Spring 2024

COMP 5120/6120 DATABASE SYSTEM I

TERM PROJECT

Populating and Querying Databases with SQL

Due: 11:59 PM (Central Time), April 30, 2024

A. PROJECT OVERVIEW

In this project, you need to create a database for a hypothetical online bookstore system. The system maintains information about books, subjects, supplier, and shipping carriers. The bookstore acquires the desired books from suppliers (e.g. Amazon.com), and ships the books by carrier services (e.g. UPS). The system also keeps track of orders, manages customers and employees.

Your tasks are the following:

1. Create and populate the bookstore database with MySQL.
2. Create a simple web interface using PHP and HTML to query the data. 3. Write correct SQL statements for the queries given and use them to test your web service.

B. DETAILS DESCRIPTION

1. Request a personal web space on Auburn University's Mallard server

Your database website will be hosted on Auburn University's Mallard server, and can be displayed by a browser at *auburn.edu/~username*. All Auburn students have a personal space on the university web server (Mallard), which has PHP support. You will need to upload your project PHP files to the web server for the grader to access them. To request a personal space on Mallard, please follow the instructions at
https://auburn.service-now.com/it/en/web-hosting?id=kb_article_view&sysparm_article=KB0000001&sys_kb_id=49cf4203db69fd10f2d4e7dcd39619e1&spa=1
(or simply search for "Auburn University Mallard server").

Mallard server also include MySQL/SQL. For your website, you will have a database created on MySQL server maintained by OIT. To request a MySQL database, please refer to: https://cws.auburn.edu/oit/database/mySQL/Create. *It is important to note that your database password **SHOULD NOT** be the one used for your AUAccess.*

Additionally, the request may take at least two days for the OIT to approve. Request this as soon as possible when you are starting this project. If there is a problem with the requests, e-mail the OIT Office or Mark Bransby at bransby@auburn.edu.

2. Connect to Auburn University's Mallard server and Access to your MySQL database

Once the requests are approved, you can log into Mallard server with your Auburn username/password using SecureCRT (downloadable from auburn.edu/download) or WinSCP. For Mac/Linux users, you can ssh to the server (mallard.auburn.edu) using Terminal.

To access to MySQL database via MySQL Workbench, ssh or PHP Script. More details can be found at https://auburn.service now.com/it?id=kb_article_view&sysparm_article=KB0000188&sys_kb_id=5900b99c1be fad50f9cffeaf034bcb4e&spa=1.

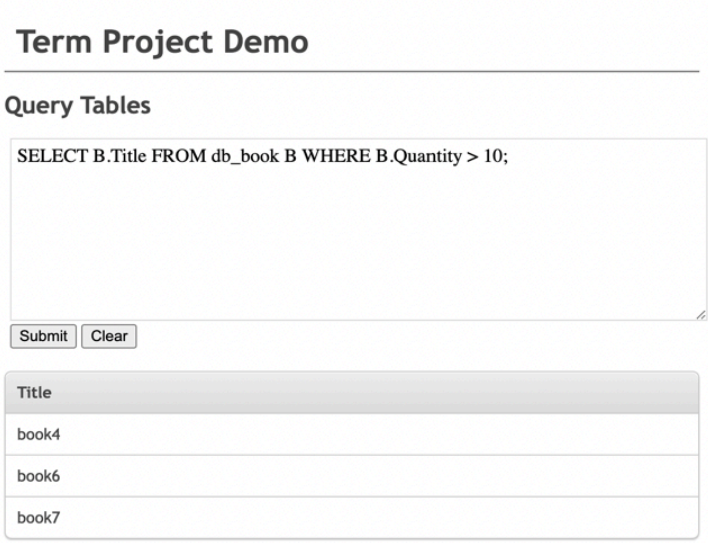
Note that if you are using these services from off campus or via *AU_Wifi*, you must be on the VPN (This requires that Duo has been set up). For install AU Global Protect VPN, visit: https://libguides.auburn.edu/vpn .

3. Populating the Database (20 points)

Create the following tables for the bookstore database and populate the database from the **data.zip.** You may optionally first create your user interface and populate the database through your interface.

4. Interface Implementation (30 points)

Implement an interface by using PHP and HTML. A simple example is as shown in the following figure, which includes a text box to accept a SQL statement and then submit it to the MySQL database.



Interface Requirements:

1. Your interface should **not** accept SQL DROP statements.

2. For any other SQL statements, your interface should not only accept it, but also return the execution result. For example, a select statement will return the query results (including the attribute name for each column) and **the number of rows retrieved**. A create/delete/update/insert statement will display "Table Created/Updated", "Row Inserted" or "Row(s) Deleted" messages on your interface.
3. An error message should be displayed if an incorrect SQL statement was submitted.
4. You should also have a title for this interface, indicating your name.

5. Execution of SQL queries on your website (50 points)

First, you need to write the SQL statements for the following queries (can also be found in *query.txt* in **data.zip**). Then, you submit each of them through your interface to get the correct result.

1. Show the subject names of books supplied by *supplier2*.
2. Show the name and price of the most expensive book supplied by *supplier3*.
3. Show the unique names of all books ordered by *lastname1 firstname1*. 4. Show the title of books which have more than 10 units in stock. 5. Show the total price *lastname1 firstname1* has paid for the books. 6. Show the names of the customers who have paid less than $80 in totals. 7. Show the name of books supplied by *supplier2*.
8. Show the total price each customer paid and their names. List the result in descending price.
9. Show the names of all the books shipped on 08/04/2016 and their shippers' names.
10. Show the unique names of all the books *lastname1 firstname1* and *lastname4 firstname4* *both* ordered.
11. Show the names of all the books *lastname6 firstname6* was responsible for. 12. Show the names of all the ordered books and their total quantities. List the result in ascending quantity.
13. Show the names of the customers who ordered at least 2 books. 14. Show the name of the customers who have ordered at least a book in *category3* or *category4* and the book names.
15. Show the name of the customer who has ordered at least one book written by *author1*.
16. Show the name and total sale (price of orders) of each employee. 17. Show the book names and their respective quantities for open orders (the orders which have not been shipped) at midnight 08/04/2016.
18. Show the names of customers who have ordered more than 1 book and the corresponding quantities. List the result in the descending quantity.
19. Show the names of customers who have ordered more than 3 books and their respective telephone numbers.

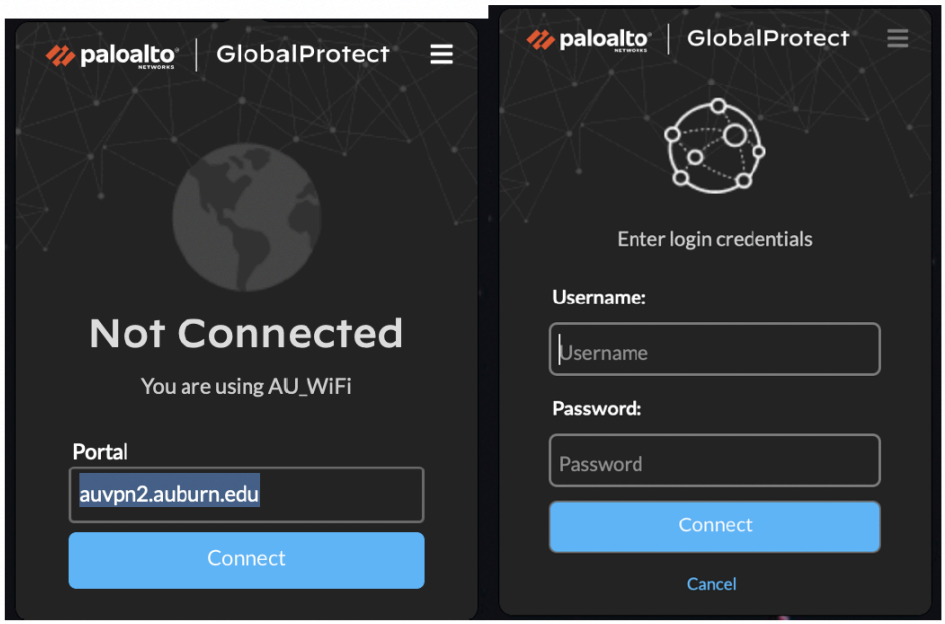
6. Materials to Hand In

Please submit the following in a zip file named **YourAuburnUsername.zip** to canvas with the subject **COMP5120/6120 Term Project** by deadline.

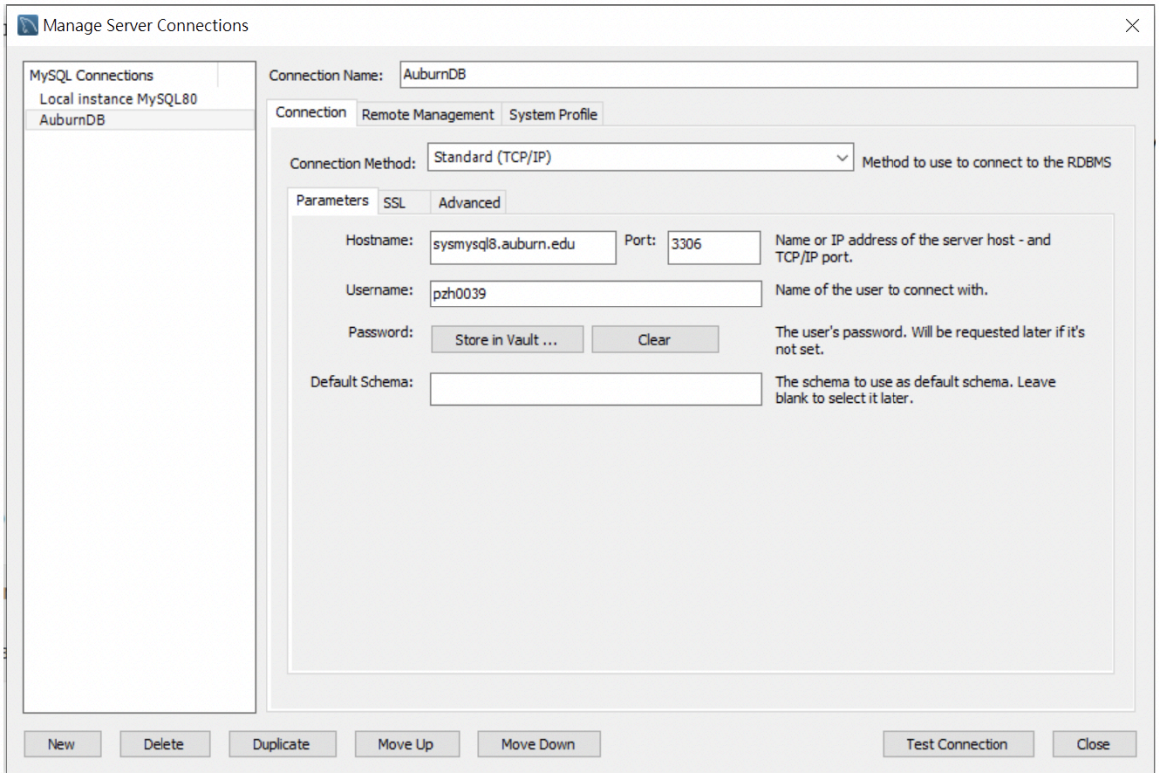
1. URL of your PHP/HTML interface in a file named **url.txt**. Please make sure the interface can be used to query the database. The TA will use the interface to verify that the database has been populated with correct schema and data.
2. Your code organized in **src/** folder for implementing the interface. This folder includes the index.php file and other files of your website.
3. The execution of SQL queries in **sql.txt**. Please have queries numbered and in the same order as queries given.

C. USEFUL TIPS

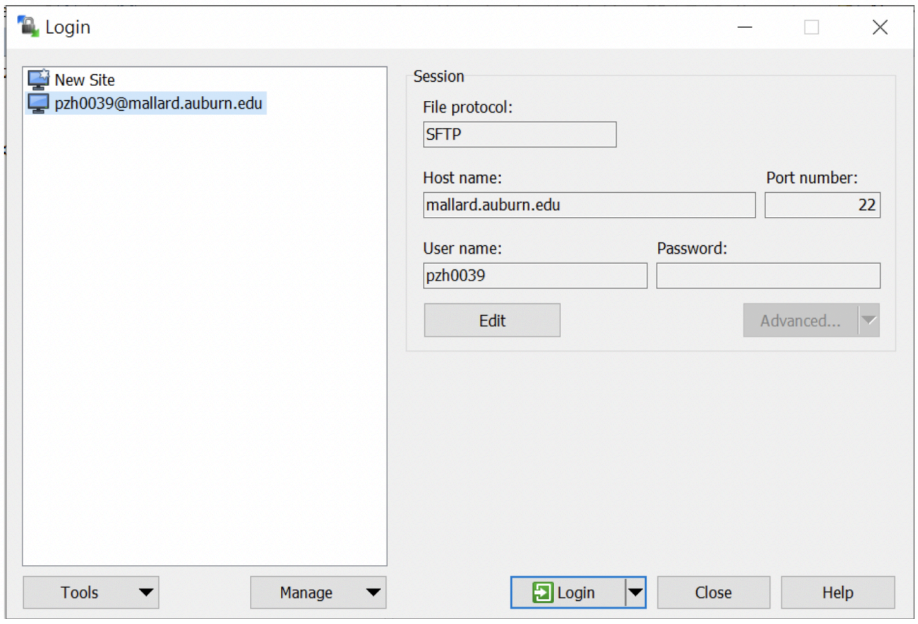
1. Connect to AU Global Protect VPN



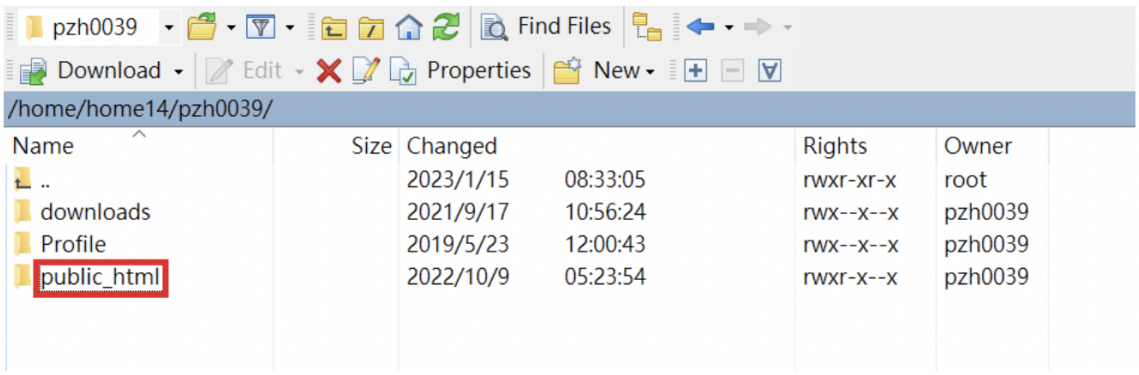
2. Connect to MySQL Workbench



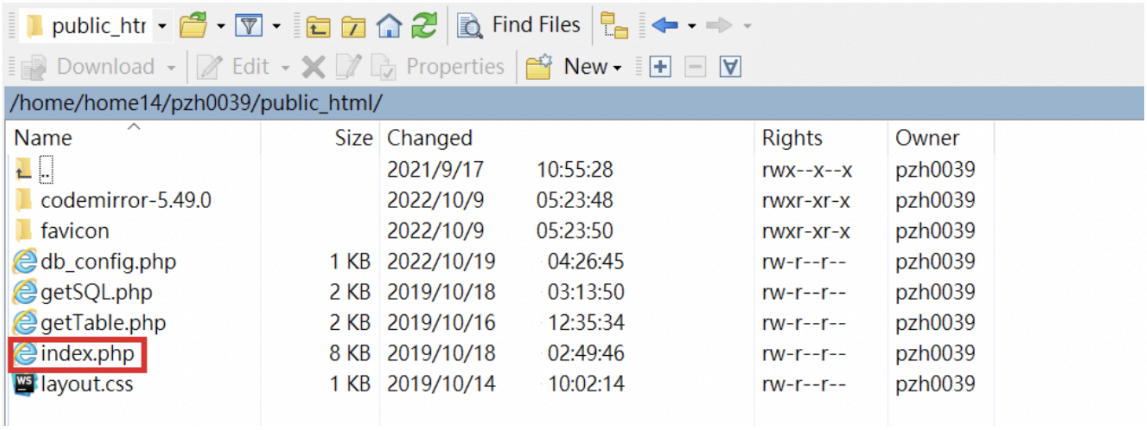
3. Login to Web server with WinSCP



4. Your folder on Mallard server



5. Upload your PHP files onto /home/home$$/$AUusername/public_html



Note that your interface code must be named *index.php*. WinSCP allows you to upload your code to the server. For Linux/Mac users, you can use *scp* command to upload your scripts to the server from your local machine (e.g., *scp index.php $AUusername @mallard.auburn.edu: /home/home$$/$AUusername/public_html*).

5. Useful self-study resources

https://www.w3schools.com/php/

Robin Nixon, *Learning PHP, MySQL & JavaScript*, 6th Edition.