

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 web interface. On the left is a sidebar with a tree view of databases and tables. The main area displays the 'students' table from the 'school_db' database. The table structure is shown with columns: id, name, email, and course. Two rows of data are listed. Below the table, there are controls for filtering, sorting, and performing operations on the query results.

	id	name	email	course
<input type="checkbox"/>	1	sapana	sapana@gmail.com	BSc.HONS
<input type="checkbox"/>	2	yunisa	yunisa@gmail.com	BIT

```

1  <?php
2
3  require 'db.php';
4
5  try{
6
7  if($_SERVER['REQUEST_METHOD']==="POST" && isset($_POST['add_student'])){
8      $name=$_POST['name'];
9      $email=$_POST['email'];
10     $course=$_POST['course'];
11
12     $sql="INSERT INTO students(name,email,course)VALUES(?,?,?)";
13     $stmt = $pdo-> prepare($sql);
14     $stmt-> execute([$name,$email,$course]);
15
16     echo"<h3 style='color:green;'>Student added</h3>";
17
18     header("Location:index.php");
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>

```

```
edit.php x index.php x delete.php x create.php x
C:\xampp\htdocs\FullStack\week_6\index.php
1
2 <?php
3 require 'db.php';
4 try{
5     $sql="SELECT * FROM students";
6     $stmt=$pdo->query($sql);
7     $students = $stmt->fetchAll();
8 }
9 catch(PDOException $e){
10     die("Unable to f=get data from database".$e->getMessage());
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>Student 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>
48
49
```

```

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

```

```
edit.php x | index.php x delete.php
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: tokki

Email: tokki123@gmail.com

Course: BA




Update

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](#)

<div><div>↩</div><div>⌵</div><div>→</div></div>						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:

Search this table

 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](#)