**Week 4 Assignment – Database Development and Class Registration**

Maria Reyes

The University of Arizona Global Campus

CST 499 Capstone for Computer Software Technology

Professor Charmelia Butler

October 19, 2023

As the progress of the enrollment course system build continues, more and more pages are being constructed to form the puzzle pieces. Those puzzle pieces enlist the access to the system from the user’s side and the administrator’s side. From the last section, only the pages were designed and the connection was established with the database. That formed the basis of a registration page, a log in page, and the landing page that acts similarly to the home page. Within this paper, the script pages that continue are the ones that form the courses, the register, the different tables, and the ability to add and delete courses, which will be explained in the following paragraphs.

The first step was to add PHP code to the registration page and the log in page. Within the registration page, the form from the HTML script had to match the data elements from the database. If those data elements do not match, then it would not save into the database. Rather, it would generate an error. If it is correct, then when the user registers to the system, then the database saves their information into the “student” table. After, the log in page is constructed in its script. Since the two elements that are required in the log in process, the necessary fields to fill out were the “Student ID” and the “Password”. The script uses the POST method to insert the user’s information from each field that they entered into the databases. Then, the SELECT statement is used. “It is used to retrieve data from the database” and a query variable was used “for running a SELECT statement” (Connolly & Hoar, 2018, p. 633). Those two combinations demonstrate how the registration page works and interacts with the database. In Figure 7, the result of one user registered is shown.

Now, the second step would be to create a page for the administrator as well to insert classes into the database. Similar to the form created within the registration page, the administrator needs access to a form with fields to enter the different classes and their attributes. The administrator can enter as many times that they can to fill the different courses in the different semesters. Unfortunately, this is where I had a difficult time within the script. Although I kept looking for ways to change the database variables within the “online\_course\_table”, the errors consisted of some of the variables being unidentified. There were only three of them that caused the error, so I was very stumped because the rest of them did not cause a problem. Therefore, I spent various days trying to identify the problem. For a time, however, I decided to concentrate on other elements that needed to be completed.

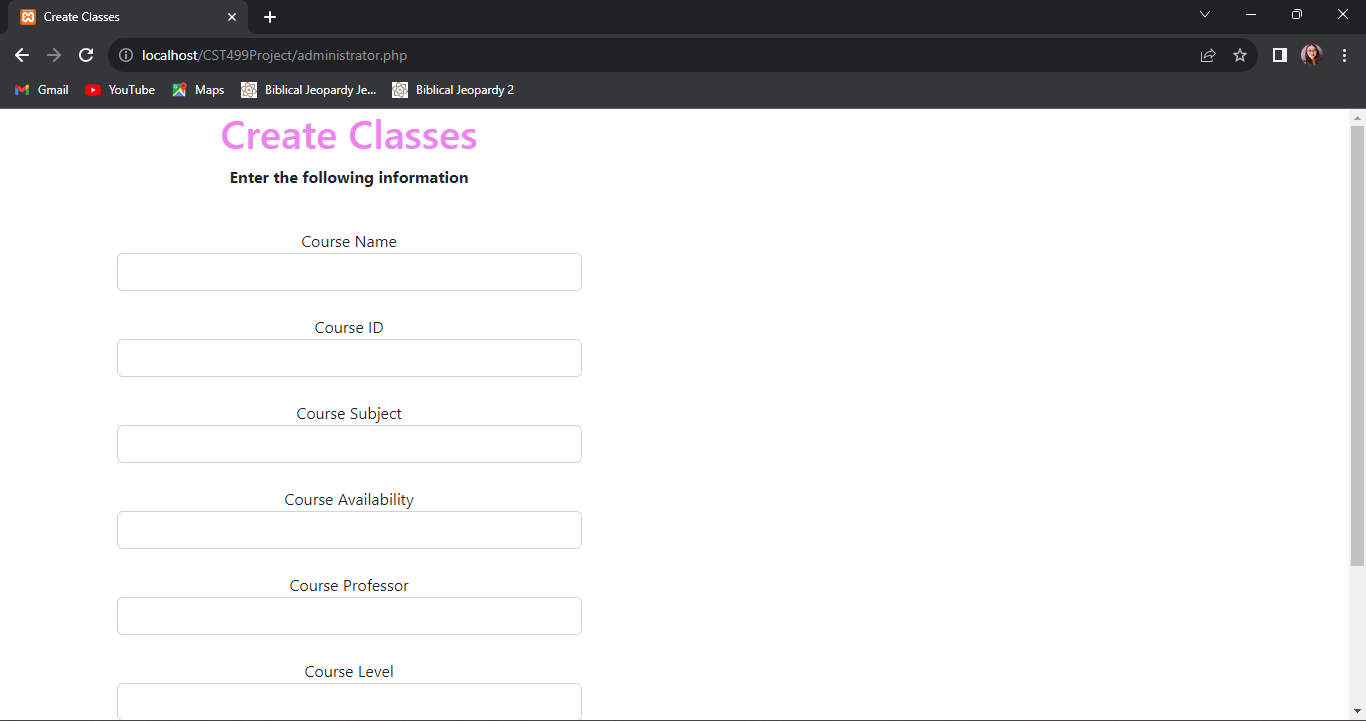
On the other hand, the third step was also very vital. This was the classes page. Within this page, the results of the “online\_course\_list” were to be displayed within a page. A form and table were styled in the HTML script. This provided the different course attributes to be displayed in an organized manner and the user can clearly identify them. Proceeding was the form for the user, where they can choose the courses, they would like to register to. However, the difference between this form from the rest is that it includes options that they can select. Since the options are not listed yet because of the errors that keep turning up, they are currently left blank. Finally, the form includes a button that they can register to.

Furthermore, that leads to another table that has to be created. The new table generated is called “enrolled\_students”. The primary key that will aid the administrator in their search for a certain user is the “student\_id”. The following fields consist of the courses that they chose during the course registration process. This will also allow the administrator to count how many members are signing up for the courses and if there is any more room left. In turn, it will connect to the “waiting\_list” table, where the members who could not sign up in time or the course was already full, then their information would be stored there.

Another page that was created, was the Success Page. This is where the administrator page and the classes page leads to. That demonstrates that the administrator was able to add course and the student was successfully able to add courses to their register. What will be displayed in the page is a simple success message. For both the administrator and the user, the message says “Fields completed.” While it is not complete for now, the script is just something where the administrator and user can observe some type of completion from the process. It is also a test page to test if my code works.

Something that I learned from this process is that it is difficult to solve some of the code and it can be overwhelming. As reading the textbook from Tsui et al., (2018), I learned that the reason may be that the “design, code, and test cycle of a problem-fix is not very different from the design, code, and test activities of the development cycle”, but “the danger is that the fix cycle may not utilize the same level of the software engineering discipline because may of the problem-fixes may involve only one or two lines-one-code fix” (p. 261). Although this relates to the deployment process, where users have the product at their disposal, it can certainly be a learning lesson during the implementation process. Therefore, solving the problem involves similar methods, but it should be done continuously, even during updates. Another insight was how PHP statements are organized. They “are considered as dynamic typing statements, and its variable does not need to be declared (just write and use without defining data type” (Odeh, 2019, p. 1520). Knowing this, I grew more knowledge about PHP syntax and the statements in order to connect with the database.

Overall, this experience was not easy because I was constantly was faced with problems concerning unidentified variables from the administrator page and the log in page. Fortunately, it was very exciting creating the HTML pages for the administrator, classes, and the new table. As a result, the scripts are becoming even more complete along with the tables.

Figure 1: Administrator Pages

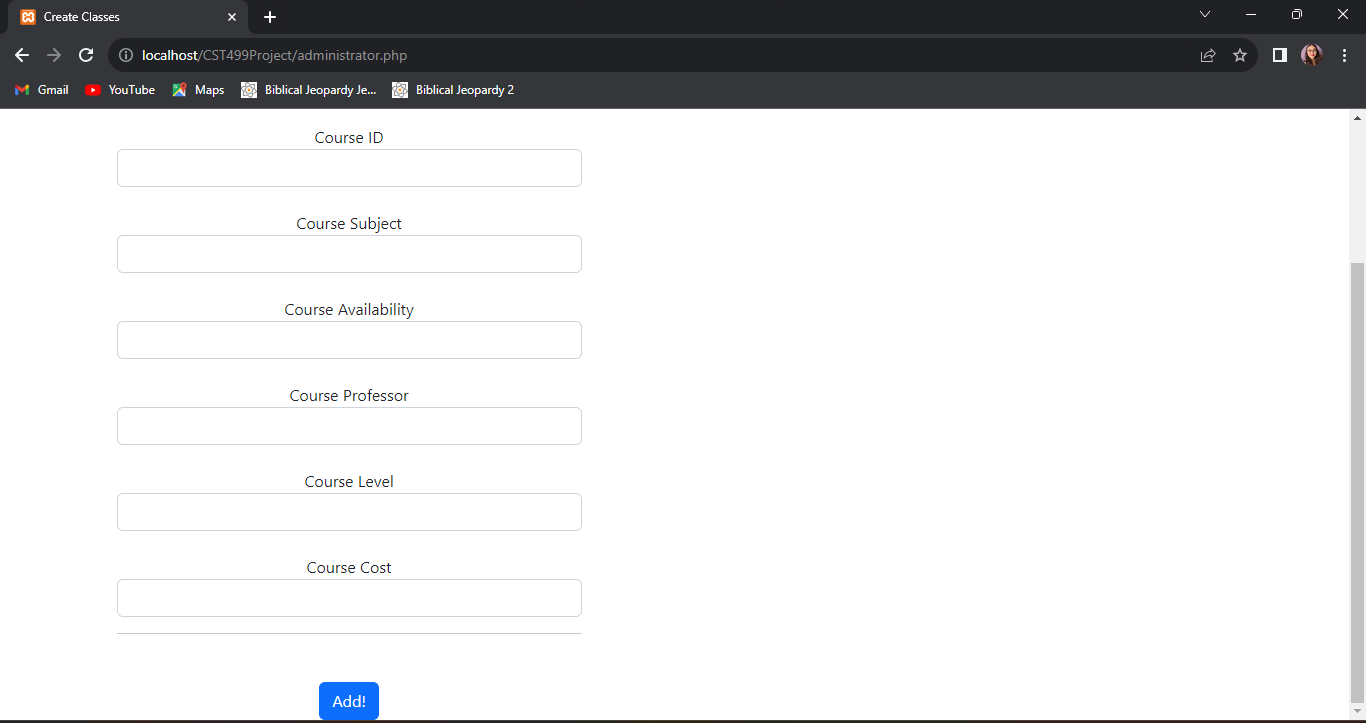


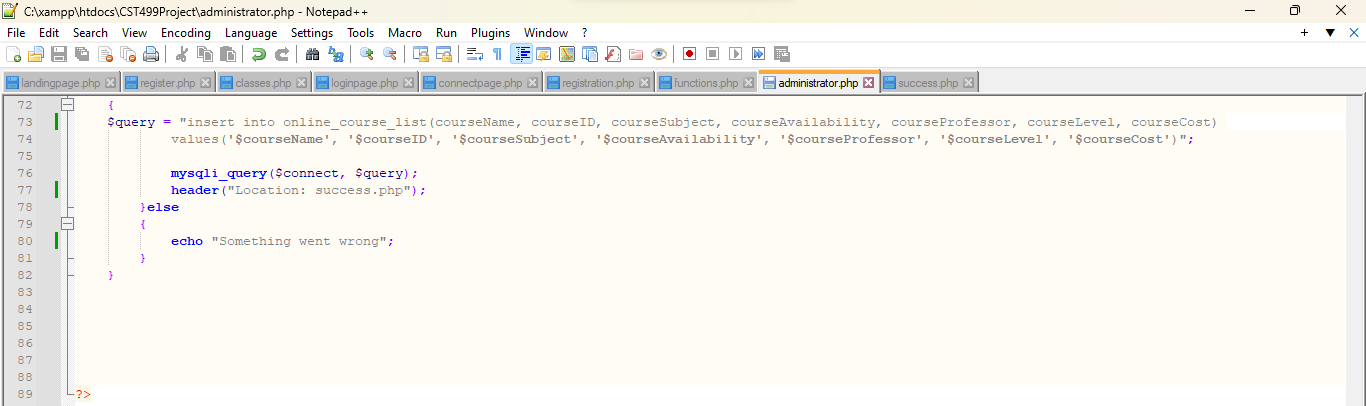
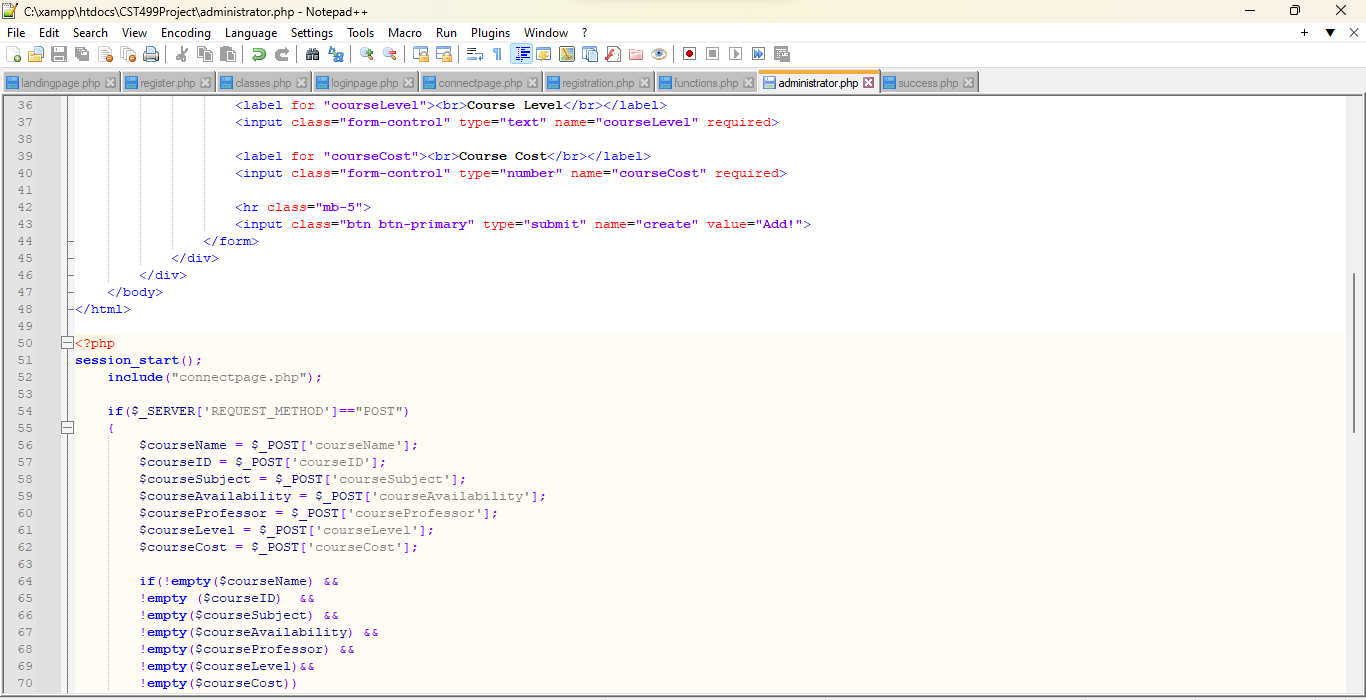
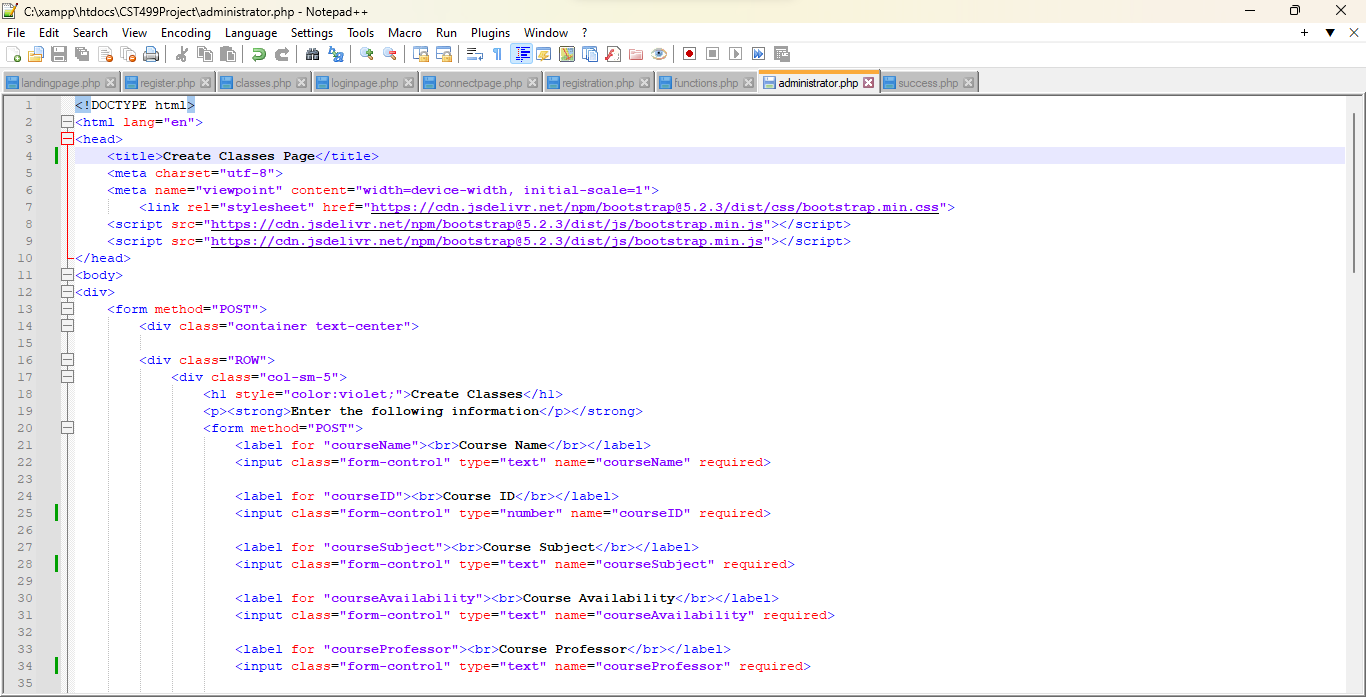
Figure 2: Administrator Page Scripts

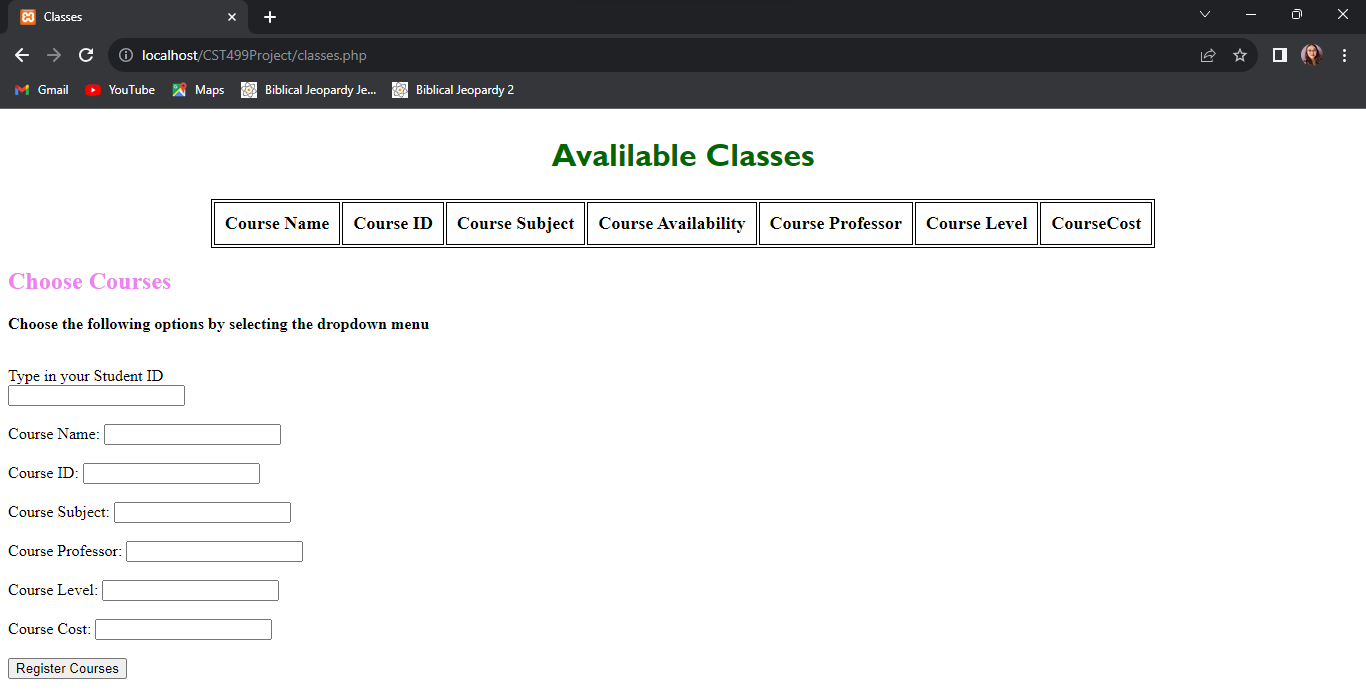
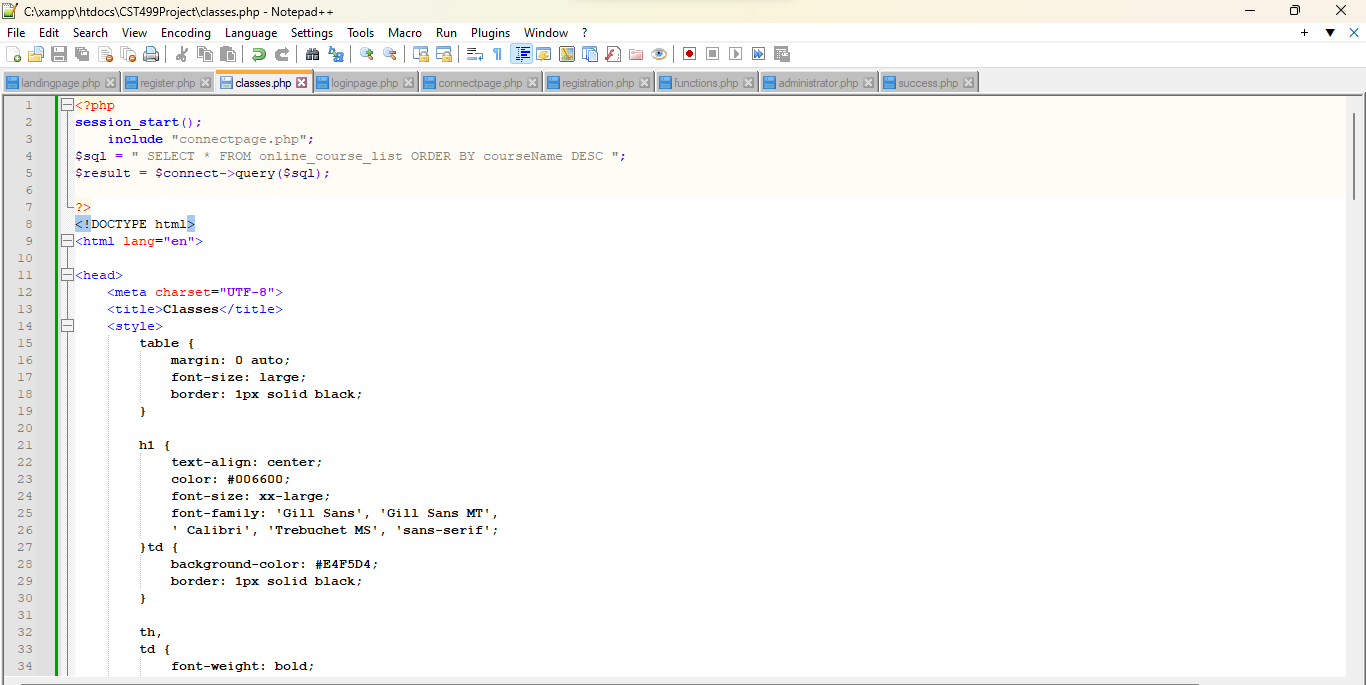
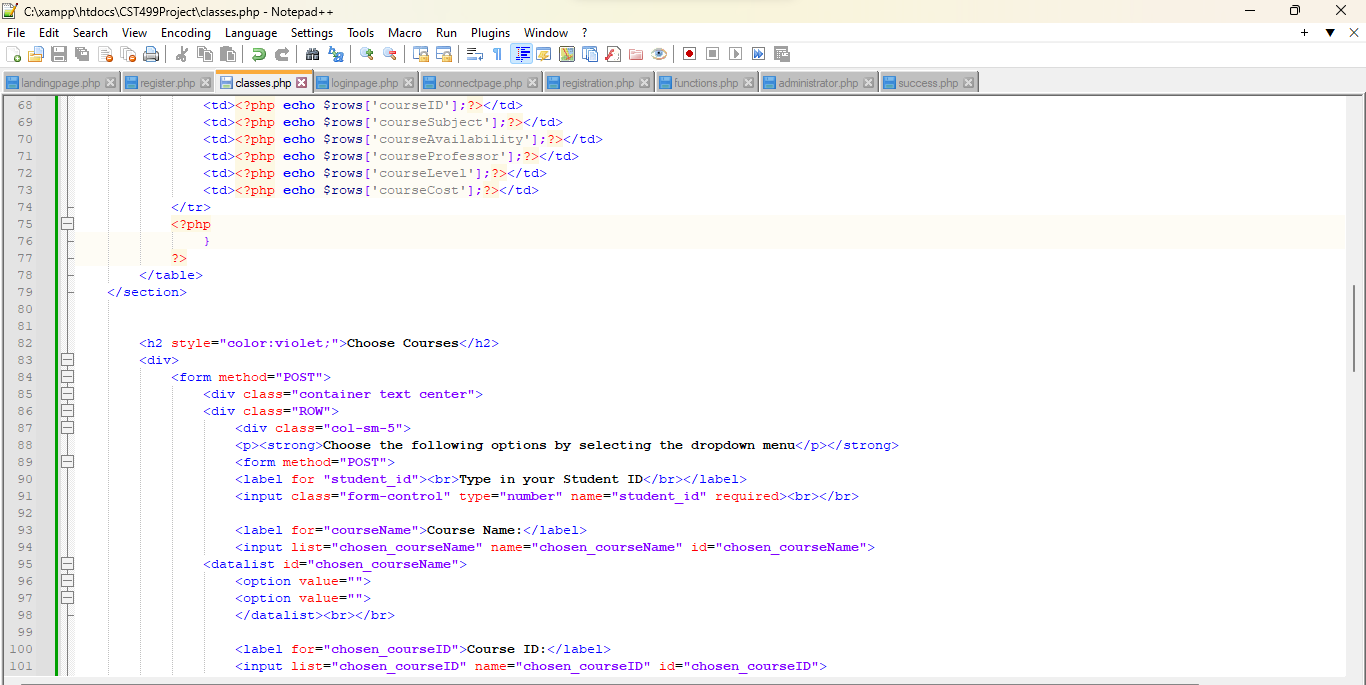
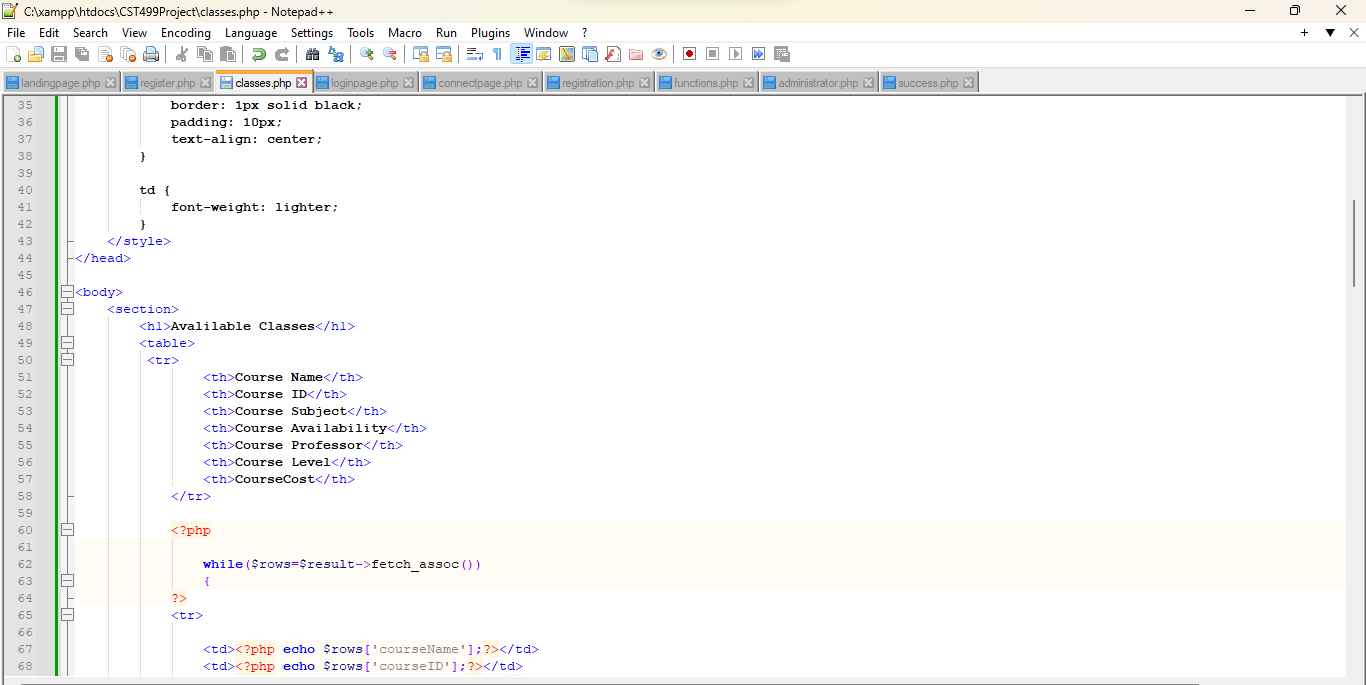
Figure 3: Classes Page

Figure 4: Classes Scripts



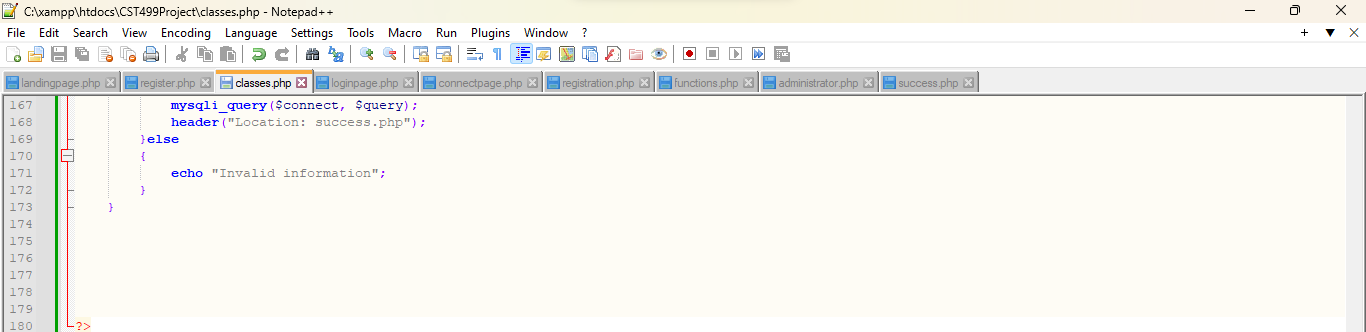
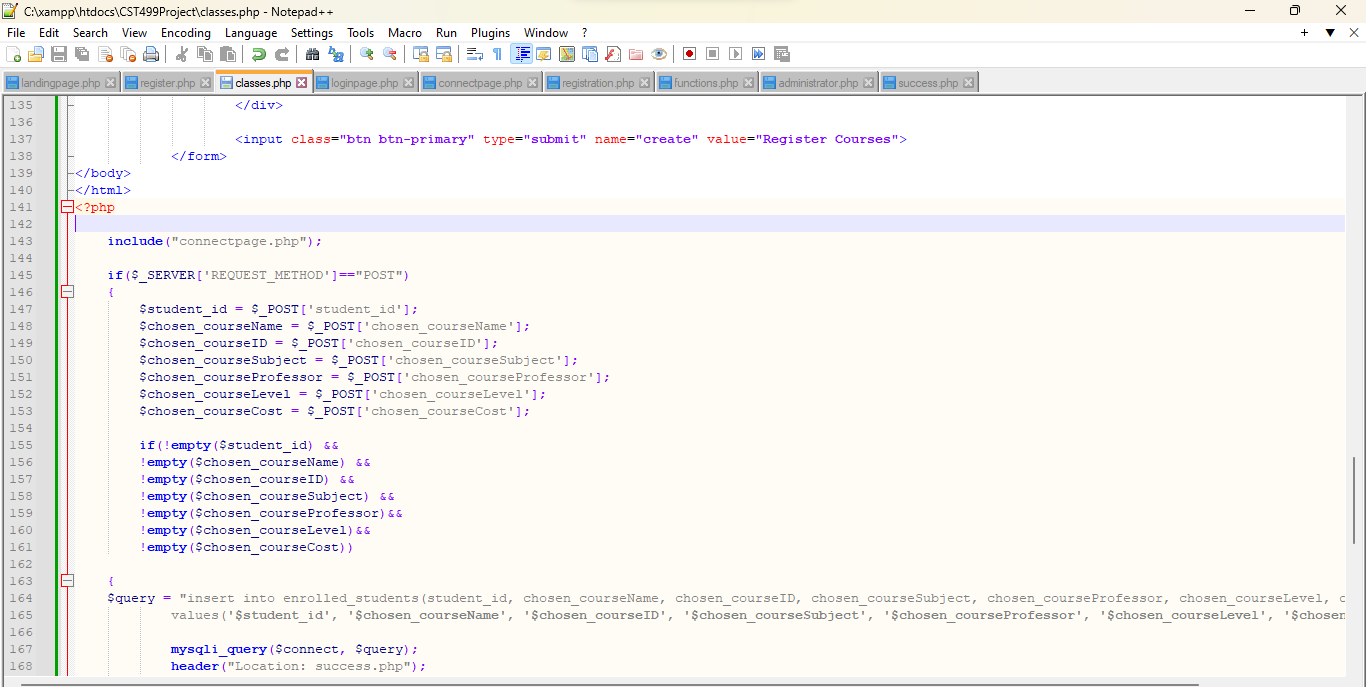
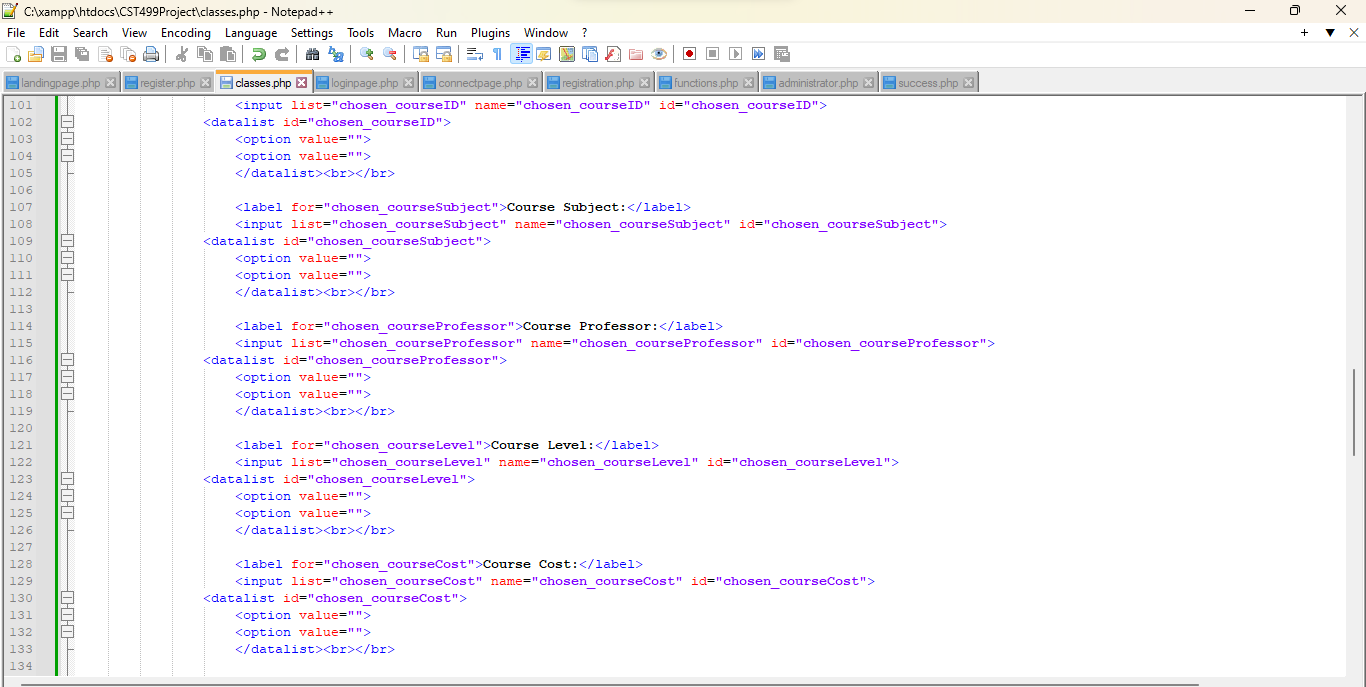


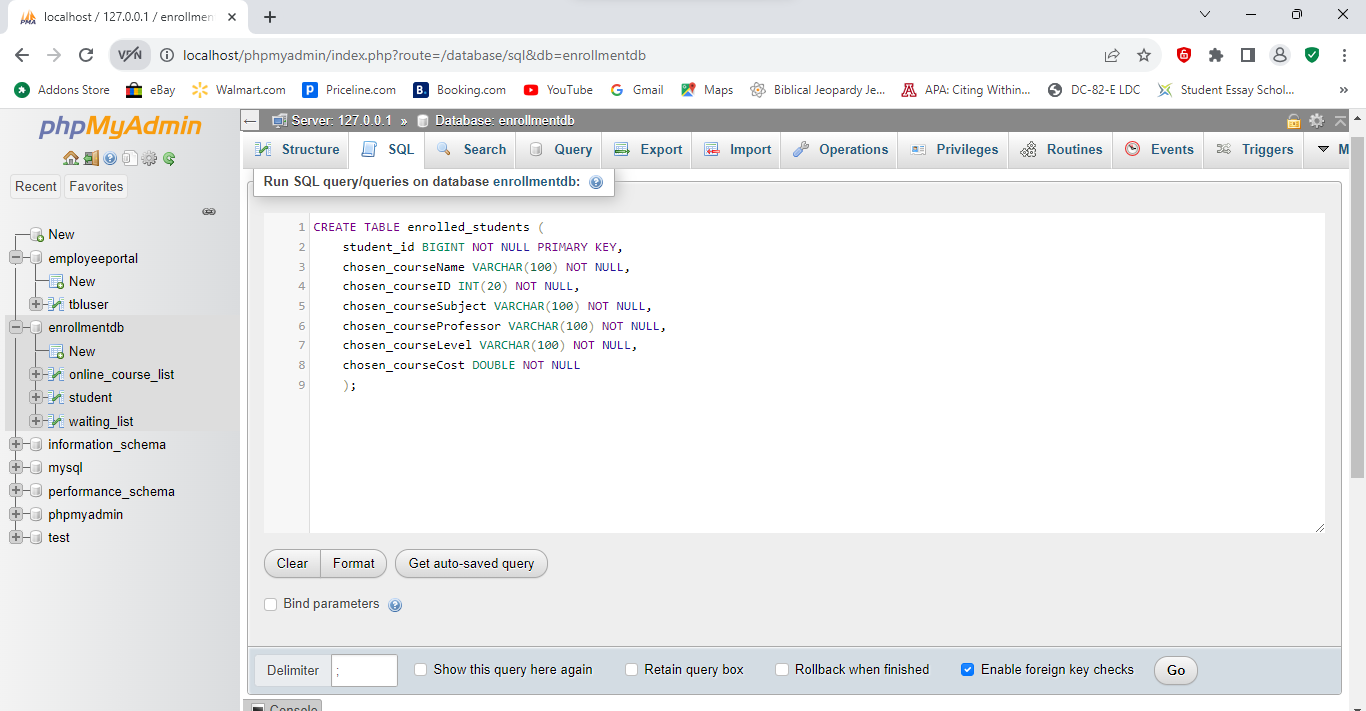
Figure 5: Enrolled Student Table 

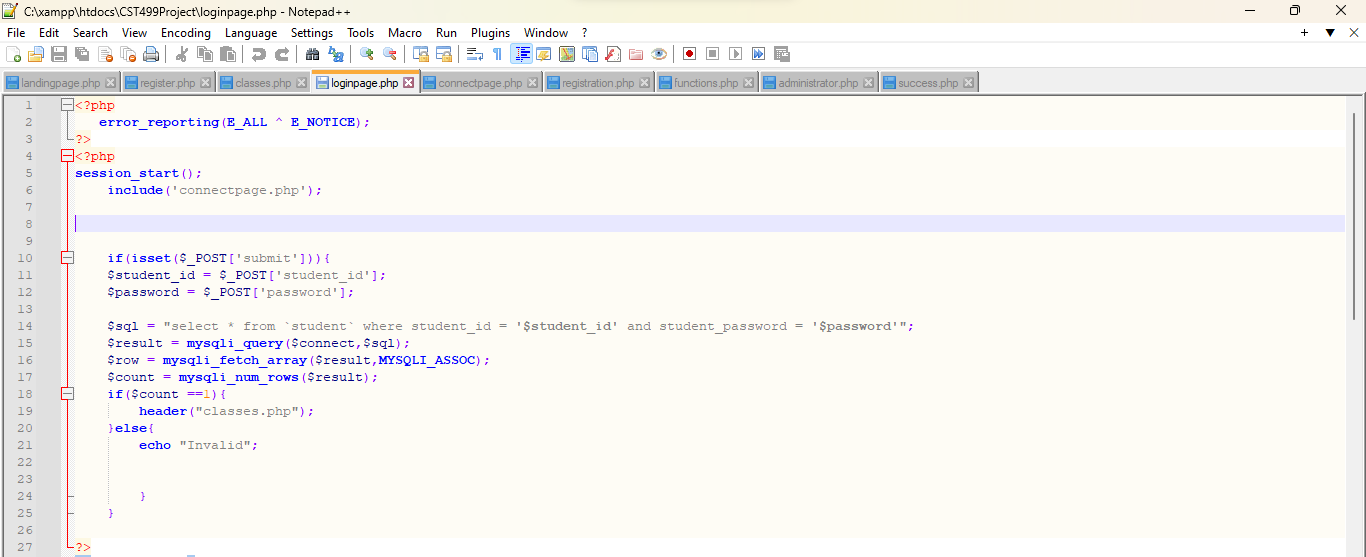
Figure 6: Login Script

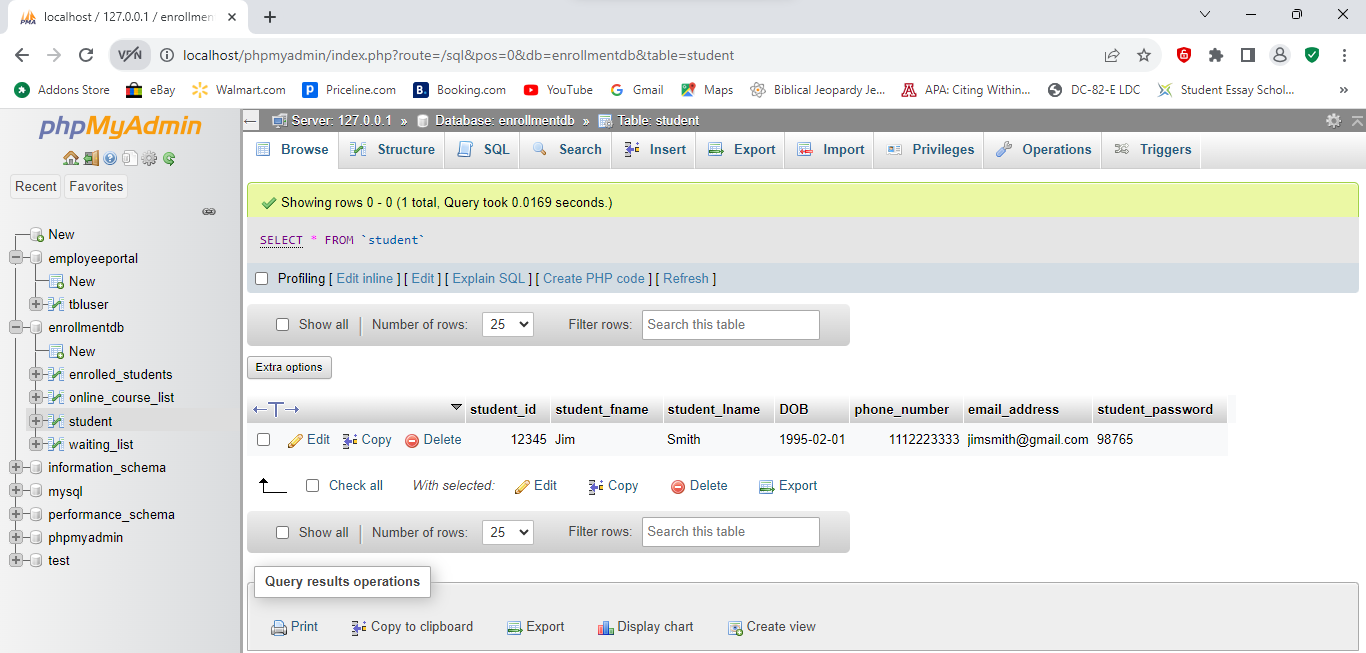
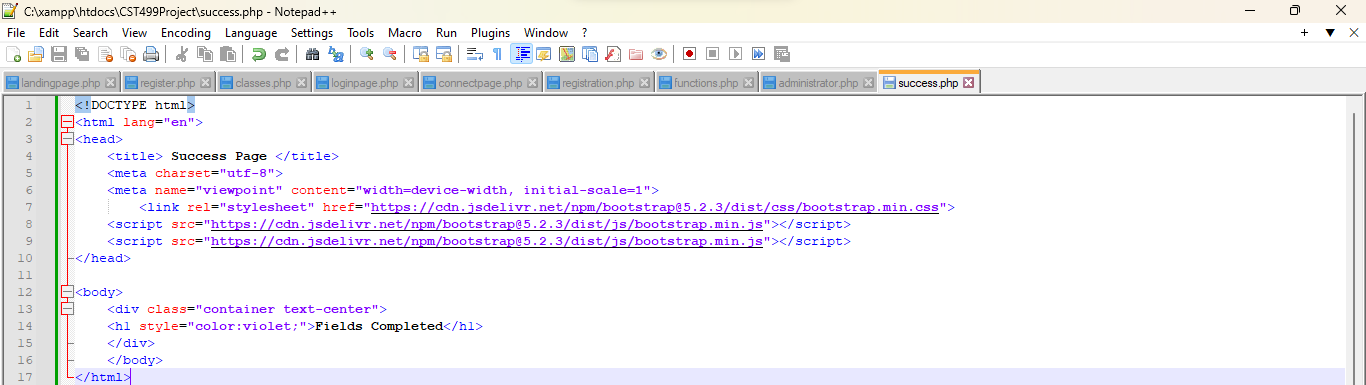
Figure 7: Registration Result

Figure 8: Success Page



**References**

Connolly, R., & Hoar, R. (2018). [*Fundamentals of web development*](https://uagc.instructure.com/courses/122857/modules/items/6249130)(2nd ed.). Pearson.

Odeh, A. H. (2019). Analytical and Comparison Study of Main Web Programming Languages --ASP and PHP. *TEM Journal*, *8*(4), 1517–1522. <https://doi.org/10.18421/TEM84-58>

Tsui, F., Karam, O., & Bernal, B. (2018). [*Essentials of software engineering*](https://uagc.instructure.com/courses/122857/modules/items/6249130)(4th ed.). Jones & Bartlett Learning.