LAB 2: Javascript, DB, Jquery

Javascript, DB, JQuery
Name: Pralhad Parajuli
Roll No: ACE078BCT046

Objective:

To create a web application that interacts with a database using PHP and MySQL, implements user authentication with form validation, performs CRUD (Create, Read, Update, Delete) operations, and utilizes JavaScript/JQuery to display system time.

Theory:

PHP (Hypertext Preprocessor) PHP is a widely-used open-source general-purpose scripting language that is especially suited for web development and can be embedded into HTML. It runs on the server side and is used to manage dynamic content, databases, and session tracking.

MySQL Database MySQL is a relational database management system (RDBMS). It stores data in tables consisting of rows and columns. In web development, PHP is often used to connect to a MySQL database to store user information, product data, and more. **SQL Commands** * **INSERT:** Adds new data into a database table.

- **SELECT:** Retrieves data from a database.
- **UPDATE:** Modifies existing data in a database.
- **DELETE:** Removes data from a database.

JavaScript & JQuery JavaScript is a client-side scripting language used to create dynamic interactions on webpages. JQuery is a fast, small, and feature-rich JavaScript library that simplifies HTML document traversal and manipulation.

Form Validation Validation ensures that the input provided by the user is clean and correct. Client-side validation using JQuery provides immediate feedback by catching errors (like empty fields) before the data is sent to the server.

Code

index.php

```
<?php
mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);
```

```php
try {
 $conn = new mysqli("127.0.0.1", "root", "root", "lab_db", 5555);
} catch (mysqli_sql_exception $e) {
 die("Database Connection Failed: " . $e->getMessage());
}

if (isset($_GET['delete'])) {
 $id = $_GET['delete'];
 $stmt = $conn->prepare("DELETE FROM User WHERE id = ?");
 $stmt->bind_param("i", $id);
 $stmt->execute();
 header("Location: index.php");
}

if (isset($_POST['login'])) {
 $user = $_POST['username'];
 $pass = $_POST['password'];
 $id = $_POST['user_id'];

 if (!empty($user) && !empty($pass)) {
 if (!empty($id)) {
 $stmt = $conn->prepare("UPDATE User SET Username=?, Passwd=? WHERE id=?");
$stmt->bind_param("ssi", $user, $pass, $id);
 } else {
 $stmt = $conn->prepare("INSERT INTO User (Username, Passwd) VALUES (?, ?)");
$stmt->bind_param("ss", $user, $pass);
 }
 $stmt->execute();
 header("Location: index.php");
 }
}

$edit_user = ""; $edit_pass = ""; $edit_id = "";
if (isset($_GET['edit'])) {
 $id = $_GET['edit'];
 $res = $conn->query("SELECT * FROM User WHERE id=$id");
 $row = $res->fetch_assoc();
 $edit_user = $row['Username'];
```

```
$edit_pass = $row['Passwd'];
$edit_id = $row['id'];
}
?>

<!DOCTYPE html>
<html>
<head>
 <title>Lab 2 - PHP & JS</title>
</head>
<body>

 <div style="border: 1px solid black; padding: 20px; width: 300px; margin-bottom: 20px;">
<button id="showtimeBtn">Showtime</button>
 <p id="timeDisplay"></p>
 </div>

 <h3><?php echo $edit_id ? "Edit User" : "Create a database login page"; ?></h3>
<form id="loginForm" method="POST" onsubmit="return validateForm()">
 <input type="hidden" name="user_id" value="<?php echo $edit_id; ?>">

 <label>Username</label>
 <input type="text" name="username" id="username" value="<?php echo $edit_user; ?>"><br><br>
 <label>Passwd</label>
 <input type="password" name="password" id="password" value="<?php echo $edit_pass; ?>"><br><br>
 <input type="submit" name="login" value="<?php echo $edit_id ? "Update" : "Login"; ?>">
<?php if($edit_id): ?> <a href="index.php">Cancel</a> <?php endif; ?>
 </form>
 <p id="msg" style="color: red;"></p>

 <hr>

 <table border="1" cellpadding="10">
 <tr>
 <th>Username</th>
 <th>Passwd</th>
<th colspan="2">Action</th>
 </tr>
 <?php
```

```
$result = $conn->query("SELECT * FROM User");
while ($row = $result->fetch_assoc()) {
echo "<tr>
<td>" . $row['Username'] . "</td>
<td>" . $row['Passwd'] . "</td>
<td><a href='index.php?edit=" . $row['id'] . "'>edit</a></td>
<td><a href='index.php?delete=" . $row['id'] . "' onclick='return confirm(\"Are you sure?\")'>delete</a></td>
</tr>";
}
?>
</table>

<script>
document.getElementById('showtimeBtn').onclick = function() {
document.getElementById('timeDisplay').innerHTML = Date();
};

function validateForm() {
var user = document.getElementById('username').value;
var pass = document.getElementById('password').value;
if (user == "" || pass == "") {
document.getElementById('msg').innerHTML = "Please enter Username & Password";
return false;
}
return true;
}
</script>
</body>
</html>
```

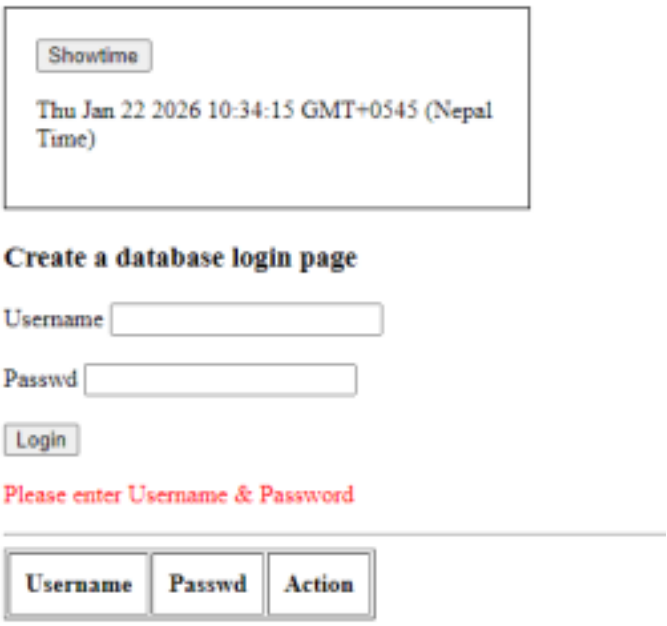
database.sql

```
CREATE TABLE IF NOT EXISTS User (
id INT AUTO_INCREMENT PRIMARY KEY,
Username VARCHAR(50) NOT NULL,
Passwd VARCHAR(50) NOT NULL
);
```

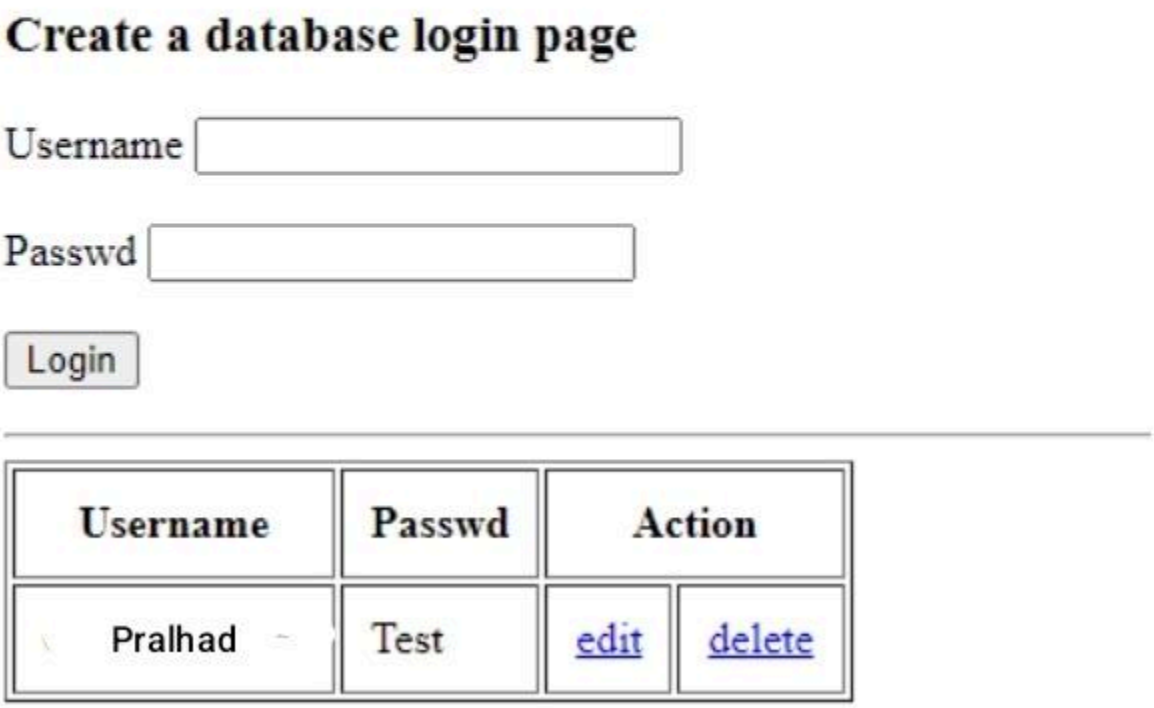
Output

Default Screen

Show time and null input



After addition of data



Discussion and Conclusion

In this lab, I successfully integrated client-side technologies with server-side technologies to build a functional web application.

By using JQuery, I was able to handle UI interactions such as the "Showtime" feature and perform form validation more efficiently than with plain JavaScript. I ensured the application prevents submission when inputs are empty, providing clear feedback to the user as per the task requirements.

On the backend, I established a connection to a MySQL database using PHP. I implemented logic to insert records and display them in a tabular format. This exercise helped me understand the full workflow of a data-driven web application, from the initial user input and validation to persistent storage and data retrieval.