Assignment 2 – PHP & SQL

This assignment involves the creation of some PHP pages and making use of the provided SQL database to create a student information system.

This assignment is due on Friday Nov 21st, 2025 by 11:15am (end of class).

General Requirements

An example of a possible finished product has been provided on the course webpage. Please look at and examine these pages to see the suggested implementation. You are free to improve on the styling or arrangement, but the example should serve as the minimum quality required.

You should be aware that this assignment, and all others, will be locked from your account precisely at the time of the assignment deadline.

Placement of Assignment Files

All the files you create for this assignment should be in the directory:

/var/www/html/students/<YOUR FOLDER>/XXXXXX/assn2/

Here, <YOUR FOLDER> is replaced with your student folder and XXXXXX is the secret subdirectory that was created for your account. **Do not share this with others.**

Specific Instructions

For this assignment you will build three pages.

- *students.php* : This page will be focused on displaying student data.
- *courses.php* : This page will be focused on displaying course data.
- *teachers.php* : This page will be focused on displaying teacher data.
- **DO NOT CREATE ANY OTHER PAGES**
- All pictures are stored in the "/sidpics/" folder in "student number.JPG" format. Ex: "/sidpics/338829286.JPG"

Global requirements (5 marks)

- All three pages should have a common header in the top right that allows the user to jump between the three different pages.
- Styling should be consistent across pages
- Titles are expected on every page, and be different for each view as shown in the example.
- Your pages should not display any errors
- Any time there is a student name, that name should be a link to that student's page
- Any time there is a teacher name, that name should be a link to that teacher's page
- Any time there is a course code (section), that course code should be a link to that section's page.

---

students.php – The students page (10 marks)

This page must provide three different views:

- A general view, asking the user to choose a letter that will be used for the "letter view"
- A letter view which displays all the students who have a last name starting with the provided letter, sorted by grade.
- A specific student view, that shows that student's timetable. Specifically:
  - Properly showing schedule in a table (semesters, periods, etc)
  - Each period should display the course, course name, room and teacher.
  - The period should be empty if the student does not have any registration that period.
  - The student picture should be displayed similar to the example.
  - Remember to handle courses that are in periods 3-4, or 1-2. (Example: Billy MacTerrelly)
  - Remember to handle courses that have term 1 & term 2. (Example: Harper Dellatorre)

courses.php – The courses page (10 marks)

This page must also provide three different views:

- A general view, asking the user to choose a letter that will be used for the "letter view"
- A letter view, which displays all the sections starting with the provided letter in some kind of tabular view.
- A specific section view, that lists all students registered in this section. Specifically:
  - This page should list the description of the course, and the teacher's name.
  - The list of students should be in alphabetical order by last name.
  - The pictures for all students should be displayed the same way as the example.

teachers.php – The teachers page (10 marks)

This page must provide two different views:

- A general view, asking the user to choose a teacher. This list of teachers should be some kind of tabular view.
- A specific teacher view, that shows that teacher timetable. Specifically:
  - Properly showing schedule in a table (semesters, periods, etc)
  - Each period should display the course(s), course name(s), and room.
  - The period should be empty if the teacher does not have any registration that period.
  - If there are multiple registrations in one period, they should all be displayed together.
  - There should be a link in the top left ("BACK") to return to the general view of the teachers page.
  - Remember to handle teachers who are teaching co-op classes (period 1-2, or 3-4).
    - Example: Elysia Hopkins

How should you approach the work for this assignment?

- Make your three pages (students.php, courses.php, teachers.php) in your assignment folder.
- Establish a common header to link to the three pages
- Experiment with making links, and passing variables through them and accessing via GET.
  - *Ex: <a href='teachers.php?id=42'>Emmell, Stephen</a>*
- Investigate the database using phpMyAdmin and learn how the data is recorded.
- Slowly build up the functionality of each page. The goal is to always have a working page. Fix errors immediately – do not assume you can continue developing other features while pages are throwing errors.

APPENDIX: Database description

```
+---------------------------+
| Tables_in_assignments_sid |
+---------------------------+
| registrations             |
| sections                  |
| students                  |
| teachers                  |
+---------------------------+
```

```
describe students;
+--------------+------------------+
| Field        | Type             |
+--------------+------------------+
| stud_id      | int              |
| stud_stud_id | int              |   //stud_stud_id is their student number.
| stud_lname   | varchar(string)  |
| stud_fname   | varchar(string)  |
| stud_grade   | int              |
+--------------+------------------+
```

```
describe teachers;
+-------------+-----------------+
| Field       | Type            |
+-------------+-----------------+
| teach_id    | int             |
| teach_lname | varchar(string) |
| teach_fname | varchar(string) |
+-------------+-----------------+
```

```
describe sections;
+--------------+------------------+
| Field        | Type             |
+--------------+------------------+
| sect_id      | int              |
| sect_code    | varchar(string)  |
| sect_period  | varchar(string)  |   //Ex: 1,2,3,4, or 1-2, or 3-4
| sect_room    | varchar(string)  |
| sect_term    | int              |   //sect_term refers to which half of the semester.
| sect_sem     | int              |   //   This only applies to half credit courses like
| sect_desc    | varchar(string)  |   //   Civics & Careers
| sect_teacher | int              |
+--------------+------------------+   //sect_teacher is the ID of the teacher
```

```
describe registrations;                //Stores a row for each student registered in
+-------------+---------+              // each section.
| Field       | Type    |
+-------------+---------+              //To find out who is registered in section 42:
| reg_id      | int     |              // search: “where reg_sect_id = 42”
| reg_sect_id | int     |
| reg_stud_id | int     |              //To find out what sections student 99 is in:
+-------------+---------+              // search: “where reg_stud_id = 99”
```