**Database Development and Class Registration**

Benjamin Brown

The University of Arizona Global Campus

CST499: Capstone for Computer Software Technology

Professor Joseph Rangitsch

March 10, 2025

**Introduction**

In this project, we will create a web application that connects to a MySQL database and allows users to manage their class schedules. Registration for classes, course schedules, adding new courses, and deleting courses are all available through the application. By building and linking tables in the MySQL database, the project aims to streamline course registration and schedule management, allowing students to organize their academic commitments in an easy-to-use manner.

**Database Design**

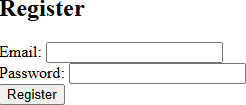
* There are three tables in this database: Students, Classes, and Enrollment. Students, Classes, and Enrollment tables contain information about students, courses, and their enrolled classes. In order for students to register for classes and schedule their classes, these tables are used together.

**Screenshot 1**:

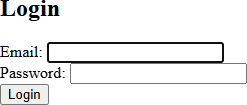


**Pages Created**

* **Screenshot 3**: Register page



* **Screenshot 4**: Login page

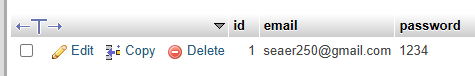


* **Screenshot 5**: Profile page

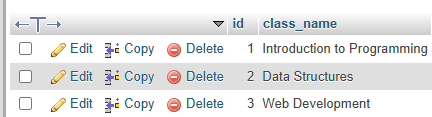


**Database Operation**

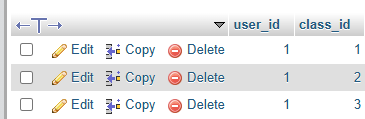
* **Screenshot 6**: Registered Users



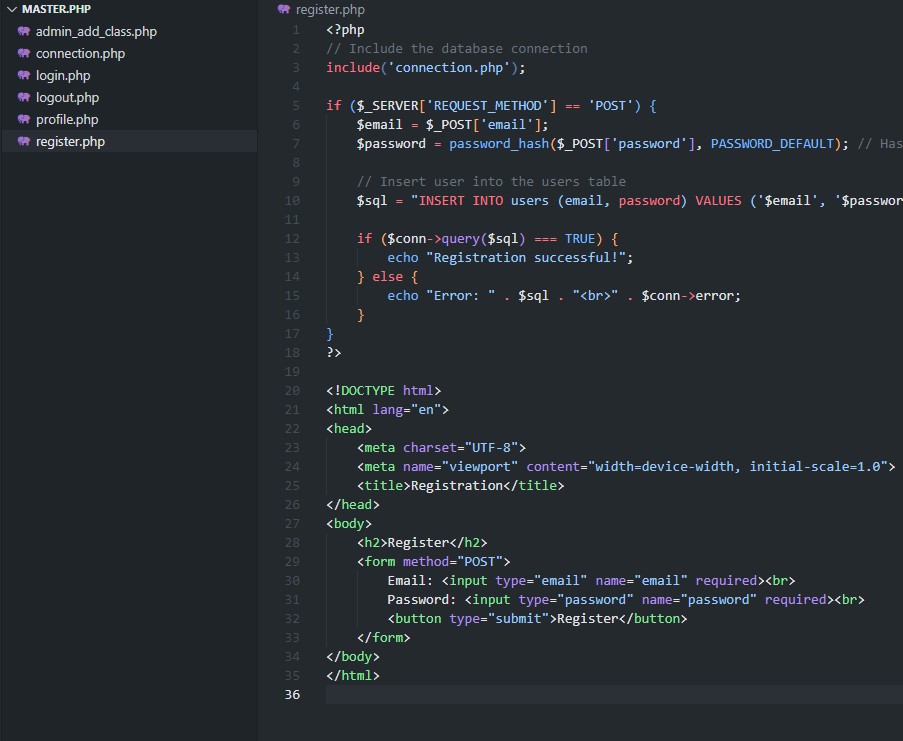
* **Screenshot 7:** Classes tables



* **Screenshot 8**: Enrollment



**Code**

* **Screenshot 10**:
* **Screenshot 11**:

**SQL Inquries**

PHP code allows users to enroll in or drop classes through HTML forms. A secure INSERT query with prepared statements inserts the selected class\_id into the enrollments table when a user enrolls. When a class is dropped, a DELETE query is run to remove the corresponding record from enrollments. A SELECT query retrieves the available classes and displays them in a dropdown, while a JOIN query retrieves the user's enrolled classes. As a result, enrollment and class removal actions are handled securely and efficiently in the database.

**Summary of Experience**

By reworking my past project, which was originally designed for a different purpose, I was able to gain valuable experience developing the student registration, login, and class enrollment system. By using prepared statements for database queries and managing user sessions effectively, we were able to adapt the system to handle student data securely and protect against SQL injection. In order to avoid duplication of records, existing records must be checked before new ones can be inserted. It was through this process that I learned how to manage secure sessions, validate input properly, and interact efficiently with the database. As part of future improvements, database triggers and audits could be added to track system usage, enforce data integrity automatically, and ensure greater transparency and accountability.

**References**

W3Schools. (n.d.). *SQL syntax*. W3Schools. Retrieved December 2, 2024, from <https://www.w3schools.com/sql/sql_syntax.asp>