



KP34903: WEB TECHNOLOGY

Faculty of Computing and Informatics

Semester 1 2025/2026

LAB TEST 2: PHP & MYSQL with CRUD

Instructions:

1. Develop a **Student Management Web Application** with **PHP** and **MySQL**.

Your system must allow the user to:

- **Insert** a new student
- **View** a list of students
- **Update** student details
- **Delete** a student

2. You must use:

- PHP
- MySQL
- XAMPP / WAMP
- MySQLi (procedural or OOP)

3. No frameworks allowed

Database Requirements:

1. Create a database named:
 - *labtest2*
2. Create a table named:
 - *students*
3. With these fields:

Field	Type
id	INT(AUTO_INCREMENT, PRIMARY KEY)
name	VARCHAR(100)
email	VARCHAR(150)
course	VARCHAR(100)

4. Students **MUST** successfully create this before continuing.

Required Files:

File	Purpose
db.php	Database connection
index.php	Display student list (READ)
add.php	Add student (CREATE)
edit.php	Edit student (UPDATE)
delete.php	Delete student (DELETE)

Feature Requirement:

1. Display Records (index.php)

- Fetch all students from database
- Display in HTML table
- Include **Edit** and **Delete** buttons

2. Add Student (add.php)

Form fields:

- Name
- Email
- Course

On submission → Insert into database.

3. Edit Student (edit.php)

- Load existing student data into form
- Allow updating student
- Save changes to database

4. Delete Student (delete.php)

- Delete using student ID
- Redirect back to index.php

Bonus (Optional, 2 points)

- Basic CSS styling allowed
- Table formatting
- Buttons
- Form alignment

Not required but encouraged.

Submission Checklist

- **Project Files & Zip Folder Structure:**

- labtest2/
 - |— index.php
 - |— add.php
 - |— edit.php
 - |— delete.php
 - |— db.php
 - |— style.css
 - |— sqlscript.sql
 - |— (any images or additional assets, if used)

- **Self-Check Before Upload Zip File**

- Before submitting, confirm:
 - ☐ All CRUD functions work
 - ☐ No PHP errors
 - ☐ No database connection errors
 - ☐ SQL file included
 - ☐ **ZIP file opens properly**

Evaluation Rubric (30 Marks)

Criteria	Weight	Excellent (5)	Good (4)	Satisfactory (3)	Poor (2)	Unsatisfactory (1)
1. Database Setup	15%	13–15 marks: Correct DB + table, proper fields, PK set, no errors	10–12 marks: Mostly correct, minor structure issues	6–9 marks: Table exists but structure partly wrong	3–5 marks: Many mistakes; schema partially incorrect	0–2 marks: No DB or wrong setup preventing CRUD
2. READ Operation (index.php)	20%	17–20 marks: Data loads perfectly, clean table, links working	14–16 marks: Data loads with minor layout issues	10–13 marks: Partially displays; missing fields	5–9 marks: Attempted but broken layout or errors	0–4 marks: Not working at all
3. CREATE Operation (add.php)	20%	17–20 marks: Inserts correctly, validation done, smooth redirect	14–16 marks: Inserts successfully, minor issues	10–13 marks: Inserts but lacks validation/missing structure	5–9 marks: Form incomplete or errors on submit	0–4 marks: No create function
4. UPDATE Operation (edit.php)	20%	17–20 marks: Loads correct data, updates successfully, no errors	14–16 marks: Works, minor form/data issues	10–13 marks: Works but not fully correct (e.g., form not pre-filled)	5–9 marks: Attempted but flawed; errors or missing fields	0–4 marks: No update function
5. DELETE Operation (delete.php)	10%	9–10 marks: Deletes accurately, smooth redirect	7–8 marks: Works with minor issues	5–6 marks: Partially works (requires refresh, ID issues)	2–4 marks: Attempted but faulty	0–1 mark: Not implemented
6. Code Quality & Structure	10%	9–10 marks: Clean, readable, consistent, well-structured	7–8 marks: Good clarity, minor indentation issues	5–6 marks: Functional but messy formatting	2–4 marks: Confusing structure, redundant code	0–1 mark: Poor quality, many errors
7. UI/UX Styling (Optional)	5%	5 marks: Clean, styled forms/tables, consistent	4 marks: Some styling, acceptable layout	3 marks: Basic styling	1–2 marks: Minimal or inconsistent styling	0 marks: No styling

Total Weight: 100% (Converted to 10%)