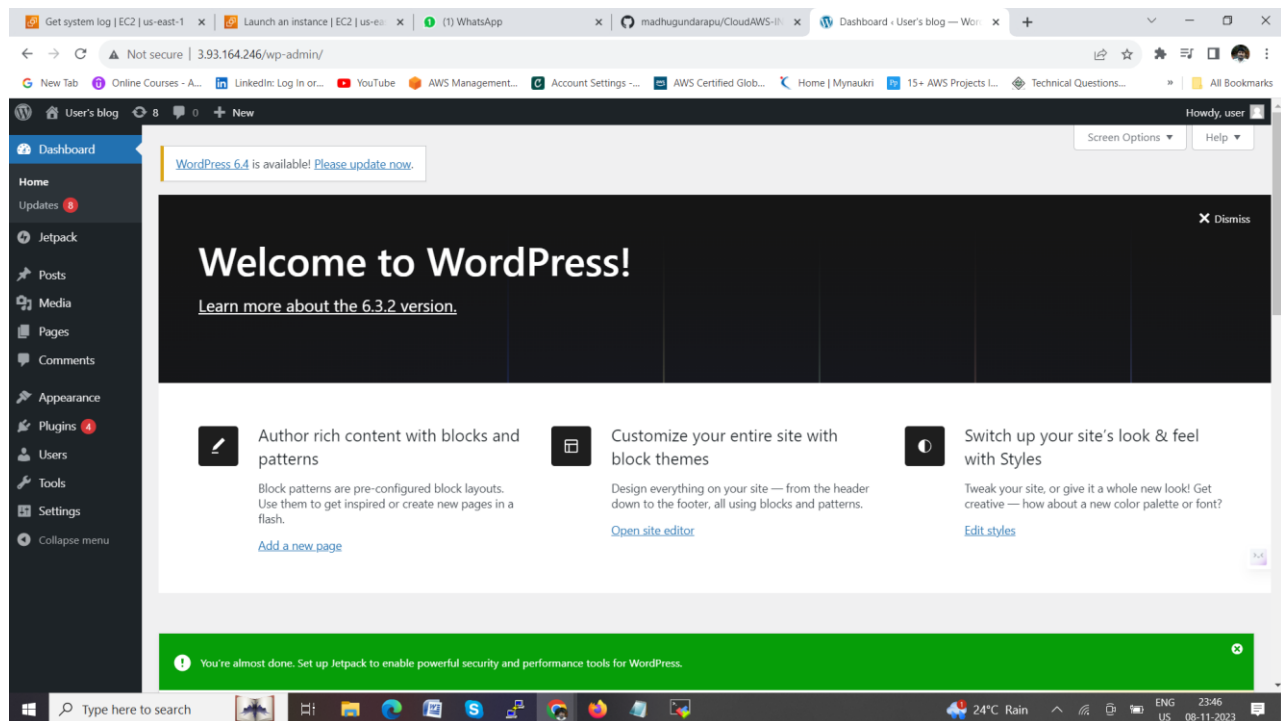
Answer private resource DNS name: IPv4 (A)

Click actions and Monitor troubleshoot and access Get the system log
Get System Log



Click the password for WordPress Login

Dashboard Showing in WordPress



1 EC2 Instance for MYSQL :

Create a MYSQL Server instance Launch

Command:

```
sudo apt update
```

```
sudo apt install mysql-server
```

```
sudo systemctl status mysql
```

```
sudo mysql
```

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH
```

```
mysql_native_password BY 'Prasan@12';
```

```
CREATE DATABASE mysql_test;
```

```
USE mysql_test;
```

```
CREATE TABLE table1 ( id INT, name VARCHAR(45));
```

```
INSERT INTO table1 VALUES(1, 'INFOTRIX'), (2,'CLOUD'),
```

```
(3,'DEVOPS'), (4,'PRASAN');
```

```
SELECT *FROM table1;
```

The screenshot displays the AWS Management Console interface. The left sidebar shows the navigation menu with categories like EC2 Dashboard, Instances, Images, and Elastic Block Store. The main content area shows the 'Instances (1/4)' page. A table lists the instances, with the 'mysql-server' instance (ID: i-064d2de87fe435876) selected. Below the table, the details for this instance are shown, including its state (Running), IP addresses, and DNS information.

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
wordpressserver	i-0aa99e75cef238843	Running	t2.micro	2/2 checks passed	No alarms	us-east-1c	ec2-3-93-164-24
wordpress	i-09c810c812cf3a479	Stopped	t2.micro	-	No alarms	us-east-1c	ec2-54-85-243-2
wordpress-mic...	i-0df5322a7041d2591	Stopped	t2.micro	-	No alarms	us-east-1c	ec2-3-221-38-19
mysql-server	i-064d2de87fe435876	Running	t2.micro	2/2 checks passed	No alarms	us-east-1d	ec2-52-7-49-59.c

Instance: i-064d2de87fe435876 (mysql-server)

Details | Security | Networking | Storage | Status checks | Monitoring | Tags

Instance summary

- Instance ID: i-064d2de87fe435876 (mysql-server)
- Public IPv4 address: 52.7.49.59 [open address](#)
- Private IPv4 addresses: 172.31.33.92
- Instance state: **Running**
- Public IPv4 DNS: ec2-52-7-49-59.compute-1.amazonaws.com [open address](#)
- Private IP DNS name (IPv4 only): ip-172-31-33-92.ec2.internal
- Instance type: t2.micro
- IPV6 address: -
- Answer private resource DNS name: IPV4 (A)