

# Setting up the Lamp stack and website

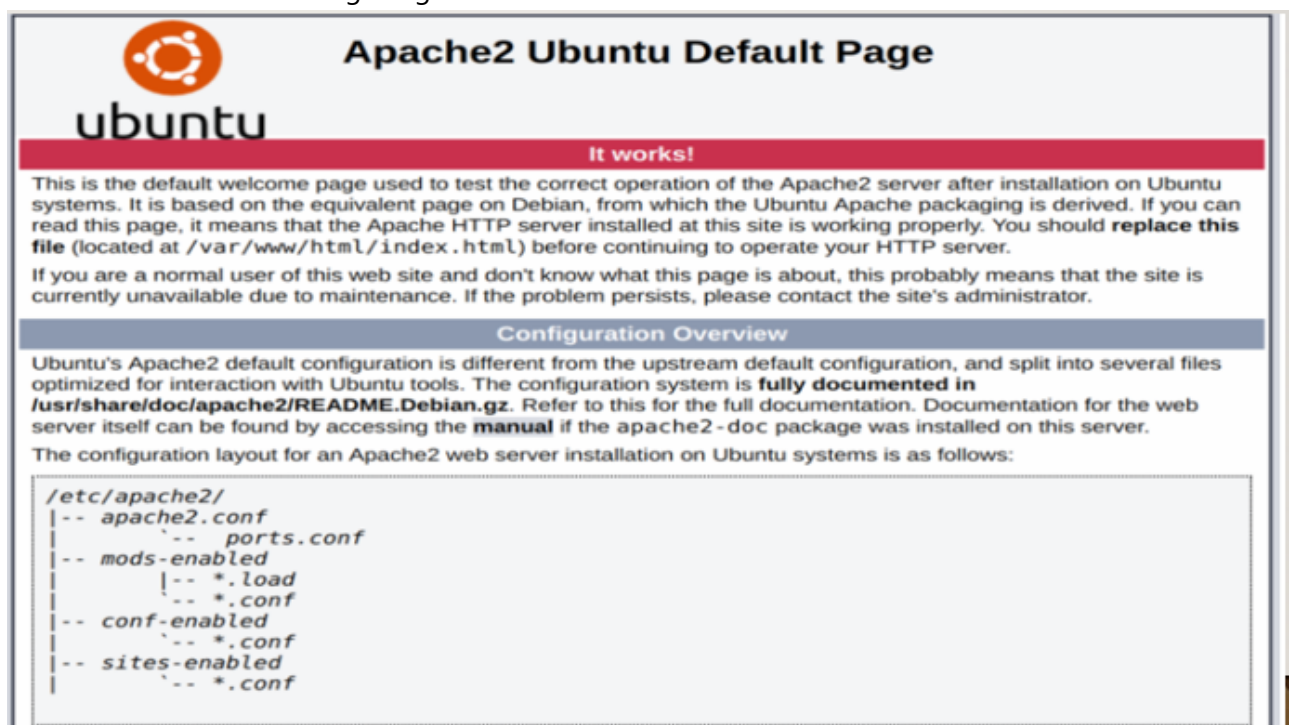
## 1. Setting up the Apache server

- Download and install apache2 from ubuntu
  1. `sudo apt update`
  2. `sudo apt install apache2`
- Change firewall settings to allow traffic
  1. `sudo ufw app list`
  2. `sudo ufw app info "Apache Full"`
  3. `sudo ufw allow in "Apache Full"`
- See if you can access this server by putting the following in your browser : `http://server_ip`

where server\_ip is your ip\_address of server machine, get this using ifconfig in your terminal and copying the ipv4 address

Note: You can also directly access the webpage by using : `http://localhost`

- You should see the following image:



## 2. Setting up MySql server

- Download and install mysql: `sudo apt install mysql-server`
- Set up secure installation.

if you try to run secure installation directly, you might encounter error related to password for root, as such refer to the steps below first.

1. `sudo mysql`-logs in as root
2. inside mysql prompt run, `ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password by 'mynewpassword'` replacing mynewpassword with what you

want

3. Now run `sudo mysql_secure_installation`

- Login to mysql server now using: `sudo mysql -u root -p` followed by entering the password you set up above.

### 3. Installing PHP

- Download php compatible with apache2 and mysql  
`sudo apt install php libapache2-mod-php php-mysql`
- Check if installation was successful by running: `php -v`
- You should see the following:

```
malhar@LAPTOP-4011SI2M:~$ php -v
PHP 7.4.3 (cli) (built: Jun 13 2022 13:43:30) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
    with Zend OPcache v7.4.3, Copyright (c), by Zend Technologies
```

- Make .php pages have higher priority over .html pages for the apache server
  1. `sudo nano /etc/apache2/mods-enabled/dir.conf`
  2. This file looks as follows

```
<IfModule mod_dir.c>
DirectoryIndex index.html index.cgi index.pl index.php index.xhtml index.htm
</IfModule>
```

3. Modify this as follows

```
<IfModule mod_dir.c>
DirectoryIndex index.php index.html index.cgi index.pl index.php index.xhtml
index.htm
</IfModule>
```

### 4. Create a virtual host for your website

- First navigate to `/var/www` and create a new folder named your domain and add `index.html` to it.
  1. `sudo mkdir /var/www/your_domain`
  2. `sudo chown -R $USER:$USER /var/www/your_domain`
  3. `sudo chmod -R 755 /var/www/your_domain`
  4. `sudo nano /var/www/your_domain/index.html`
  5. Inside the html file add the following :

```
<html>
  <head>
    <title>your_domain website</title>
  </head>
```

```
<body>
  <h1>Hello World!</h1>
  <p>This is the landing page of <strong>your_domain</strong>.</p>
</body>
</html>
```

- Navigate to '/etc/apache2/sites-available/' and add a new configuration for this domain
  1. `sudo nano /etc/apache2/sites-available/your_domain.conf`
  2. Inside this file, add the following :

```
<VirtualHost *:80>
ServerAdmin webmaster@localhost
ServerName your_domain
ServerAlias www.your_domain
DocumentRoot /var/www/your_domain
ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

- Enable this configuration
  1. `sudo a2ensite your_domain.conf`
  2. `sudo a2dissite 000-default.conf`
  3. `sudo apache2ctl configtest`
  4. `sudo systemctl restart apache2`

Congratulations!!! The lamp stack is now setup, now lets setup our website

## 1. Create a mysql user and a database

- First open mysql console using the root as mentioned in step 2 of setup.
- Create new user as :

```
CREATE USER 'teach'@ '%' IDENTIFIED WITH mysql_native_password BY 'P@ssw0rd';
```

This is the user I have used for my website, if you want to change the user, please make corresponding changes in the **connect.php** file.

- Create new database and give this user permissions as :
  1. `CREATE DATABASE student_marking;`
  2. `GRANT ALL ON student_marking.* TO 'teach'@ '%';`

This is the database I have used for my website, if you want to change the database, please make the corresponding changes in the **connect.php** file

3. `exit`

- Create teacher and student tables :
  1. login as teach user : `mysql -u teach -p`
  2. `CREATE TABLE student_marking.teacher_id(name VARCHAR(255),identity int,PRIMARY_KEY(identity))`

3. Add few entries to this table using **INSERT INTO** commands, the table will look as follows :

```
mysql> select * from student_marking.teacher_id
-> ;
```

name	identity
teacher2	9876
teacher1	12345

4. Now create another empty table for storing student marks : **CREATE TABLE**  
**student\_marking.marks(roll\_no INT,name VARCHAR(255),marks INT,**  
**PRIMARY\_KEY(roll\_no))**

These are the tables used in my website, if one wants to use other tables, please make the corresponding changes in the **add.php, update.php, index.php and marks\_page.php** files.

## 2. Get the website on the server

- Add the index.php,marks\_page.php,add.php,update.php,delete.php,connect.php files and the css folder to the /var/www/your\_domain folder, this would look as follows:

```
malhar@LAPTOP-4011SI2M:/var/www/malhar_lamp_stack$ ls
add.php  connect.php  css  delete.php  index.html  index.php  marks_page.php  readme.md  readme_images  update.php
```

- To run the website, simply start apache2 and mysql services and access the server using your browser.

Now that we have setup the website, lets see it in action

## 1. The login/validation page :

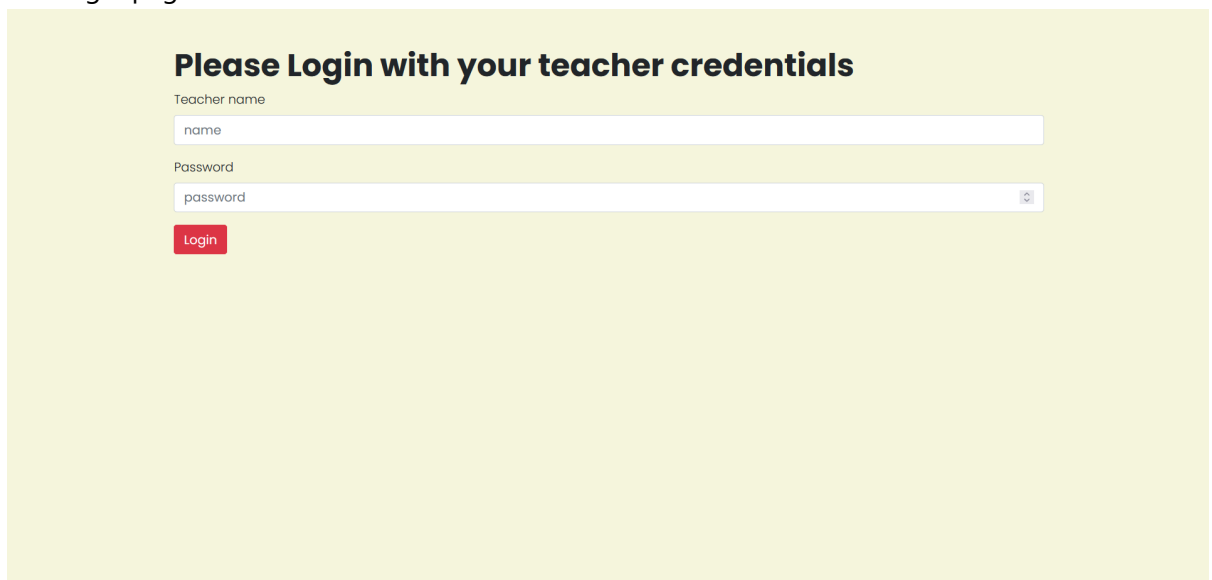
- On the login page, you're asked to enter the teacher name and password for it, this information is verified against the entries of the database and action is taken accordingly. Let us see this in action.

1. Our table :

```
mysql> select * from student_marking.teacher_id
-> ;
```

name	identity
teacher2	9876
teacher1	12345

## 2. Our login page :



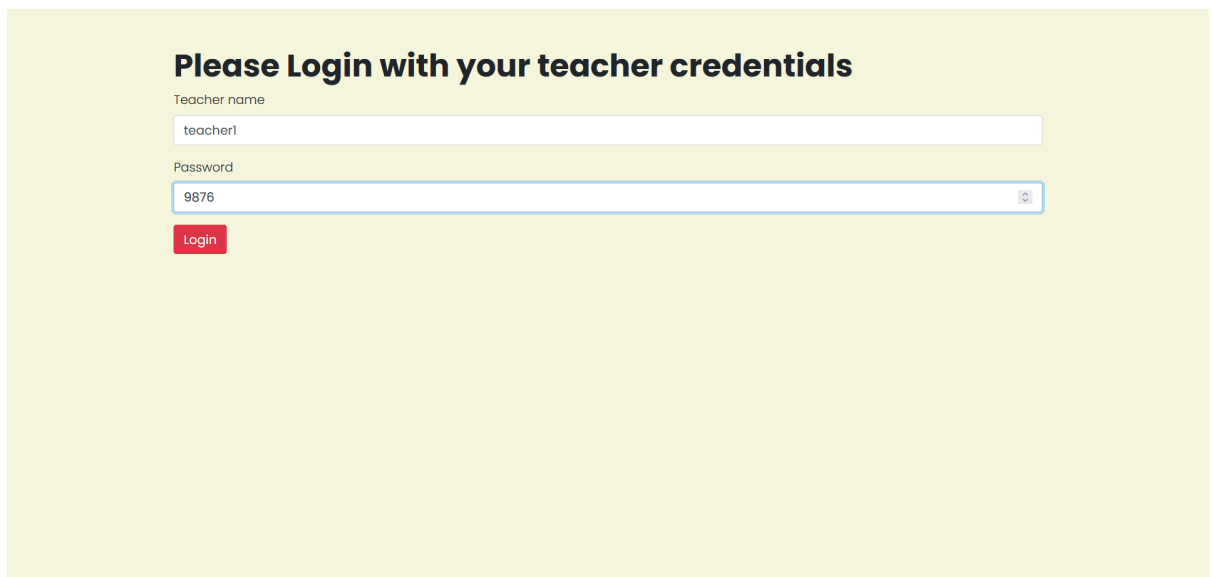
**Please Login with your teacher credentials**

Teacher name

Password

Login

## 3. Incorrect details :



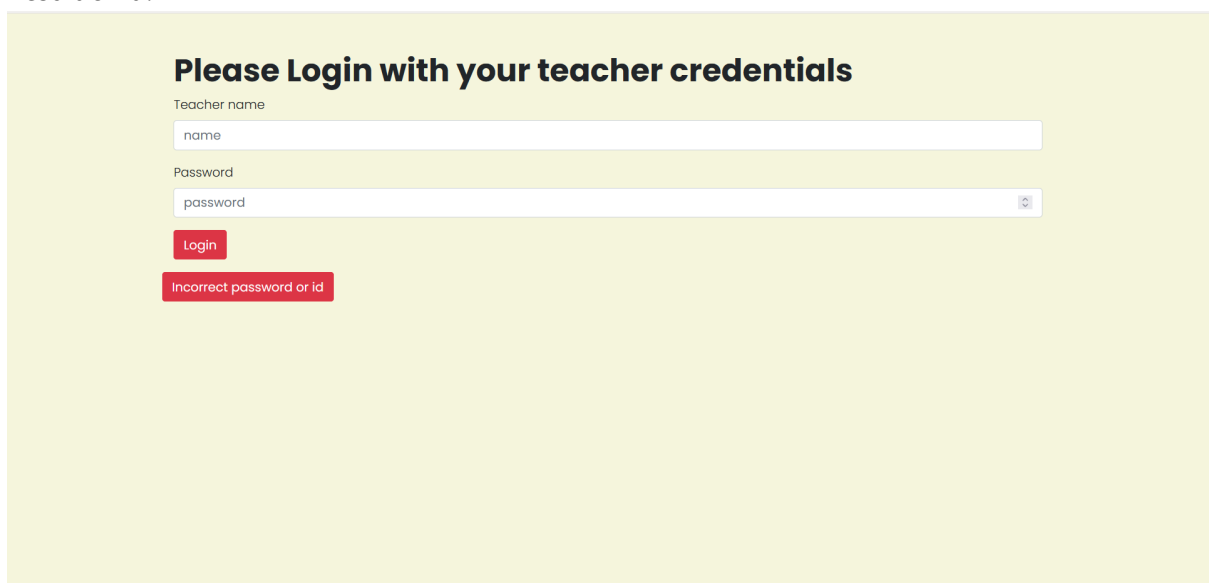
**Please Login with your teacher credentials**

Teacher name

Password

Login

## 4. Result of it :



**Please Login with your teacher credentials**

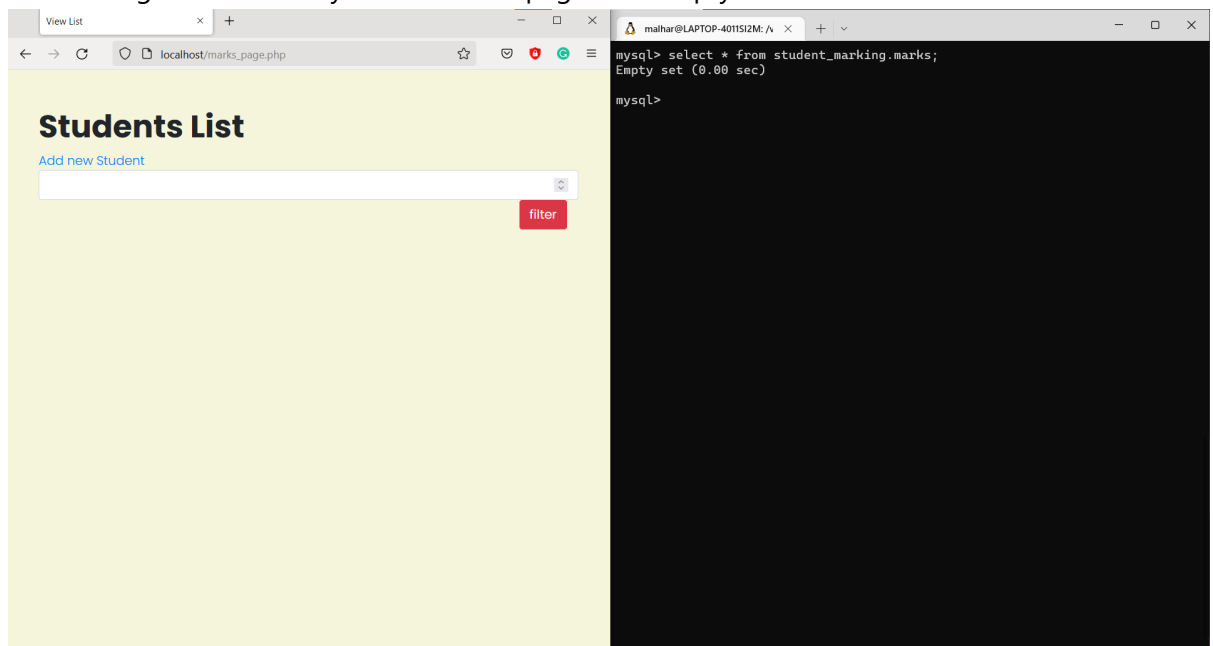
Teacher name

Password

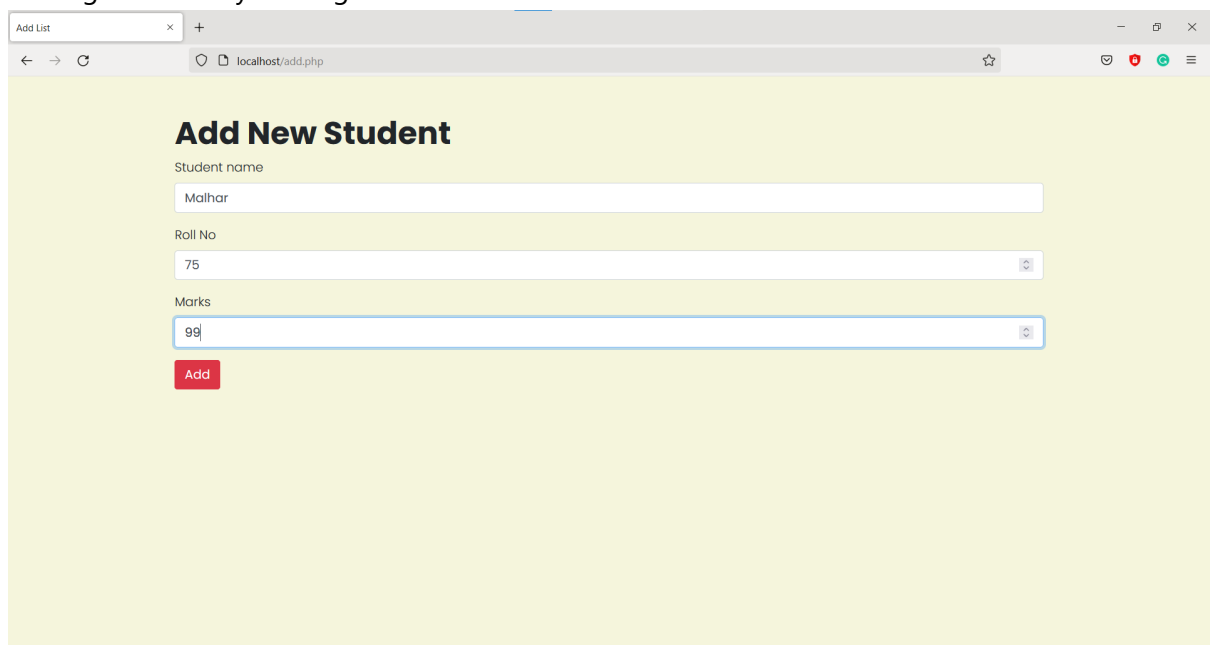
Login

Incorrect password or id

5. Correct Login details take you to the next page with empty student table :



6. Adding students by clicking on the add student link :



## Add New Student

Student name

Roll No

Marks

Add

Add List

## Add New Student

Student name

Roll No

Marks

Add

### 7. Results of adding students on the site and database :

View List

## Students List

[Add new Student](#)

filter

**Malhar**  
Roll-No: 75  
99/100  
[Delete](#) [Update](#)

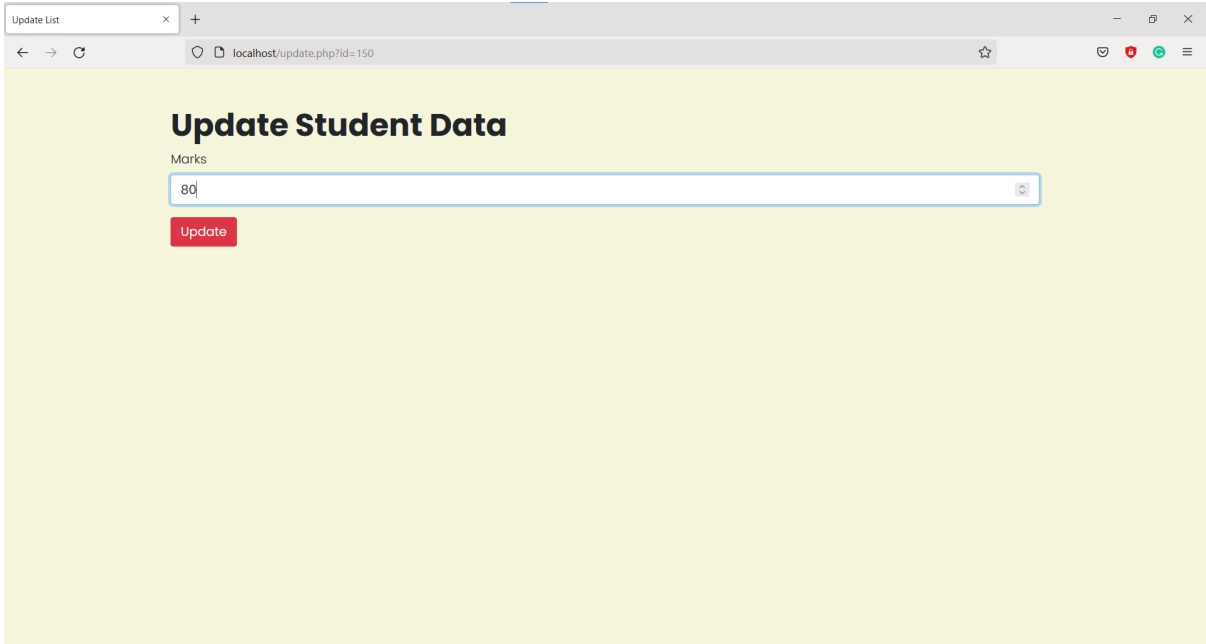
**Madhav**  
Roll-No: 76  
100/100  
[Delete](#) [Update](#)

**abc**  
Roll-No: 150  
75/100  
[Delete](#) [Update](#)

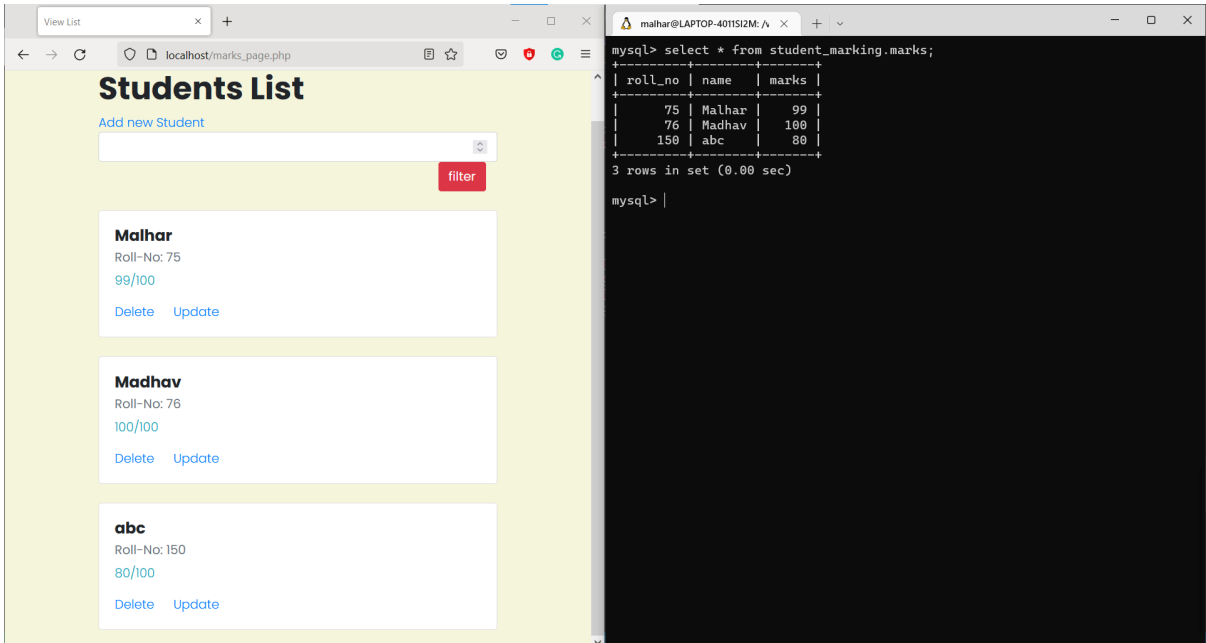
```
mysql> select * from student_marking.marks;
+-----+-----+-----+
| roll_no | name  | marks |
+-----+-----+-----+
| 75      | Malhar | 99    |
| 76      | Madhav | 100   |
| 150     | abc   | 75    |
+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> |
```

8. Updating marks of a student :



9. Results of updating on site and database :





10. Result of deleting a user with delete button :

The screenshot displays a web application interface on the left and a terminal window on the right. The web application, titled "Students List", features a form to "Add new Student" and a "filter" button. Below the form, two student entries are listed: "Malhar" (Roll-No: 75, 99/100) and "Madhav" (Roll-No: 76, 100/100). Each entry has "Delete" and "Update" buttons. The terminal window on the right shows the execution of the MySQL command `mysql> select * from student_marking.marks;`, which returns 3 rows of data. A handwritten note "before" with an arrow points to the first row of the terminal output.

roll_no	name	marks
75	Malhar	99
76	Madhav	100
150	abc	80

3 rows in set (0.00 sec)

mysql> select \* from student\_marking.marks;

roll_no	name	marks
75	Malhar	99
76	Madhav	100

2 rows in set (0.00 sec)

mysql> |

Submission by- BT20CSE075 Malhar Limaye.

Thank You!