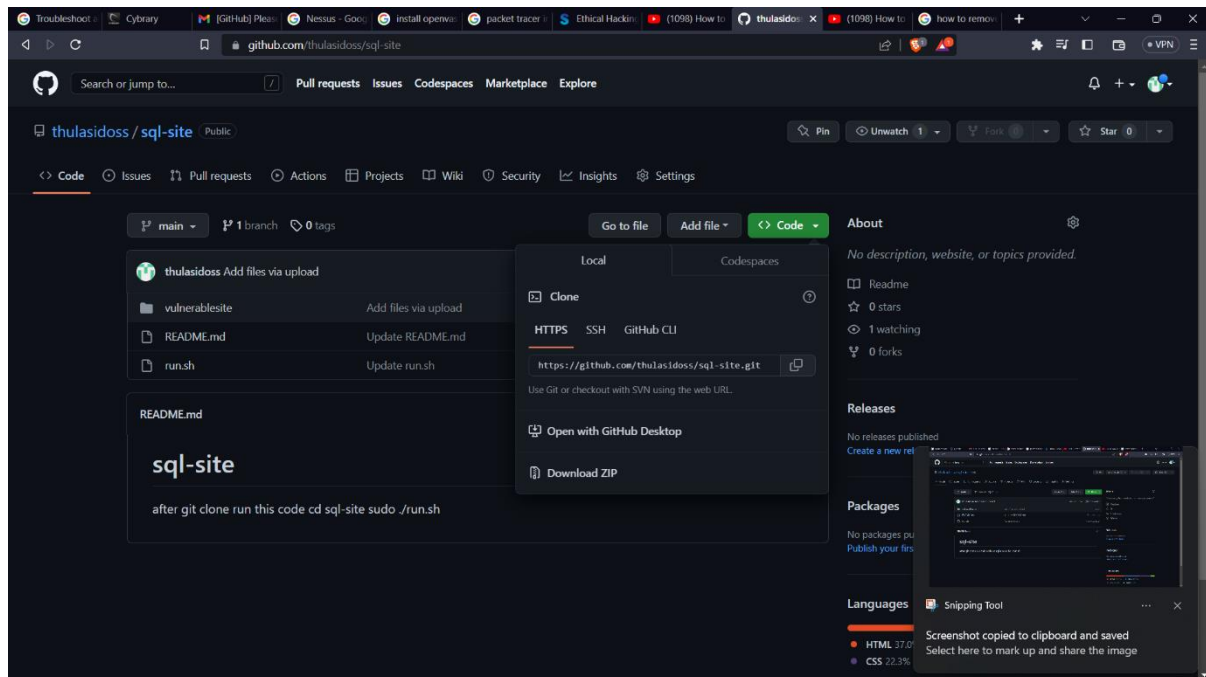


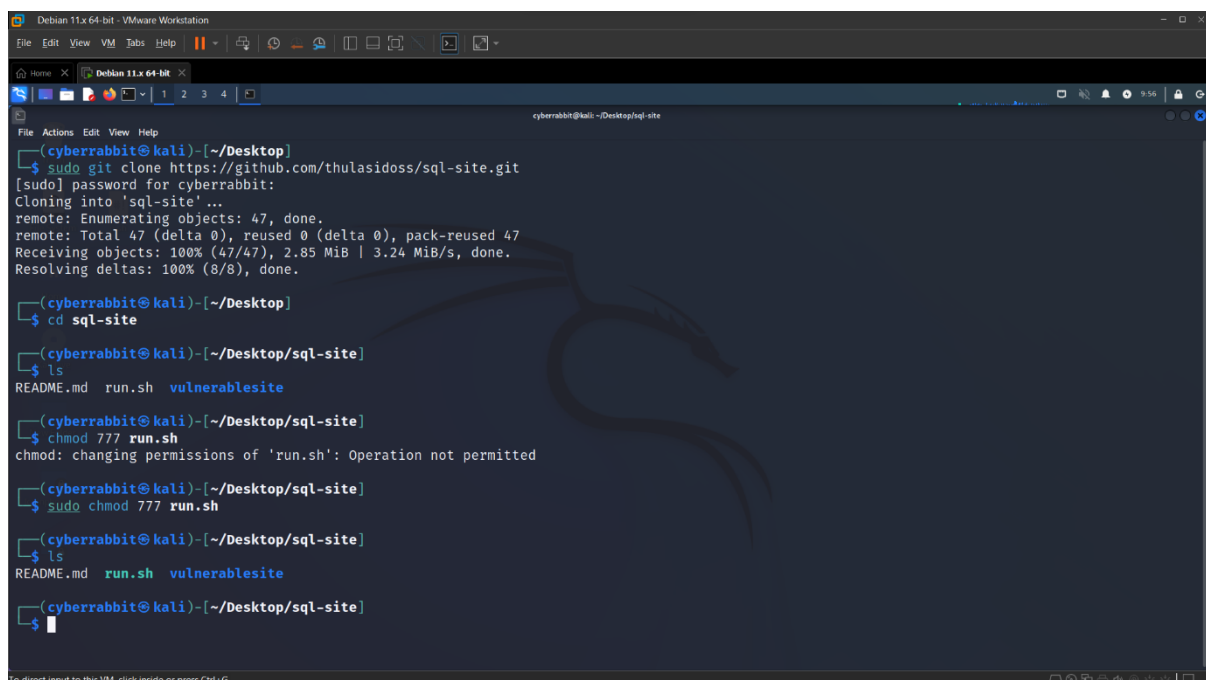
HOW TO SETUP VULNERABLESITE TO CORRESPONDING PLACE:

STEP 1: open terminal and type `sudo git clone https://github.com/thulasidoss/sql-site.git`

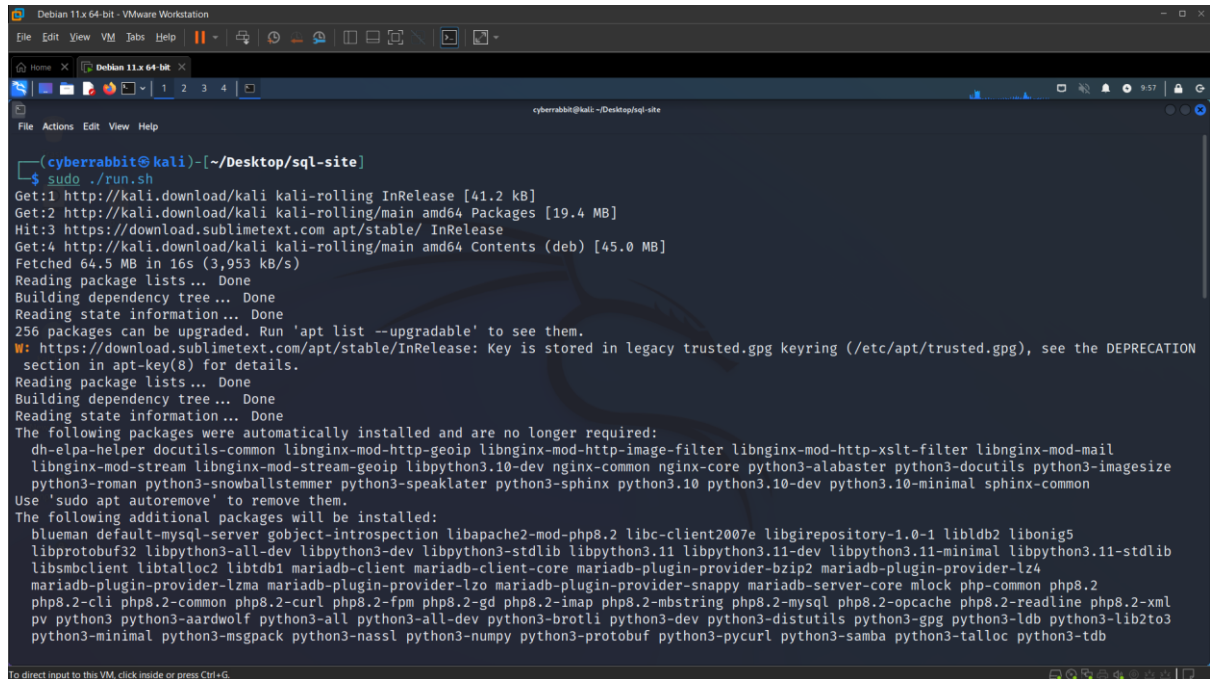


STEP 2:

Just follow the below steps in image



STEP 3: run sudo ./run.sh interterminal



The screenshot shows a terminal window titled "Debian 11.x 64-bit - VMware Workstation". The user is logged in as "cyberrabbit@kali" in the directory "~/Desktop/sql-site". The command "sudo ./run.sh" has been executed, resulting in the following output:

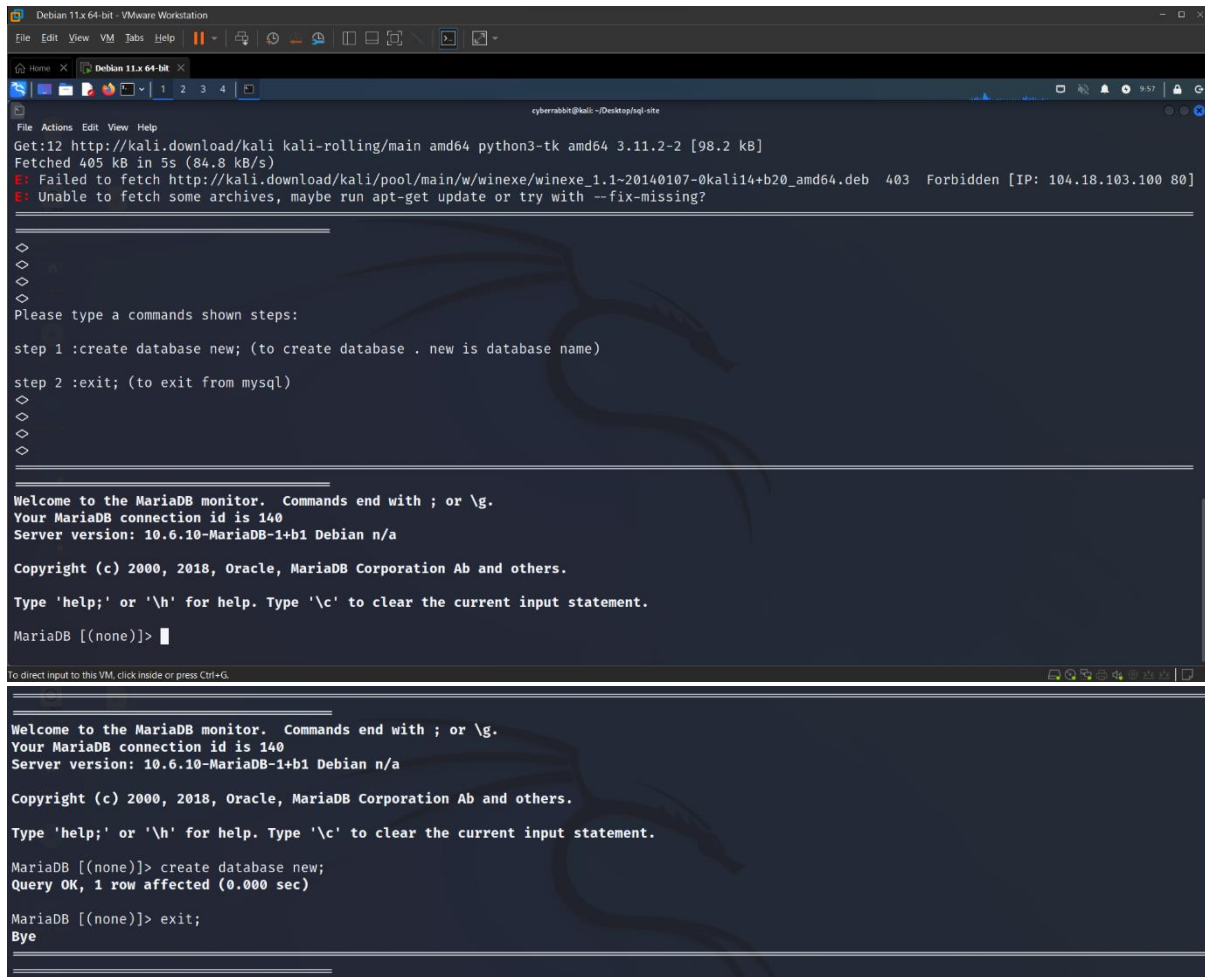
```
(cyberrabbit@kali)-[~/Desktop/sql-site]
$ sudo ./run.sh
Get:1 http://kali.download/kali kali-rolling InRelease [41.2 kB]
Get:2 http://kali.download/kali kali-rolling/main amd64 Packages [19.4 MB]
Hit:3 https://download.sublimetext.com apt/stable/ InRelease
Get:4 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [45.0 MB]
Fetched 64.5 MB in 16s (3,953 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
256 packages can be upgraded. Run 'apt list --upgradable' to see them.
W: https://download.sublimetext.com/apt/stable/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION
section in apt-key(8) for details.
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
dh-elpa-helper docutils-common libnginx-mod-http-geoip libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail
libnginx-mod-stream libnginx-mod-stream-geoip libpython3.10-dev nginx-common nginx-core python3-alabaster python3-docutils python3-imagesize
python3-roman python3-snowballstemmer python3-speaklater python3-sphinx python3.10 python3.10-dev python3.10-minimal sphinx-common
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
blueman default-mysql-server gobject-introspection libapache2-mod-php8.2 libc-client2007e libgirepository-1.0-1 libldb2 libonig5
libprotobuf32 libpython3-all-dev libpython3-dev libpython3-stdlib libpython3.11 libpython3.11-dev libpython3.11-minimal libpython3.11-stdlib
libsmclient libtalloc2 libtdb1 mariadb-client mariadb-client-core mariadb-plugin-provider-bzip2 mariadb-plugin-provider-lz4
mariadb-plugin-provider-lzma mariadb-plugin-provider-lzo mariadb-plugin-provider-snappy mariadb-server-core mlock php-common php8.2
php8.2-cli php8.2-common php8.2-curl php8.2-fpm php8.2-gd php8.2-imap php8.2-mbstring php8.2-mysql php8.2-opcache php8.2-readline php8.2-xml
pv python3 python3-aardwolf python3-all python3-all-dev python3-brotli python3-dev python3-distutils python3-gpg python3-ldb python3-lib2to3
python3-minimal python3-msgpack python3-nassl python3-numpy python3-protobuf python3-pycurl python3-samba python3-talloc python3-tdb
```

At the bottom of the terminal window, there is a status bar that reads: "To direct input to this VM, click inside or press Ctrl+G."

STEP 4:now type

MariaDB[(none)]>create database new;

MariaDB[(none)]>exit;



```
Debian 11.x 64-bit - VMware Workstation
File Edit View VM Tabs Help
Debian 11.x 64-bit
cyberrabbit@kali: ~/Desktop/cgi-site
File Actions Edit View Help
Get:12 http://kali.download/kali kali-rolling/main amd64 python3-tk amd64 3.11.2-2 [98.2 kB]
Fetched 405 kB in 5s (84.8 kB/s)
E: Failed to fetch http://kali.download/kali/pool/main/w/winexe/winexe_1.1~20140107-0kali14+b20_amd64.deb 403 Forbidden [IP: 104.18.103.100 80]
E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?

Please type a commands shown steps:

step 1 :create database new; (to create database . new is database name)

step 2 :exit; (to exit from mysql)

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 140
Server version: 10.6.10-MariaDB-1+b1 Debian n/a

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)]>

Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 140
Server version: 10.6.10-MariaDB-1+b1 Debian n/a

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 new;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> exit;
Bye
```

STEP 5:

Follow the steps in below image

Give y press enter

Give y press enter

Set password for database

Re-enter password

Give y press enter

Give y press enter

Enter database password

```
>
>
>
>
step 1 : Setup a Password and Remember
step 2 : Enter "Yes" to All
step 3 : now smmae localhost to 127.0.0.1 in connect.php
step 4 : now Change the password on connect.php which is you previesly setup password
step 5 : and now change same things in register.php
>
>
>
```

```
Debian 11.x 64-bit - VMware Workstation
File Edit View VM Tabs Help
Debian 11.x 64-bit
cyberabhi@kali: ~/Desktop/leg-ote
File Actions Edit View Help
Switch to unix_socket authentication [Y/n] y
Enabled successfully!
Reloading privilege tables..
... Success!

You already have your root account protected, so you can safely answer 'n'.

Change the root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into 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] y
... Success!

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] y
... Success!

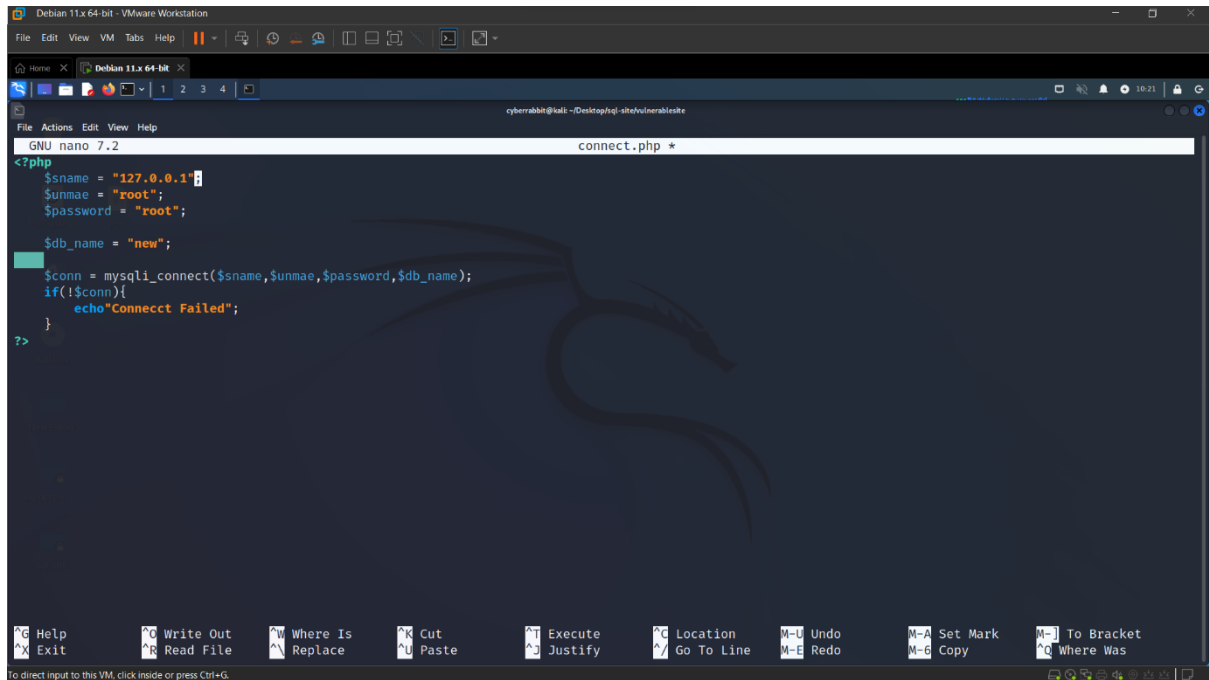
By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
```

STEP 6:now on window will open which is connect.php in that

Change \$sname="127.0.0.1"

Change \$password="enter password you create for database"

Save it and exit



```
<?php
$sname = "127.0.0.1";
$unmae = "root";
$password = "root";

$db_name = "new";

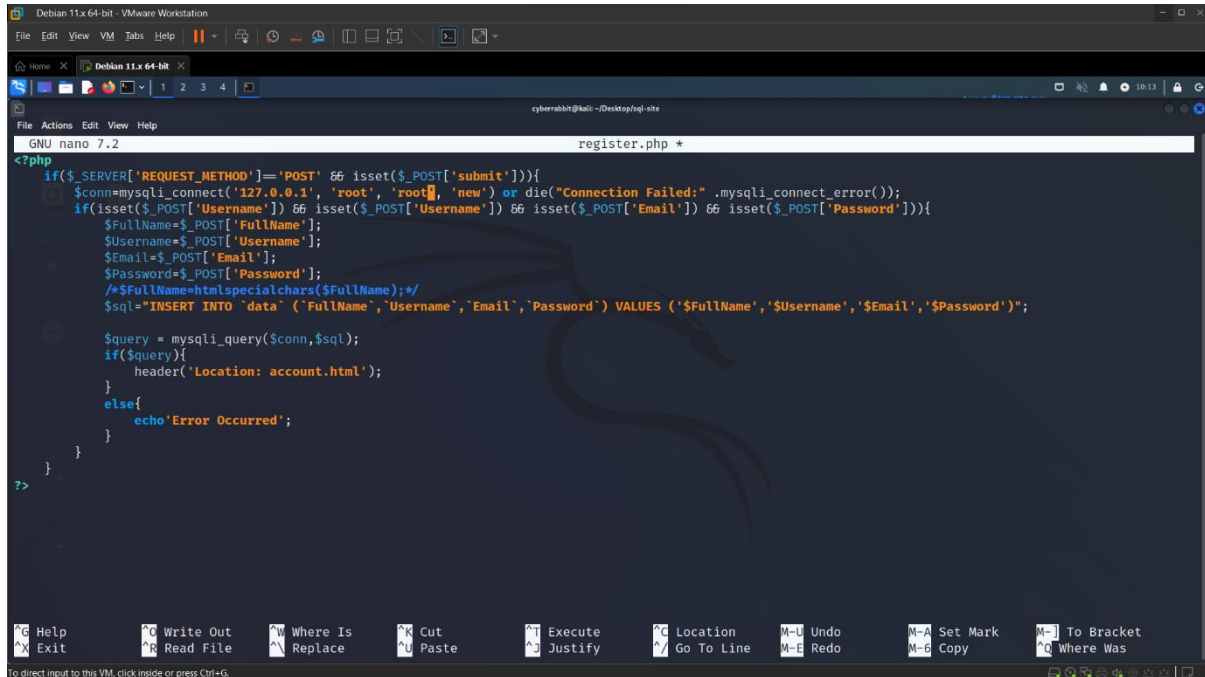
$conn = mysqli_connect($sname,$unmae,$password,$db_name);
if(!$conn){
    echo"Connecct Failed";
}
?>
```

STEP 7: now on window will open which is register.php in that

Change \$sname="127.0.0.1"

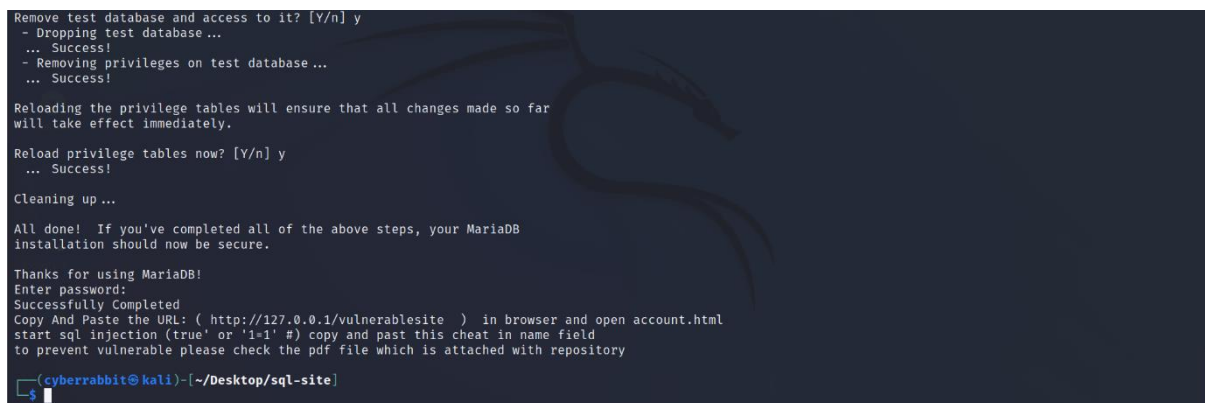
Change \$password=" enter password you create for database"

Save it and exit



```
Debian 11.x 64-bit - VMware Workstation
File Edit View VM Tabs Help
Debian 11.x 64-bit
cyberrabbit@kali: ~/Desktop/sql-site
GNU nano 7.2 register.php *
<?php
if($_SERVER['REQUEST_METHOD']=='POST' && isset($_POST['submit'])){
    $conn=mysqli_connect('127.0.0.1', 'root', 'root', 'new') or die("Connection Failed:".mysql_connect_error());
    if(isset($_POST['Username']) && isset($_POST['Email']) && isset($_POST['Password'])){
        $FullName=$_POST['FullName'];
        $Username=$_POST['Username'];
        $Email=$_POST['Email'];
        $Password=$_POST['Password'];
        /*$FullName=htmlspecialchars($FullName);*/
        $sql="INSERT INTO `data` (`FullName`,`Username`,`Email`,`Password`) VALUES ('$FullName','$Username','$Email','$Password')";

        $query = mysqli_query($conn,$sql);
        if($query){
            header('Location: account.html');
        }
        else{
            echo'Error Occurred';
        }
    }
}
?>
```



```
Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] y
... Success!

Cleaning up ...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
Enter password:
Successfully Completed
Copy And Paste the URL: ( http://127.0.0.1/vulnerablesite ) in browser and open account.html
start sql injection (true' or '1=1' #) copy and past this cheat in name field
to prevent vulnerable please check the pdf file which is attached with repository

cyberrabbit@kali)-[~/Desktop/sql-site]
-$
```












STEP 8:

Now open the <http://127.0.0.1/vulnerablesite>

Open account.html

And start attacking

Index of /vulnerablesite

<u>Name</u>	<u>Last modified</u>	<u>Size</u>	<u>Description</u>
 Parent Directory		-	
 account.html	2023-03-21 01:33	5.7K	
 connect.php	2023-03-21 03:38	229	
 image/	2023-03-14 12:32	-	
 login.php	2023-03-21 09:40	2.3K	
 logout.php	2023-03-21 09:15	104	
 new.sql	2023-03-21 03:32	1.8K	
 reg.php	2023-03-21 09:23	4.6K	
 register.php	2023-03-21 02:04	912	
 retry.html	2023-03-20 17:41	3.4K	
 style.css	2023-03-08 22:55	5.5K	

Apache/2.4.54 (Win64) OpenSSL/1.1.1p PHP/8.2.0 Server at localhost Port 80

Check pdf to prevent vulnerable:

SESSION_HIJACKING.pdf

XSS.pdf

SQL-INJECTION.pdf

In sql-site