Task:-

1. WORDPRESS installation
2. PHPMYADMIN
3. Multiple PHP version

Pre-Request:

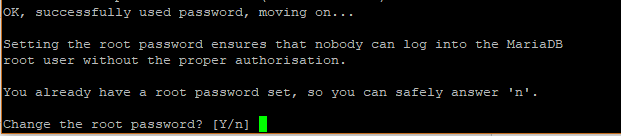
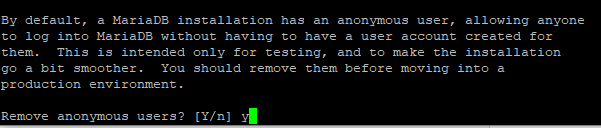
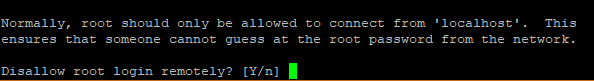
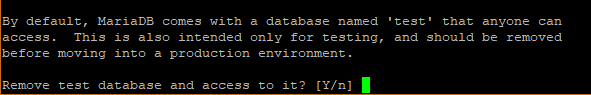
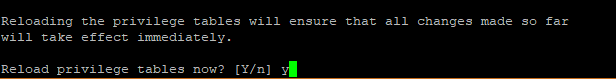
Ubuntu server on t2 micro

1. Launch a ubuntu sever on t2 micro and connect with ssh
2. apt update && apt upgrade
3. Install Apache on server with apt install apache2 -y
4. systemctl start apache2 && systemctl enable apache2
5. Next, we are going to install the MariaDB database engine to hold our Wordpress files. MariaDB is an open-source fork of MySQL and most of the hosting companies use it instead of MySQL.

apt install mariadb-server mariadb-client

Let’s now secure our MariaDB database engine and disallow remote root login.

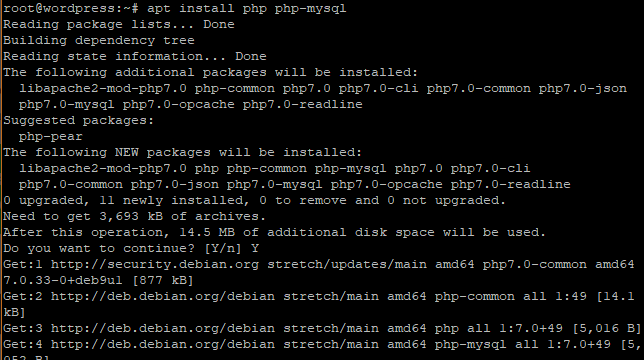
$ mysql\_secure\_installation

The first step will prompt you to change the root password to login to the database. You can opt to change it or skip if you are convinced that you have a strong password. To skip changing type n.For safety’s sake, you will be prompted to remove anonymous users. Type Y.Next, disallow remote root login to prevent hackers from accessing your database. However, for testing purposes, you may want to allow log in remotely if you are configuring a virtual serverNext, remove the test database.Finally, reload the database to effect the changes.

[**Step 3: Install PHP**](https://www.digitalocean.com/community/tutorials/install-wordpress-on-ubuntu#step-3-install-php)

Lastly, we will install PHP as the last component of the LAMP stack.

apt install php php-mysql

**Output**To confirm that PHP is installed , created a info.php file at /var/www/html/ path

vim /var/www/html/info.php

Append the following lines:

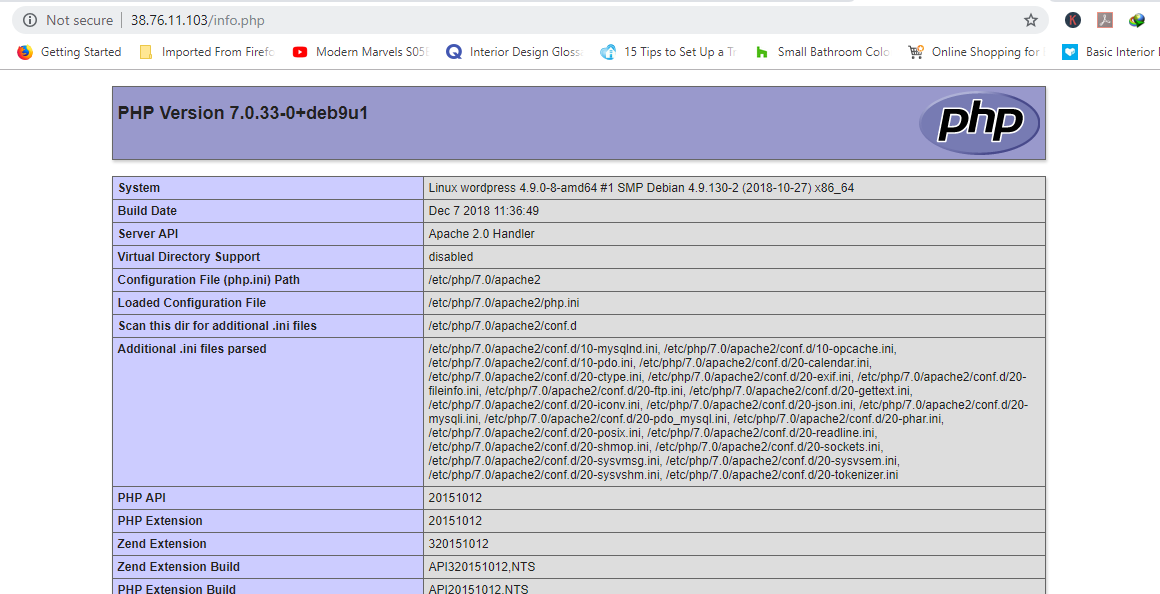
<?php

phpinfo();

?>

Save and Exit. Open your browser and append /info.php to the server’s URL.

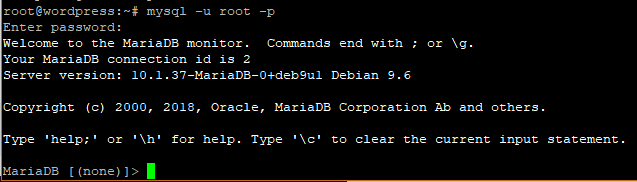
https://ip-address/info.php

**Output**

[**Step 4: Create WordPress Database**](https://www.digitalocean.com/community/tutorials/install-wordpress-on-ubuntu#step-4-create-wordpress-database)

Now it’s time to log in to our MariaDB database as root and create a database for accommodating our WordPress data.

$ mysql -u root -p

**Output**Create a database for our WordPress installation.

CREATE DATABASE wordpress\_db;

**Output**Create Wordpress DatabaseNext, create a database user for our WordPress setup.

CREATE USER 'wp\_user'@'localhost' IDENTIFIED BY 'password';

**Output**Create User For Wordpress DatabaseGrant privileges to the user Next, grant the user permissions to access the database

GRANT ALL ON wordpress\_db.\* TO 'wp\_user'@'localhost' IDENTIFIED BY 'password';

**Output**Grant Privileges To Wp User On Wordpress DatabaseGreat, now you can exit the database.

FLUSH PRIVILEGES;

Exit;

[**Step 5: Install WordPress CMS**](https://www.digitalocean.com/community/tutorials/install-wordpress-on-ubuntu#step-5-install-wordpress-cms)

Go to your temp directory and download the latest WordPress File

wget https://wordpress.org/latest.zip

**Output**Next, Uncompress the tarball which will generate a folder called “wordpress”.

unzip latest.tar.gz

**Output**Copy the wordpress folder to /var/www/html/ path.

cp -R wordpress /var/www/html/

Run the command below to change ownership of ‘wordpress’ directory.

chown -R www-data:www-data /var/www/html/wordpress/

change File permissions of the WordPress folder.

chmod -R 755 /var/www/html/wordpress/

Create ‘uploads’ directory.

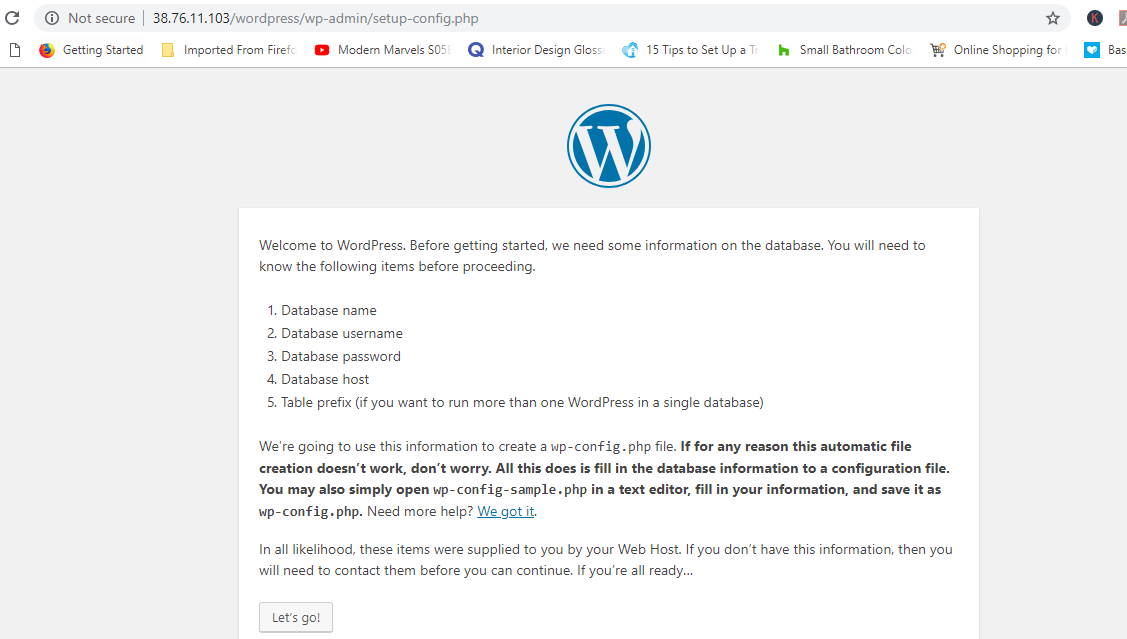
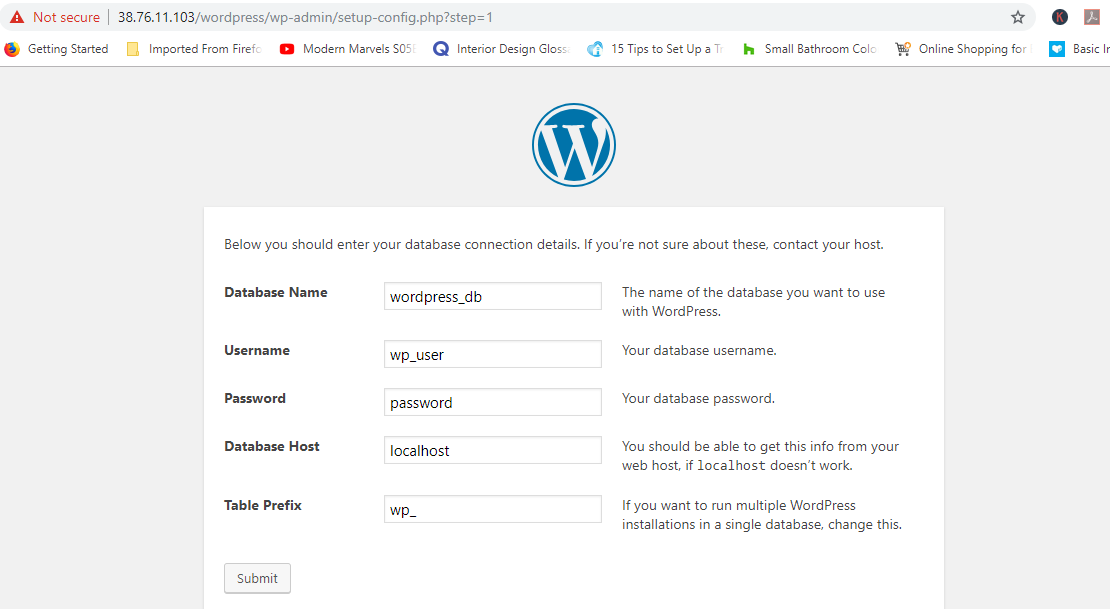
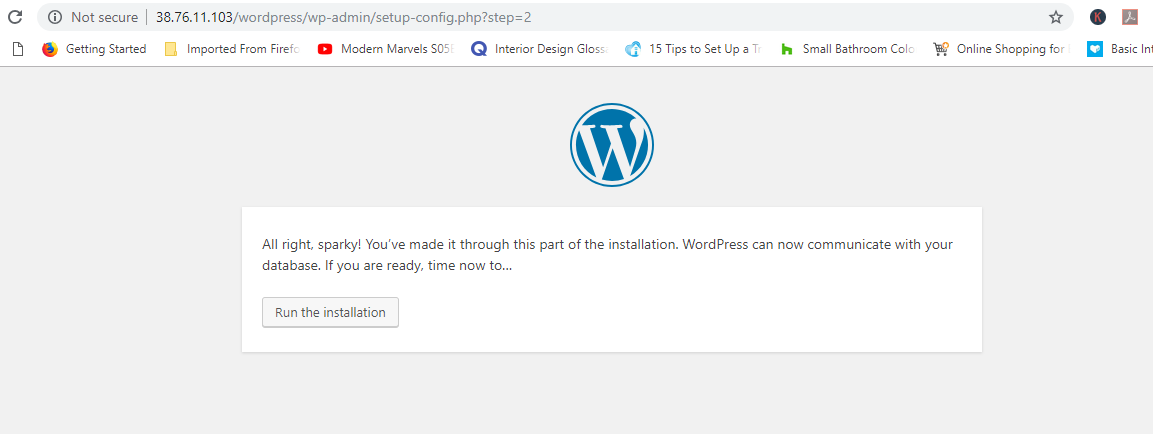
$ mkdir /var/www/html/wordpress/wp-content/uploads

Finally, change permissions of ‘uploads’ directory.

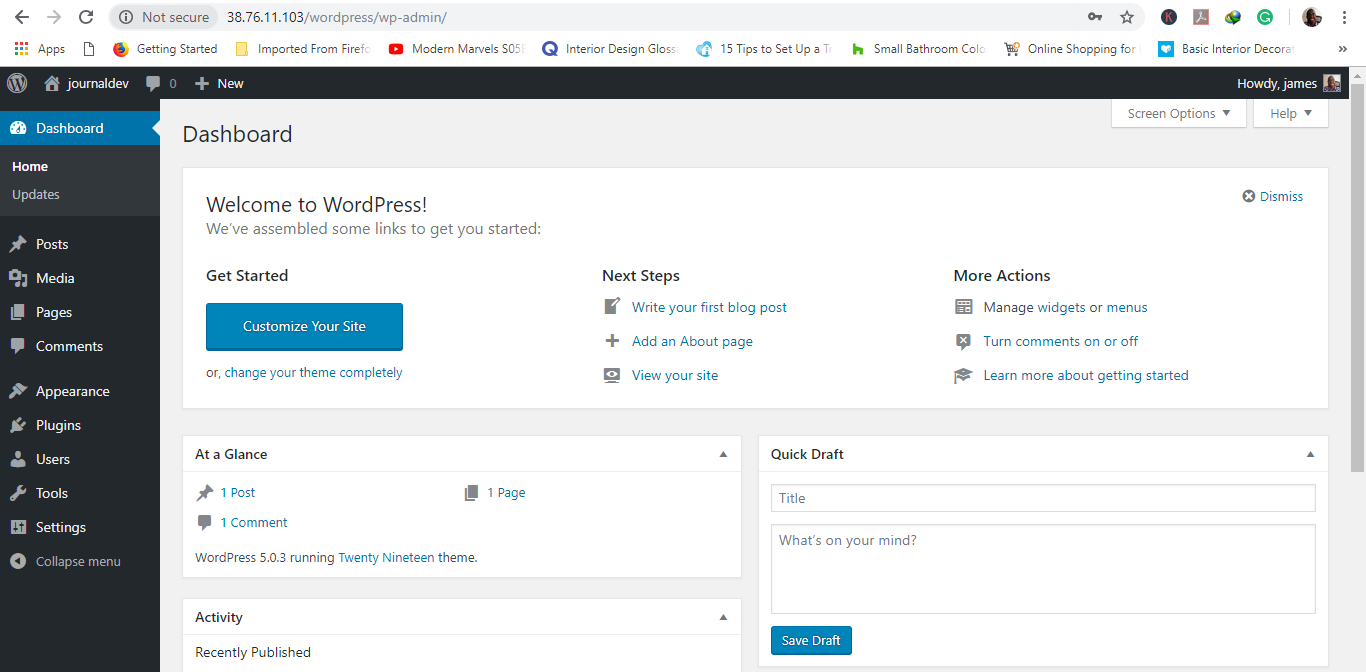
chown -R www-data:www-data /var/www/html/wordpress/wp-content/uploads/

Open your browser and go to the server’s URL. In my case it’s

https://server-ip/wordpress

You’ll be presented with a WordPress wizard and a list of credentials required to successfully set it up.Fill out the form as shown with the credentials specified when creating the WordPress database in the MariaDB database. Leave out the database host and table prefix and Hit ‘Submit’ button.If all the details are correct, you will be given the green light to proceed. Run the installation.Fill out the additional details required such as site title, Username, and Password and save them somewhere safe lest you forget. Ensure to use a strong password.Scroll down and Hit ‘Install WordPress’. If all went well, then you will get a ‘Success’ notification as shown.

Sucess

Click on the ‘Login’ button to get to access the Login page of your fresh WordPress installation.Provide your login credentials and hit ‘Login’.

1. Now we will start the Installation of Phpmyadmin
2. We already completed instllation of php & mysql on that so we only install the phpMyAdmin pakage
3. apt install phpmyadmin
4. Then we will edit one file vim /etc/apache2/apache2.conf
5. then edit Then add the following line to the end of the file:

Include /etc/phpmyadmin/apache.conf

1. systemctl restart apache2.service
2. Now we need to create a admin user who can access the the phpmyadmin page as like administrator

mysql -p -u root

Now you can add a new MySQL user with the username of your choice.

CREATE USER 'USERNAME'@'%' IDENTIFIED BY 'PASSWORD';

example like

CREATE USER root@localhost IDENTIFIED BY 'root\_password';

now need to give the access to created user

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'localhost';

now reload the privileges

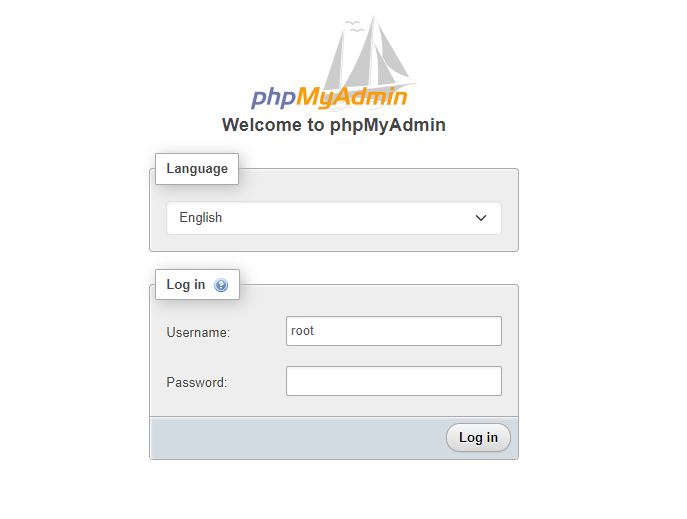
FLUSH PRIVILEGES;

Now we need to restart the apache server and access the phpadmin like

<http://ip_server/phpmyadmin>

then

login the page



**In my case I have create root as a user and password is 1234**

1. **NOW we will test the**
2. **Need to verify which version is running**
3. **php -v or in details we can use** apt show php -a
4. **To see how many php version is working we can user**

update-alternatives --config php

1. **Now add repo to install all required version**

add-apt-repository ppa:ondrej/php

apt install software-properties-common

1. Now we will start the installation of multiple php version with below commands

sudo apt-get install php8.1 php8.1-fpm

sudo apt-get install php8.1-mysql php8.1-mbstring php8.1-xml php8.1-gd php8.1-curl

sudo apt-get install php7.4 php7.4-fpm

sudo apt-get install php7.4-mysql php7.4-mbstring php7.4-xml php7.4-gd php7.4-curl

sudo apt-get install php5.6 php5.6-fpm

sudo apt-get install php5.6-mysql php5.6-mbstring php5.6-xml php5.6-gd php5.6-curl

**for updating libapache we will also install below pakage**

apt-get install -f libapache2-mod-php7.4

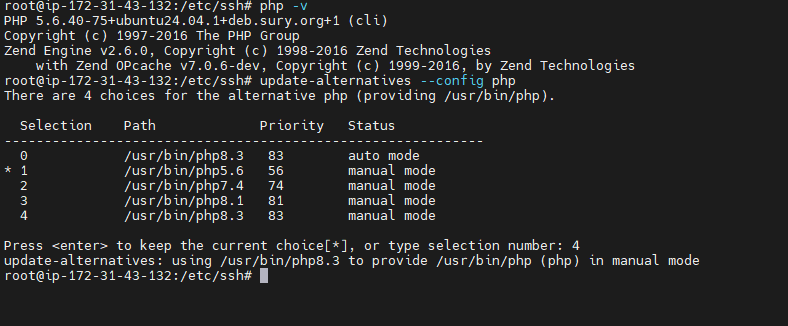
apt-get install -f libapache2-mod-php8.1

now we will understand how to to use another php version by the commands

hit

update-alternatives --config php

then enter php version according to requirement



**Now we need to very again which php version is running by php -v**

**But we will face one issue if we try to check php version on browser it will show older version so**

**We will use 2 command**

**First disable running php version**

a2enmod php8.3

then enable new version

a2enmod php7.4

then restart the apache server by systemctl restart apache2.service