sudo apt-get update

sudo apt-get upgrade -y

sudo apt-get install nginx -y

sudo systemctl start nginx

sudo systemctl enable nginx

sudo vi /etc/nginx/sites-available/default

## Now remove all the text in the file and just paste the below code. By doing this, you won’t ## get any error like me.

server {

        listen 80;

        root /var/www/html;

        index index.php index.html index.htm;

        client\_max\_body\_size 8000M;

location / {

                # try\_files $uri $uri/ =404;

                try\_files $uri $uri/ /index.php?q=$uri&$args;

        }

# Pass PHP requests to PHP\_FPM

        location ~ \.php$ {

                try\_files $uri =404;

                fastcgi\_buffer\_size  256k;

                fastcgi\_buffers  4 256k;

                fastcgi\_busy\_buffers\_size  256k;

                fastcgi\_split\_path\_info ^(.+\.php)(/.+)$;

                fastcgi\_pass unix:/var/run/php/php7.4-fpm.sock;

                fastcgi\_index index.php;

                fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

                include fastcgi\_params;

        }

location = /favicon.ico {

        access\_log off;

        log\_not\_found off;

        expires max;

}

location = /robots.txt {

        access\_log off;

        log\_not\_found off;

}

error\_page 404 /404.html;

        error\_page 500 502 503 504 /50x.html;

        location = /50x.html {

                root /usr/share/nginx/html;

        }

# Deny Access to Hidden Files such as .htaccess

location ~ /\. {

        deny all;

        access\_log off;

        log\_not\_found off;

}

# Disallow PHP files In the WordPress uploads directory

location /wp-content/uploads/ {

        location ~ \.php$ {

                deny all;

        }

}

}

sudo systemctl restart nginx

sudo nginx -t

sudo vi /etc/nginx/nginx.conf

## ADD =====================================================================

user www-data;

worker\_processes auto;

pid /run/nginx.pid;

include /etc/nginx/modules-enabled/\*.conf;

events {

worker\_connections 768;

# multi\_accept on;

}

http {

##

# Basic Settings

##

sendfile on;

tcp\_nopush on;

types\_hash\_max\_size 2048;

# server\_tokens off;

# server\_names\_hash\_bucket\_size 64;

# server\_name\_in\_redirect off;

include /etc/nginx/mime.types;

default\_type application/octet-stream;

##

# SSL Settings

##

ssl\_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE

ssl\_prefer\_server\_ciphers on;

##

# Logging Settings

##

access\_log /var/log/nginx/access.log;

error\_log /var/log/nginx/error.log;

##

# Gzip Settings

##

gzip on;

gzip\_vary on;

gzip\_proxied any;

gzip\_comp\_level 6;

gzip\_buffers 16 8k;

gzip\_http\_version 1.1;

gzip\_types text/plain text/css application/json application/javascript text/xml application/xml application/xml+rss text/javascript;

##

# Virtual Host Configs

##

include /etc/nginx/conf.d/\*.conf;

include /etc/nginx/sites-enabled/\*;

}

#mail {

# # See sample authentication script at:

# # http://wiki.nginx.org/ImapAuthenticateWithApachePhpScript

#

# # auth\_http localhost/auth.php;

# # pop3\_capabilities "TOP" "USER";

# # imap\_capabilities "IMAP4rev1" "UIDPLUS";

#

# server {

# listen localhost:110;

# protocol pop3;

# proxy on;

# }

#

# server {

# listen localhost:143;

# protocol imap;

# proxy on;

# }

#}

cd /etc/nginx >> you are in this file

sudo apt-get install mariadb-server mariadb-client -y

sudo systemctl start mariadb

sudo systemctl enable mariadb

sudo mysql\_secure\_installation<<EOF

1. n
2. y
3. newpassword : access\_log off;
4. reenter newpassword : access\_log off;
5. y
6. y
7. y

sudo apt-get install php php-fpm php-curl php-cli php-zip php-mysql php-xml php-mbstring php-gd php-xmlrpc php-imagick php-intl php-soap

cd /var/www/html

sudo wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz

sudo mkdir phpMyAdmin

sudo tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1

sudo rm phpMyAdmin-latest-all-languages.tar.gz

sudo wget https://wordpress.org/latest.tar.gz

sudo tar -xzf latest.tar.gz

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

ls -a

cd ..

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

cd ~

sudo systemctl restart nginx

## LOGIN TO PHP ======================================================== checking

User = root

Password = access\_log off;

======================================================== checking

Wordpress login

Database name =demo

User = root

Password = access\_log off;

Database host == localhost

Table prefic = wp\_

Update configartion ==================================

cd /etc/php/\*/fpm

sudo vi php.ini

update ==========================

max\_execution\_time = 300

max\_input\_ time =180

max\_input\_vars = 10000

memory\_limit = 256M

post \_max\_size = 8000M

upload\_ max\_filesize= 200

sudo systemctl restart php\*-fpm

cd ~

sudo systemctl restart nginx

sudo nginx –t

sudo rm /var/www/html/index.html

sudo vi /var/www/html/wp-config.php

## ===============================================

<?php

if (isset($\_SERVER['HTTP\_X\_FORWARDED\_PROTO']) && $\_SERVER['HTTP\_X\_FORWARDED\_PROTO'] === 'https') {$\_SERVER['HTTPS'] = 'on';}

/\*\*

\* 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', 'demo' );

/\*\* Database username \*/

define( 'DB\_USER', 'root' );

/\*\* Database password \*/

define( 'DB\_PASSWORD', 'access\_log\_off;' );

/\*\* 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', '' );

/\*\*REDIS HOST \*/

define( 'WP\_REDIS\_HOST', ' ' );

define( 'WP\_CASHE', true);

/\*\*#@+

\* Authentication unique keys and salts.

\*

\* Change these to different unique phrases! You can generate these using

\* the {@link https://api.wordpress.org/secret-key/1.1/salt/ WordPress.org secret-key service}.

\*

\* You can change these at any point in time to invalidate all existing cookies.

\* This will force all users to have to log in again.

\*

\* @since 2.6.0

\*/

define( 'AUTH\_KEY', 'S}UgYcS.Ts~=9EGuu0kZr\*ASjH0c5M#VAnnws0cFYg0bNVWbWCYxj6d>FQvm ep+' );

define( 'SECURE\_AUTH\_KEY', '{V7:[1y\_[?s@=/a:M>247ISE=5@&Pocc:7Kp:D5i`g-mg\*V5)5rMc;Sn<wHQh3zQ' );

define( 'LOGGED\_IN\_KEY', 'O|cvA7uud!+=hC% $}d;y|]S\*oxb3lIW~rFbN:}#l+wGaWPqH%QSHNUipUVc|,gr' );

define( 'NONCE\_KEY', 'k8A?6-<riDpN.9/-QZJH}2|?&Uz n26b%x1;-GCSzNGJx`I3\*B!|0[nZ3|;[<v/L' );

define( 'AUTH\_SALT', '?bZ#ENW `XqkH,W<OC,z8JG4`Z,x?Z2#.EWkewY%J4Vr=u:Sy\_a?^(:<x%|e`|ri' );

define( 'SECURE\_AUTH\_SALT', 'J8C`b6F0%em1~NU1?YV1I9b7tJN[i.faa~x=l}L0-yX4;1ADT\_LXAl3oev~QSpCA' );

define( 'LOGGED\_IN\_SALT', ']!LP\*yb=<)Sd!\_]>a73xzmFvhgf+3\_7c!-eIyld[DR9/)(h7am[|z- `&RG8q}[B' );

define( 'NONCE\_SALT', 'aBDa:{r/?Suck5s9wa~UVp8sMvI\_Te4eQ?`Jse&NIT1Sg:2k\_)RPOB9EToGZ.x B' );

/\*\*#@-\*/

/\*\*

\* WordPress database table prefix.

\*

\* You can have multiple installations in one database if you give each

\* a unique prefix. Only numbers, letters, and underscores please!

\*/

$table\_prefix = 'wp\_';

/\*\*

\* For developers: WordPress debugging mode.

\*

\* Change this to true to enable the display of notices during development.

\* It is strongly recommended that plugin and theme developers use WP\_DEBUG

\* in their development environments.

\*

\* For information on other constants that can be used for debugging,

\* visit the documentation.

\*

\* @link https://wordpress.org/documentation/article/debugging-in-wordpress/

\*/

define( 'WP\_DEBUG', false );

/\* Add any custom values between this line and the "stop editing" line. \*/

/\* That's all, stop editing! Happy publishing. \*/

/\*\* Absolute path to the WordPress directory. \*/

if ( ! defined( 'ABSPATH' ) ) {

define( 'ABSPATH', \_\_DIR\_\_ . '/' );

}

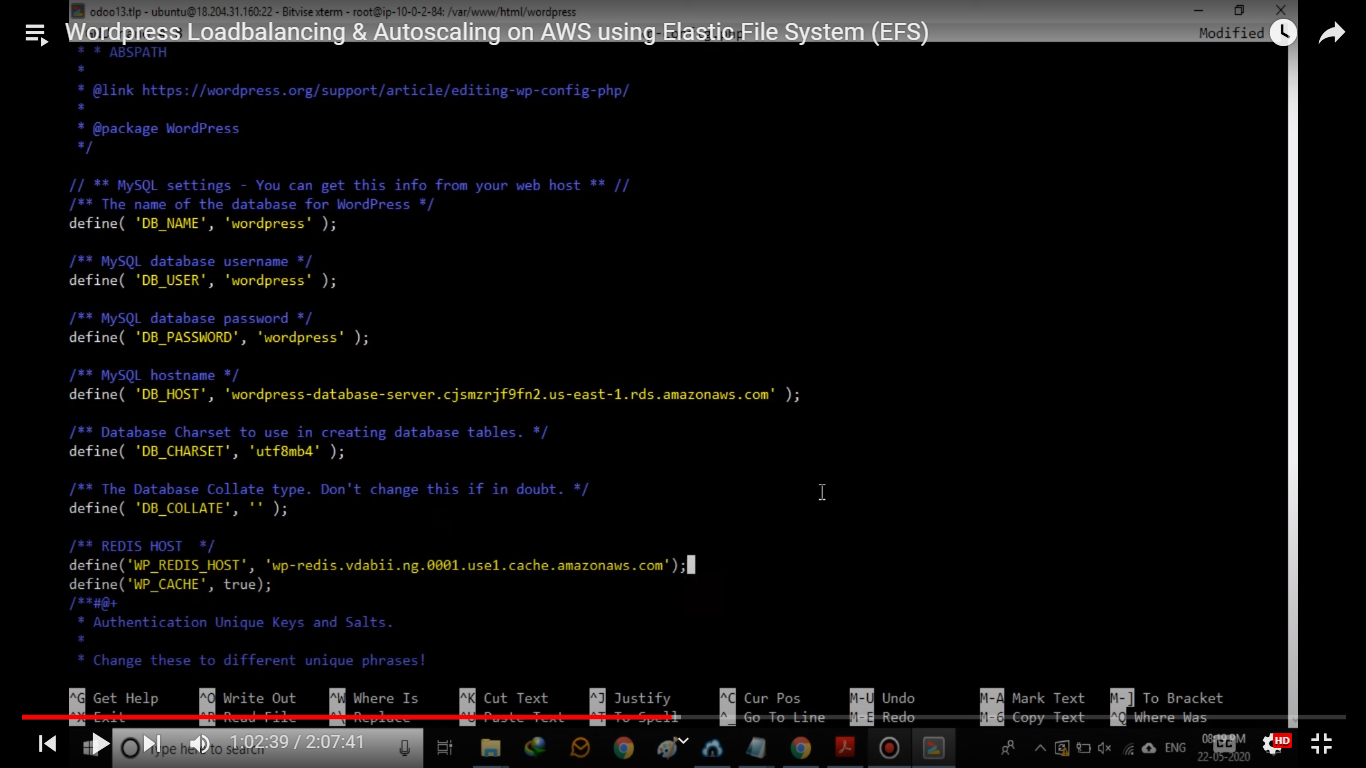
/\*\* Sets up WordPress vars and included files. \*/

require\_once ABSPATH . 'wp-settings.php';

## ===================================================

sudo systemctl restart nginx

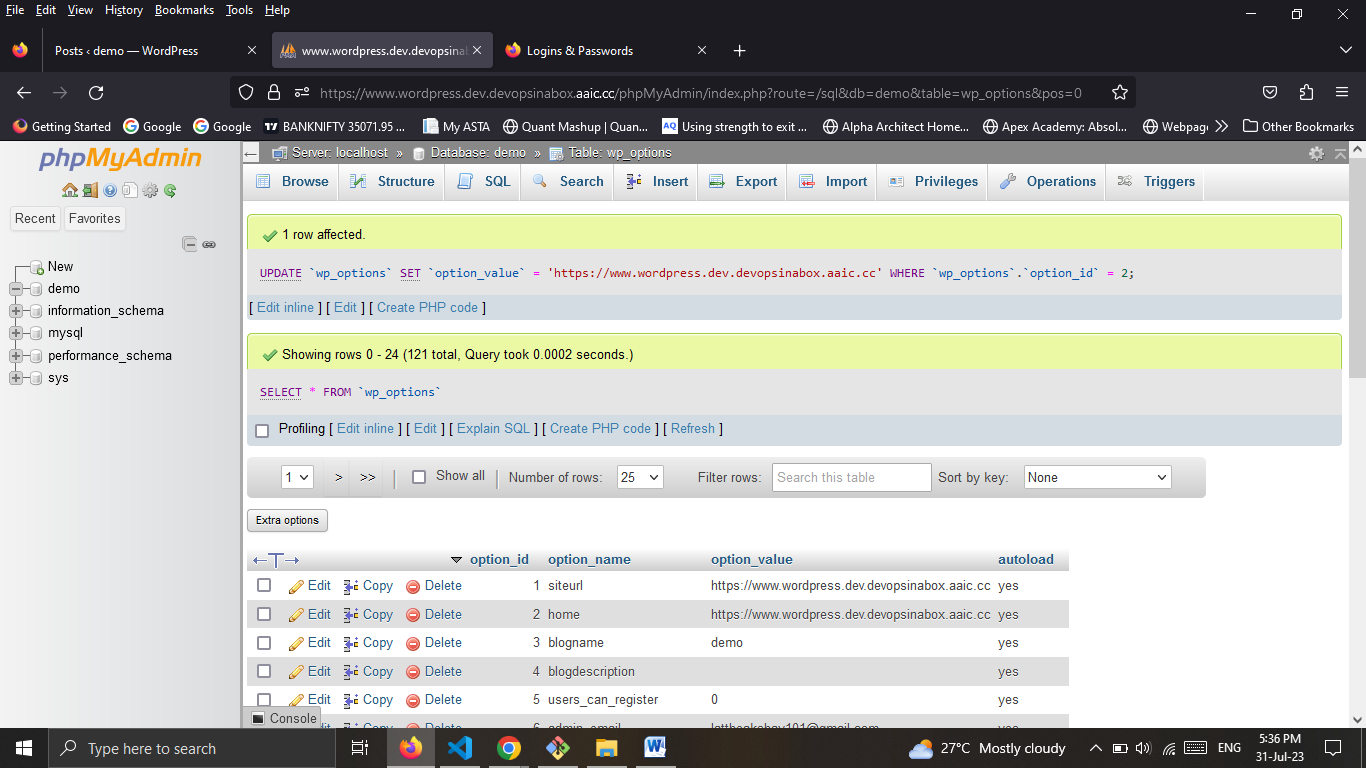
echo "Installation is done."

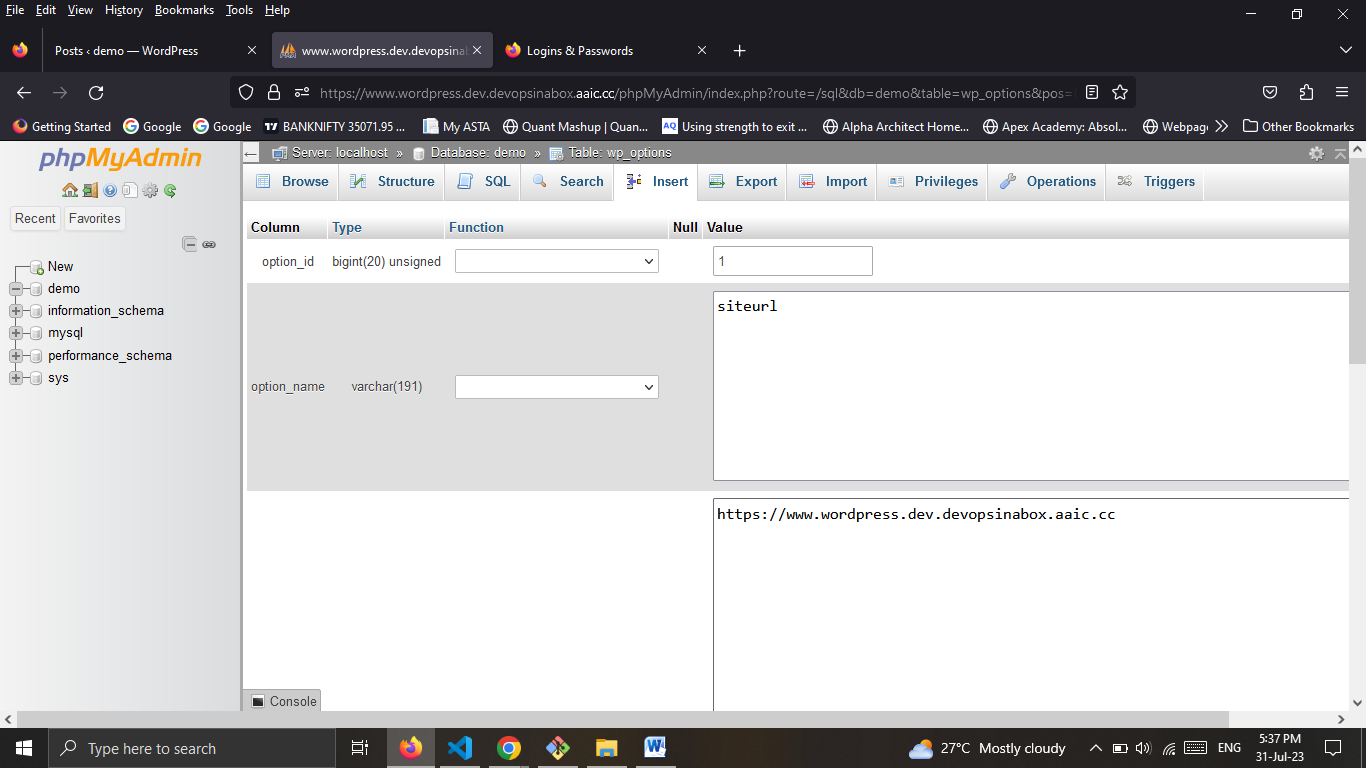


/\*\*REDIS HOST \*/

define( 'WP\_REDIS\_HOST', ' ' );

define( 'WP\_CASHE', true);





Refreance 1) https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#TargetGroups:

Refreance 2) <https://www.youtube.com/watch?v=4jTuQYMbe1s&t=528s>

sudo bash -c 'cat > /etc/nginx/sites-available/default' << EOF

server {

listen 80;

root /var/www/html;

index index.php index.html index.htm;

client\_max\_body\_size 8000M;

location / {

try\_files \$uri \$uri/ /index.php?q=\$uri&\$args;

}

location ~ \.php$ {

try\_files \$uri =404;

fastcgi\_buffer\_size 256k;

fastcgi\_buffers 4 256k;

fastcgi\_busy\_buffers\_size 256k;

fastcgi\_split\_path\_info ^(.+\.php)(/.+)$;

fastcgi\_pass unix:/var/run/php/php7.4-fpm.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME \$document\_root\$fastcgi\_script\_name;

include fastcgi\_params;

}

location = /favicon.ico {

access\_log off;

log\_not\_found off;

expires max;

}

location = /robots.txt {

access\_log off;

log\_not\_found off;

}

error\_page 404 /404.html;

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root /usr/share/nginx/html;

}

location ~ /\. {

deny all;

access\_log off;

log\_not\_found off;

}

location /wp-content/uploads/ {

location ~ \.php$ {

deny all;

}

}

}

EOF

=================================================================================

sudo sed -n -i '/gzip on;/,$!p' /etc/nginx/nginx.conf

sudo tee -a /etc/nginx/nginx.conf > /dev/null <<EOL

gzip on;

gzip\_vary on;

gzip\_proxied any;

gzip\_comp\_level 6;

gzip\_buffers 16 8k;

gzip\_http\_version 1.1;

gzip\_types text/plain text/css application/json application/javascript text/xml application/xml application/xml+rss text/javascript;

}

EOL

# Restart nginx

sudo systemctl restart nginx

My sql paswords newpassword : access\_log off;

WordOps

3q5JmpziYpIiSbNxIfADtrCr

akshay

[jNRrA1nrq0agJ1U0580xi5wu]