

# Web Lab 1 (WSL + LAMP Setup and PHP-MySQL Integration)

**Student:** Towhid

**System:** Windows 10/11 using WSL2 (Ubuntu 24.04 LTS)

**Stack Used:** Apache2, PHP 8.3, MySQL 8.0

---

## 1. Introduction

The objective of this lab was to set up a complete LAMP (Linux, Apache, MySQL, PHP) environment using Windows Subsystem for Linux (WSL).

After setting up Ubuntu inside WSL2, Apache, PHP, and MySQL were installed and configured. Finally, multiple PHP pages were created to test database connection, insertion, listing, and deletion of data.

## 2. Installing WSL and Ubuntu

### Step 1 – Install WSL

Opened PowerShell as Administrator and ran:

```
wsl --install
```

After installation completed, the system was **restarted** as instructed.

### Step 2 – Fixing WSL Startup Errors

Initially, Ubuntu showed these errors:

**WslRegisterDistribution failed with error: 0x80370102**

**WslRegisterDistribution failed with error: 0x80080005**

**Fix:**

Entered BIOS (by pressing **F10** at startup) → Enabled **Virtualization (Intel VT-x / AMD-V)**.

Then rebooted Windows and WSL started successfully.

```

towhid@DESKTOP-EJC95S2: /var/www/html

root@DESKTOP-EJC95S2:~# adduser towhid
info: Adding user `towhid' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `towhid' (1000) ...
info: Adding new user `towhid' (1000) with group `towhid (1000)' ...
info: Creating home directory `/home/towhid' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for towhid
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n] y
info: Adding new user `towhid' to supplemental / extra groups `users' ...
info: Adding user `towhid' to group `users' ...
root@DESKTOP-EJC95S2:~# usermod -aG sudo towhid
root@DESKTOP-EJC95S2:~# su - towhid
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.6.87.2-microsoft-standard-WSL2 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Sat Nov  8 19:38:20 +06 2025

System load:  0.17               Processes:            31
Usage of /:   0.1% of 1006.85GB   Users logged in:     1
Memory usage: 6%                IPv4 address for eth0: 172.30.92.226
Swap usage:   0%

This message is shown once a day. To disable it please create the
/home/towhid/.hushlogin file.
towhid@DESKTOP-EJC95S2:~$ sudo apt update && sudo apt upgrade -y

```

### 3. Creating a User in Ubuntu

Once Ubuntu opened, created a new user and added it to sudo group:

```

adduser towhid
usermod -aG sudo towhid
su - towhid

```

**Confirmed working directory:**

pwd

## Output:

/home/towhid

Then updated packages:

```
sudo apt update && sudo apt upgrade -y
```

---

## 4. Installing Apache and Testing

### Step 1 – Install Apache

```
sudo apt install apache2 -y
```

### Step 2 – Start and Check Apache Status

```
sudo systemctl start apache2
```

```
sudo systemctl status apache2
```

If Active (running) shows in green, Apache is working.

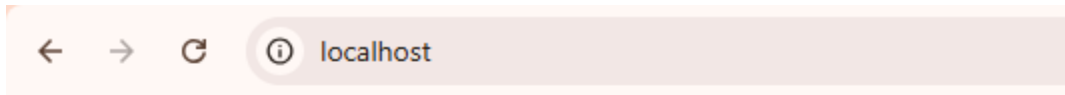
```
towhid@DESKTOP-EJC95S2:~$ sudo service apache2 start
towhid@DESKTOP-EJC95S2:~$ sudo service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
   Active: active (running) since Sat 2025-11-08 19:50:05 +06; 1min 30s ago
     Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 8426 (apache2)
   Tasks: 55 (limit: 7002)
   Memory: 5.4M (peak: 6.2M)
   CPU: 80ms
   CGroup: /system.slice/apache2.service
           └─8426 /usr/sbin/apache2 -k start
           └─8428 /usr/sbin/apache2 -k start
           └─8429 /usr/sbin/apache2 -k start

Get:1 http://archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.8 [1331 kB]
Get:7 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.8 [163 kB]
Get:8 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.8 [97.7 kB]
Get:9 http://archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.8 [90.2 kB]
Get:10 http://archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntu1 [17.8 kB]
Fetched 2086 kB in 6s (347 kB/s)
towhid@DESKTOP-EJC95S2:~$ cd /var/www/html
```

### Step 3 – Test Apache Default Page

Go to:

`http://localhost/`

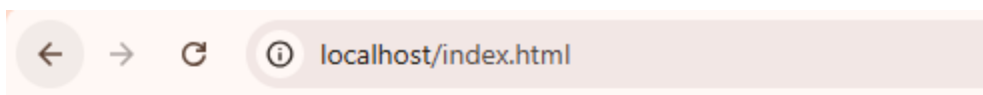


# Hello from Towhid!

This is my first web page using Apache server in WSL Ubuntu.

To create own test page:

```
cd /var/www/html  
echo "Hello from Towhid" | sudo tee index.html
```



# Hello from Towhid!

This is my first web page using Apache server in WSL Ubuntu.

# 5. Installing PHP and Testing PHP Functionality

## Step 1 – Install PHP

```
sudo apt install php libapache2-mod-php -y
sudo systemctl restart apache2
```

## Step 2 – Create a phpinfo() Test Page


```
echo "<?php phpinfo(); ?>" | sudo tee /var/www/html/info.php
```

Now open in browser:

http://localhost/info.php


localhost/info.php

PHP Version 8.3.6



System	Linux DESKTOP-EJC95S2 6.6.87.2-microsoft-standard-WSL2 #1 SMP PREEMPT_DYNAMIC Thu Jun 6 18:30:46 UTC 2025; x86_64
Build Date	Jul 14 2025 18:30:55
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.3/apache2
Loaded Configuration File	/etc/php/8.3/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.3/apache2/conf.d
Additional .ini files parsed	/etc/php/8.3/apache2/conf.d/10-opcache.ini, /etc/php/8.3/apache2/conf.d/10-pdo.ini, /etc/php/8.3/apache2/conf.d/20-calendar.ini, /etc/php/8.3/apache2/conf.d/20-ctype.ini, /etc/php/8.3/apache2/conf.d/20-exif.ini, /etc/php/8.3/apache2/conf.d/20-fileinfo.ini, /etc/php/8.3/apache2/conf.d/20-ftp.ini, /etc/php/8.3/apache2/conf.d/20-gettext.ini, /etc/php/8.3/apache2/conf.d/20-iconv.ini, /etc/php/8.3/apache2/conf.d/20-imagick.ini, /etc/php/8.3/apache2/conf.d/20-intl.ini, /etc/php/8.3/apache2/conf.d/20-ldap.ini, /etc/php/8.3/apache2/conf.d/20-mbstring.ini, /etc/php/8.3/apache2/conf.d/20-mcrypt.ini, /etc/php/8.3/apache2/conf.d/20-mysqlnd.ini, /etc/php/8.3/apache2/conf.d/20-openssl.ini, /etc/php/8.3/apache2/conf.d/20-pdo_mysql.ini, /etc/php/8.3/apache2/conf.d/20-phar.ini, /etc/php/8.3/apache2/conf.d/20-posix.ini, /etc/php/8.3/apache2/conf.d/20-readline.ini, /etc/php/8.3/apache2/conf.d/20-shmop.ini, /etc/php/8.3/apache2/conf.d/20-sockets.ini, /etc/php/8.3/apache2/conf.d/20-sysvmsg.ini, /etc/php/8.3/apache2/conf.d/20-sysvsem.ini, /etc/php/8.3/apache2/conf.d/20-sysvshm.ini, /etc/php/8.3/apache2/conf.d/20-tokenizer.ini
PHP API	20230831
PHP Extension	20230831
Zend Extension	420230831
Zend Extension Build	API420230831.NTS
PHP Extension Build	API20230831.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
Zend Max Execution Timers	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:  
Zend Engine v4.3.6. Copyright (c) Zend Technologies with Zend OPcache v8.3.6. Copyright (c), by Zend Technologies



Configuration

apache2handler

need to enable error display for debugging, open the PHP configuration file:

```
sudo nano /etc/php/8.3/apache2/php.ini
```

Uncomment and set:

```
display_errors = On
```

Then save and exit (Ctrl + O → Enter → Ctrl + X).

---

## 6. Installing MySQL and Setting Up Database

### Step 1 – Install MySQL Server

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

### Step 2 – Start and Check MySQL

```
sudo systemctl start mysql  
sudo systemctl status mysql
```

### Step 3 – Log into MySQL

```
sudo mysql
```

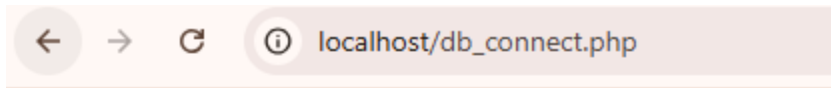
### Step 4 – Create Database and Table

```
CREATE DATABASE web_lab;  
USE web_lab;
```

```
CREATE TABLE students (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(50),  
    email VARCHAR(50)  
);
```

```
INSERT INTO students (name, email)
```

```
VALUES ('Towhid', 'towhid@example.com'),
      ('Rafi', 'rafi@example.com');
```



**Connected successfully to MySQL database!**

#### **Student Records**

ID	Name	Email
1	Towhid	towhid@example.com
2	Rafi	rafi@example.com

## **7. Enabling PHP-MySQL Integration**

If see an error like Class 'mysqli' not found, install the PHP MySQL module:

```
sudo apt install php8.3-mysql -y
sudo systemctl restart apache2
```

### **Step 2 – Create a New MySQL User**

Since the root account can cause permission issues, create a new user:

```
CREATE USER 'towhid'@'localhost' IDENTIFIED BY '12345';
GRANT ALL PRIVILEGES ON web_lab.* TO 'towhid'@'localhost';
FLUSH PRIVILEGES;
EXIT;
```

---

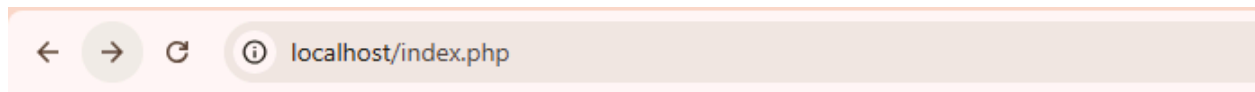
## **8. Creating PHP Pages**

All pages are stored in /var/www/html.

### a) index.php

Main homepage linking to all other files.

```
<?php
echo "<h2>Web Lab – Towhid</h2>";
echo "<p><a href='db_connect.php'>Test DB Connection</a></p>";
echo "<p><a href='add_student.php'>Add Student</a></p>";
echo "<p><a href='list_students.php'>List Students</a></p>";
?>
```



# Web Lab – Towhid

[Test DB Connection](#)

[Add Student](#)

[List Students](#)

### b) db\_connect.php

Displays database connection and student list.

```
<?php
$servername = "localhost";
$username   = "towhid";
$password   = "12345";
$database   = "web_lab";
```



```

$conn = new mysqli($servername, $username, $password, $database);
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "<h2>Connected successfully to MySQL database!</h2>";
$sql = "SELECT * FROM students";
$result = $conn->query($sql);

if ($result && $result->num_rows > 0) {
    echo "<h3>Student Records</h3>";
    echo "<table border='1'
cellpadding='6'><tr><th>ID</th><th>Name</th><th>Email</th></tr>";
    while ($row = $result->fetch_assoc()) {
        echo
        "<tr><td>".$row['id']. "</td><td>".$row['name']. "</td><td>".$row['email
        ']. "</td></tr>";
    }
    echo "</table>";
} else {
    echo "No data found!";
}
$conn->close();
?>

```

← → ↻ ⓘ localhost/db\_connect.php

# Connected successfully to MySQL database!

## Student Records

ID	Name	Email
1	Towhid	towhid@example.com
2	Rafi	rafi@example.com

### c) add\_student.php

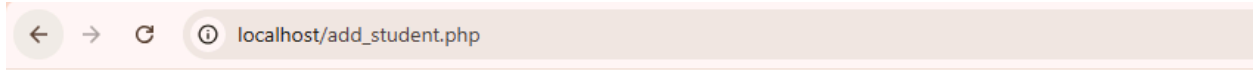
Form to add new students.

```
<?php
$servername = "localhost";
$username   = "towhid";
$password   = "12345";
$dbname     = "web_lab";

$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) { die("Connection failed: " .
$conn->connect_error); }

if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    $name = trim($_POST['name'] ?? '');
    $email = trim($_POST['email'] ?? '');
    if ($name !== '' && $email !== '') {
        $stmt = $conn->prepare("INSERT INTO students (name, email)
VALUES (?, ?)");
        $stmt->bind_param("ss", $name, $email);
        $stmt->execute();
        echo "Student added successfully! <a
href='list_students.php'>View list</a>";
        $stmt->close();
        $conn->close();
        exit;
    } else {
        echo "Name and Email required!";
    }
}
?>

<h2>Add Student</h2>
<form method="post">
    <label>Name: <input type="text" name="name"></label><br><br>
    <label>Email: <input type="email" name="email"></label><br><br>
    <button type="submit">Save</button>
</form>
```



## Add Student

Name:

Email:

Save

[Back to list](#)

### d) list\_students.php

Displays and allows deletion of students.

```
<?php
$servername = "localhost";
$username   = "towhid";
$password   = "12345";
$dbname     = "web_lab";
$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) { die("Connection failed: " .
$conn->connect_error); }

if (isset($_GET['delete'])) {
    $id = (int) $_GET['delete'];
    $conn->query("DELETE FROM students WHERE id=$id");
    header("Location: list_students.php");
    exit;
}

$result = $conn->query("SELECT * FROM students ORDER BY id");
```

```

?>
<h2>Student List</h2>
<table border="1" cellpadding="6">
<tr><th>ID</th><th>Name</th><th>Email</th><th>Action</th></tr>
<?php
if ($result && $result->num_rows > 0) {
    while ($row = $result->fetch_assoc()) {
        echo "<tr>";
        echo "<td>".$row['id']. "</td>";
        echo "<td>".$row['name']. "</td>";
        echo "<td>".$row['email']. "</td>";
        echo "<td><a href='list_students.php?delete=".$row['id']."'
onclick='return confirm(\"Delete?\");'>Delete</a></td>";
        echo "</tr>";
    }
} else {
    echo "<tr><td colspan='4'>No data</td></tr>";
}
?>
</table>
<p><a href="add_student.php">Add new</a></p>
<?php $conn->close(); ?>

```

## 9. Restarting Services

Whenever configuration changes are made:

```

sudo systemctl restart apache2
sudo systemctl restart mysql

```

## Student Records

[Add New](#) | [Home](#)

ID	Name	Email	Action
1	Towhid	towhid@example.com	<a href="#">Delete</a>
2	Rafi	rafi@example.com	<a href="#">Delete</a>
3	zahid	zahid@gmail.com	<a href="#">Delete</a>
4	tamjid	tamjid@gmail.com	<a href="#">Delete</a>
5	yasin	yasin@gmial.com	<a href="#">Delete</a>
6	mridul	mridul@gmail.com	<a href="#">Delete</a>