

Week 6 - PHP & MySQL – Simple CRUD

Application Task Overview

You will build a **Student List** application. The application will store student records in a MySQL database and allow users to add, view, edit, and delete students.

Part 1: Database Setup (MySQL)

1. Create a database named **school_db**.
2. Create a table named **students** with the following fields:
 - 1 id – INT, Primary Key, Auto Increment
 - 2 name – VARCHAR(100)
 - 3 email – VARCHAR(100)
 - 4 course – VARCHAR(100)

3. Insert at least **5 sample student records** using SQL.

Part 2: Connecting PHP to MySQL

1. Create a file named **db.php** to store your database connection code.
2. Connect PHP to MySQL using **PDO or MySQLi**.
3. Display a simple message if the connection is successful or fails.

Part 3: CRUD Operations

Create: Create an HTML form to add a new student (name, email, course). **Read:** Display all students in an HTML table.

Update: Add an Edit button to update student details.

Delete: Add a Delete button to remove a student record.

Part 4: Security

1. Use prepared statements for INSERT, UPDATE, and DELETE queries.
2. Do not write SQL queries directly using user input.

The screenshot shows the phpMyAdmin interface for the 'school_db' database. The left sidebar lists various databases, and the main area displays the 'students' table. The table has columns: id, name, email, and course. Two rows are present: one for 'sapana' with 'BSc.HONS' and another for 'yunisa' with 'BIT'. The interface includes a toolbar at the top with 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', and 'Operations' buttons. Below the toolbar, there's a message bar indicating the query was successful. The table view includes filters and sorting options. At the bottom, there are additional operations like 'Print', 'Copy to clipboard', 'Export', 'Display chart', and 'Create view'.

	Edit	Copy	Delete	Edit	Copy	Delete	Export
1	sapana	sapana@gmail.com	BSc.HONS	2	yunisa	yunisa@gmail.com	BIT

```
1 <?php
2 require'db.php';
3
4 try{
5
6 if($_SERVER['REQUEST_METHOD']=="POST" && isset($_POST['add_student'])){
7     $name=$_POST['name'];
8     $email=$_POST['email'];
9     $course=$_POST['course'];
10
11    $sql="INSERT INTO students(name,email,course)VALUES(?, ?, ?)";
12    $stmt = $pdo-> prepare($sql);
13    $stmt-> execute([$name,$email,$course]);
14
15    echo"<h3 style='color:green;'>Student added</h3>";
16    header("Location:index.php");
17
18 }
19 }
20
21 catch(PDOException $e){
22     die("Failed to insert student".$e->getMessage());
23 }
24 ?>
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39 <!DOCTYPE html>
40 <html>
41 <head>
42     <meta charset="utf-8">
43     <meta name="viewport" content="width=device-width, initial-scale=1">
44     <title>Student Database</title>
45 </head>
46 <body>
47     <form method="POST">
48         Name: <input type="text" name="name" required><br>
49         Email: <input type="email" name="email" required><br>
50         Course: <input type="text" name="course" required><br>
51         <button type="submit" name="add_student">Add Student</button>
52     </form>
53
54 </body>
55 </html>
```

```
1 <?php
2 require'db.php';
3 try{
4 $sql="SELECT * FROM students";
5 $stmt=$pdo->query($sql);
6 $students = $stmt->fetchAll();
7 }
8 catch(PDOException $e){
9 die("Unable to f=et data from database".$e->getMessage());
10 }
11 ?>
12
13
14
15
16
17 <!DOCTYPE html>
18 <html>
19 <head>
20     <meta charset="utf-8">
21     <meta name="viewport" content="width=device-width, initial-scale=1">
22     <title>Studentb Database</title>
23 </head>
24 <body>
25 <h1 style="color: red;">Welcome to Student Database!</h1>
26
27 <table>
28     <tr>
29         <th>Name</th>
30         <th>Email</th>
31         <th>Course</th>
32         <th>Edit</th>
33     </tr>
34     <?php foreach ($students as $student):?>
35     <tr>
36         <td><?= $student['name']?></td>
37         <td><?= $student['email']?></td>
38         <td><?= $student['course']?></td>
39         <td><a href="edit.php?id=<?= $student['id']?>">Edit</a>
40         <a href="delete.php?id=<?= $student['id']?>">Delete</a></td>
41     </tr>
42     <?php endforeach ?>
43 </table>
44     <br>
45     <a href="create.php">Add New Student</a>
46 </body>
47 </html>
```

```
1  <?php
2
3  require 'db.php';
4  $id=$_GET['id']??: '';
5
6  if (!$_id) {
7      die("Invalid ID");
8  }
9  $sql="SELECT * FROM students WHERE id=?";
10 $stmt= $pdo-> prepare($sql);
11 $stmt->execute([$id]);
12 $student = $stmt->fetch(PDO::FETCH_ASSOC);
13
14 if (!$student) {
15     die("Student not found");
16 }
17
18 if($_SERVER['REQUEST_METHOD']=="POST" && isset($_POST['add_student'])){
19     $name=$_POST['name'];
20     $email=$_POST['email'];
21     $course=$_POST['course'];
22
23     $updateSql = "UPDATE students SET name = ?, email = ?, course = ? WHERE id = ?";
24     $updateStmt = $pdo->prepare($updateSql);
25     $updateStmt->execute([$name, $email, $course, $id]);
26
27     header("Location:index.php");
28 }
29
30 ?>
31
32
33
34
35 <!DOCTYPE html>
36 <html>
37 <head>
38     <meta charset="utf-8">
39     <meta name="viewport" content="width=device-width, initial-scale=1">
40     <title>Student Database</title>
41 </head>
42 <body>
43     <form method="POST">
44         Name: <input type="text" name="name" value=<?=htmlspecialchars($student['name'])??> required><br>
45         Email: <input type="email" name="email" value=<?=htmlspecialchars($student['email'])??> required><br>
46         Course: <input type="text" name="course" value=<?=htmlspecialchars($student['course'])??> required><br>
47         <button type="submit" name="add_student">Update</button>
48     </form>
49 </body>
50 </html>
```

The image shows a code editor interface with three tabs at the top: "edit.php", "index.php", and "delete.php". The "edit.php" tab is currently selected. The code in the editor is as follows:

```
1 <?php
2
3 require 'db.php';
4
5 $id=$_GET['id']??'';
6 if(isset($id)){
7     $sql="DELETE FROM students WHERE id=?";
8     $stmt=$pdo->prepare($sql);
9     $stmt->execute([$id]);
10    echo "Student deleted successfully!";
11 } else {
12     echo "Invalid ID";
13 }
14
15 header("Location: index.php");
16 exit;
17
18
19 ?>
```

Database connected successfully

Name: tokki

Email: tokki123@gmail.com

Course: BIT

[Add Student](#)

Database connected successfully

Welcome to Student Database!

Name	Email	Course	Edit
sapana	sapana@gmail.com	BSc.HONS	Edit Delete
yunisa	yunisa@gmail.com	BIT	Edit Delete
tokki	tokki123@gmail.com	BIT	Edit Delete

[Add New Student](#)

Database connected successfully

Name:

Email:

Course:

Database connected successfully

Welcome to Student Database!

Name	Email	Course	Edit
sapana	sapana@gmail.com	BSc.HONS	Edit Delete
yunisa	yunisa@gmail.com	BIT	Edit Delete
tokki	tokki123@gmail.com	BA	Edit Delete

[Add New Student](#)

← → ↴ ↵

		id	name	email	course
<input type="checkbox"/>	 Edit	 Copy	 Delete	2 yunisa	yunisa@gmail.com BIT
<input type="checkbox"/>	 Edit	 Copy	 Delete	3 tokki	tokki123@gmail.com BA

↑ Check all With selected: [Edit](#) [Copy](#) [Delete](#) [Export](#)

Show all | Number of rows: [25](#) Filter rows: Sort by key: [None](#)

Database connected successfully

Welcome to Student Database!

Name	Email	Course	Edit
yunisa	yunisa@gmail.com	BIT	Edit Delete
tokki	tokki123@gmail.com	BA	Edit Delete

[Add New Student](#)