Assignment 4: The Online Food Ordering Database -Task 4

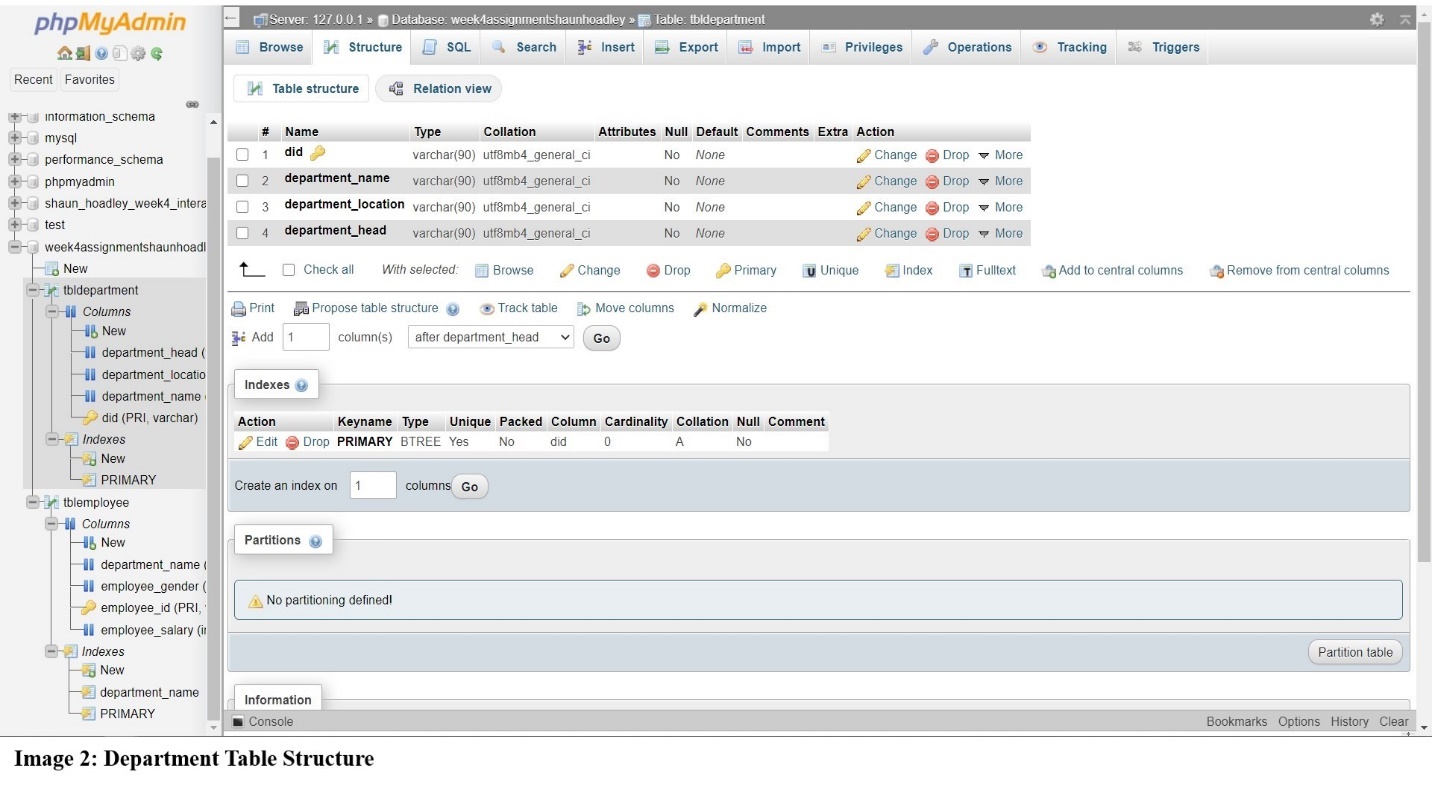
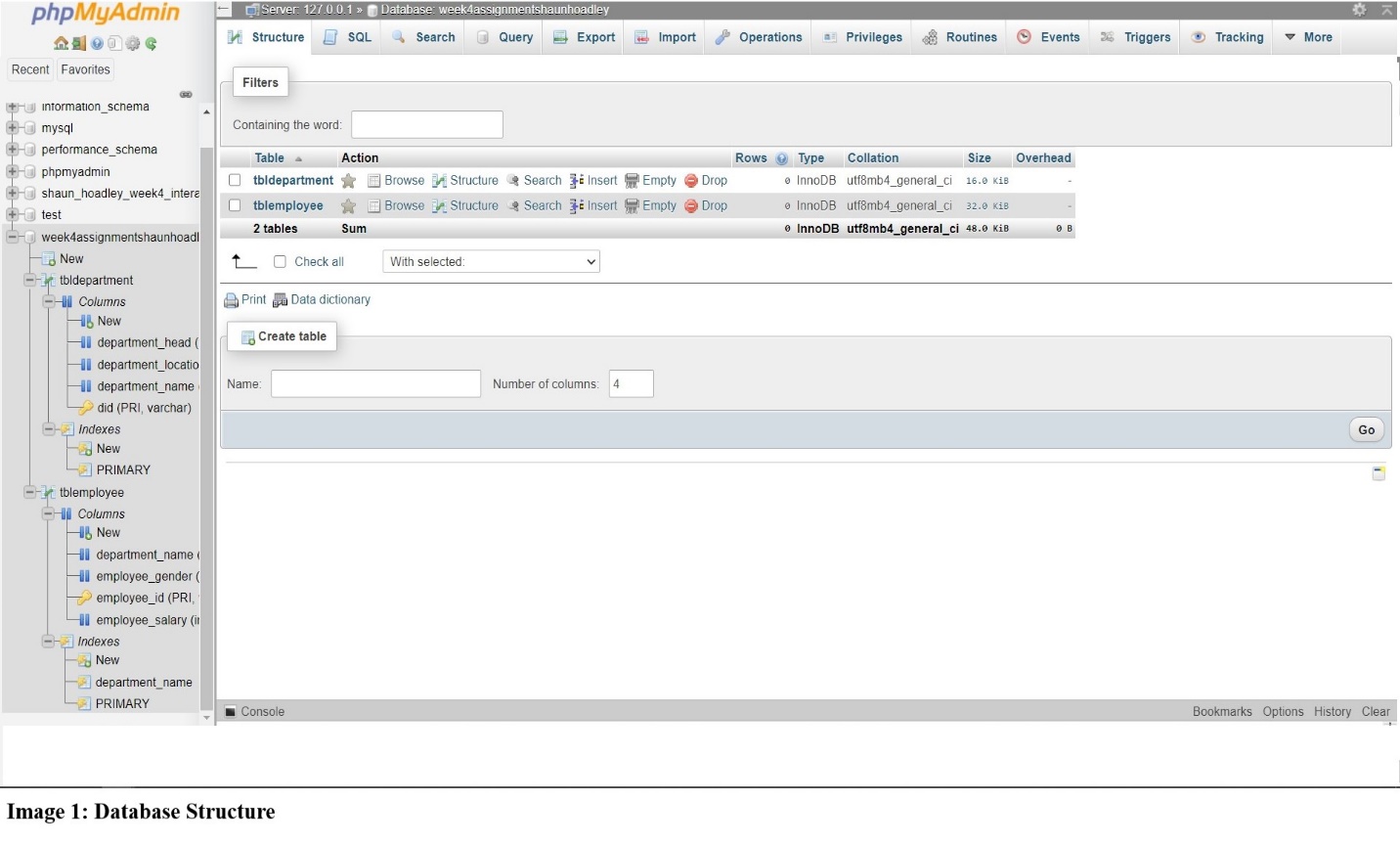
Shaun Hoadley

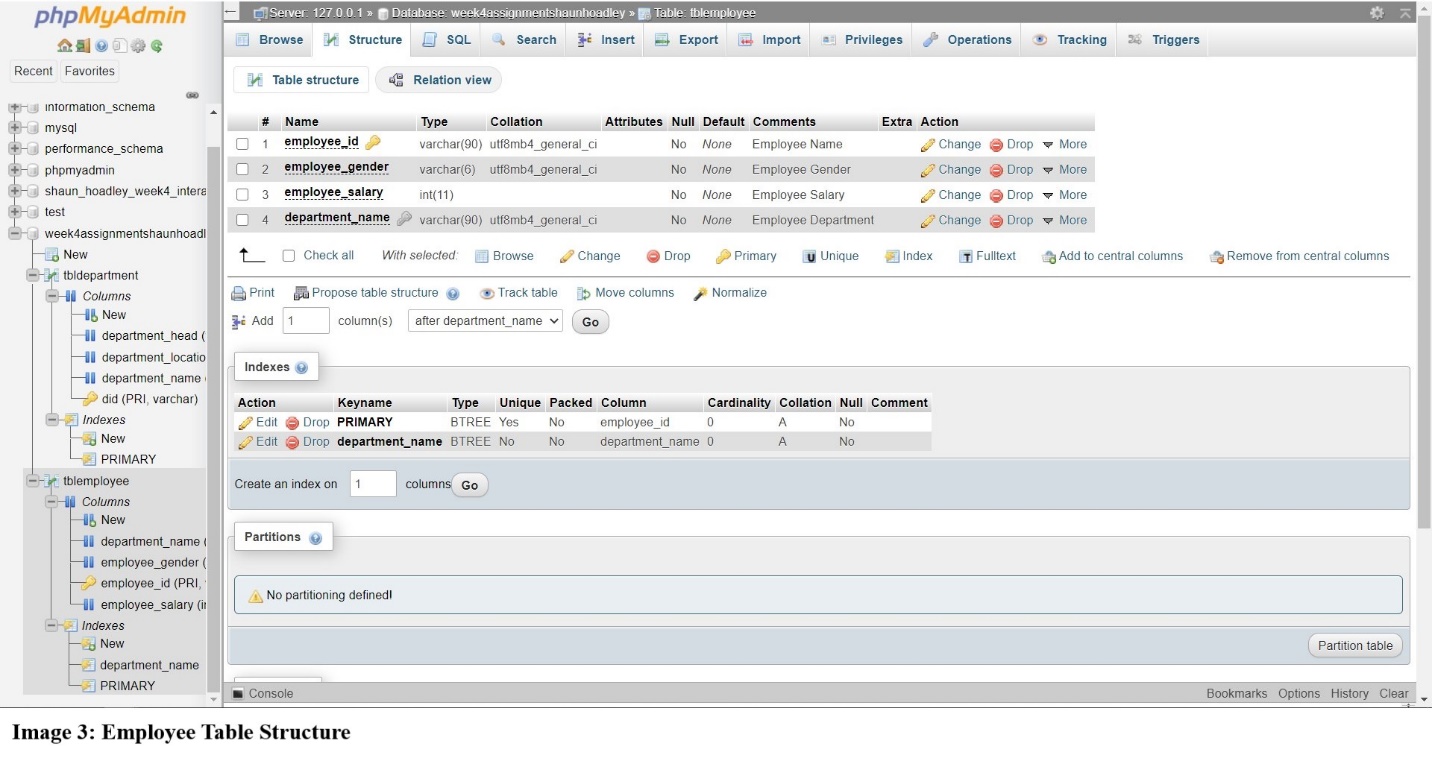
CPT310: Database Systems & Management

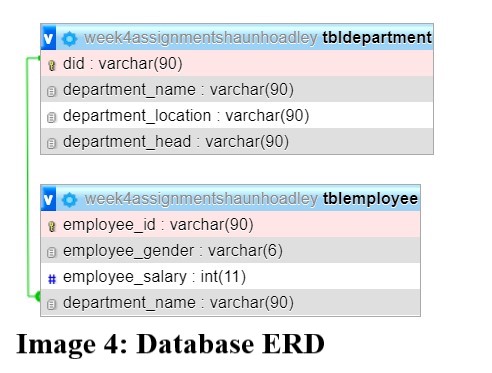
Austine Ohwobete

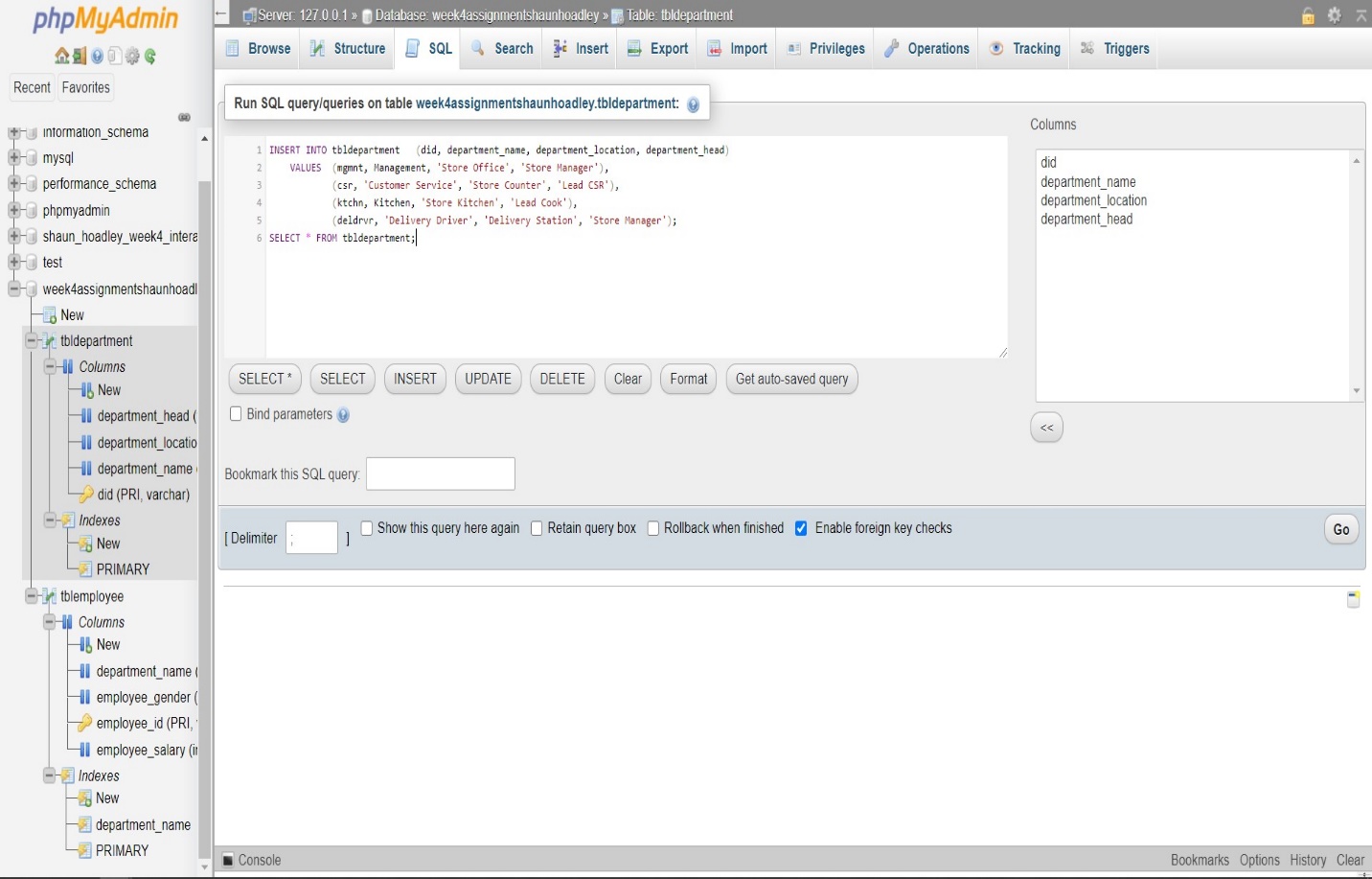
November 23, 2020

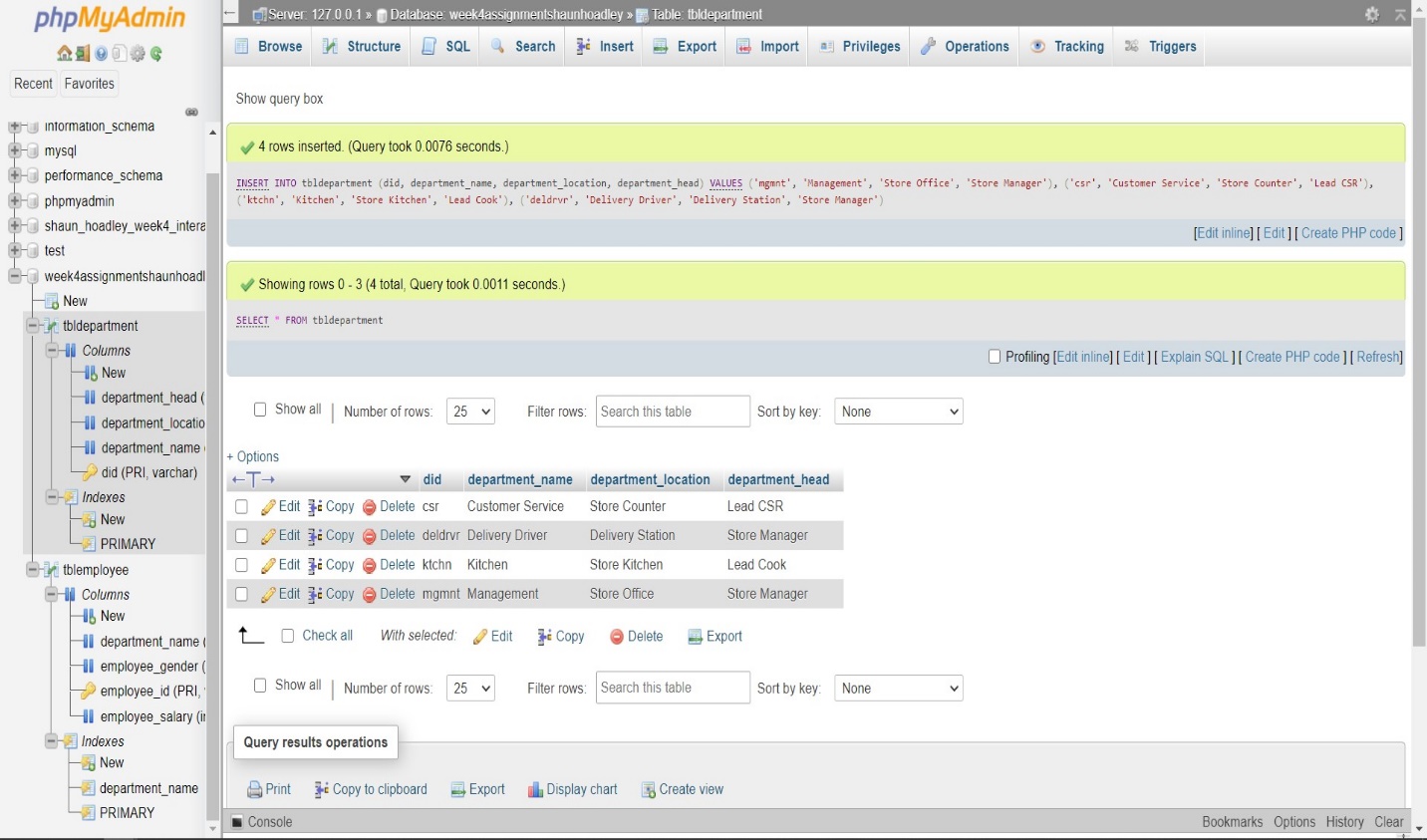
**Task 4 Database Screenshots**

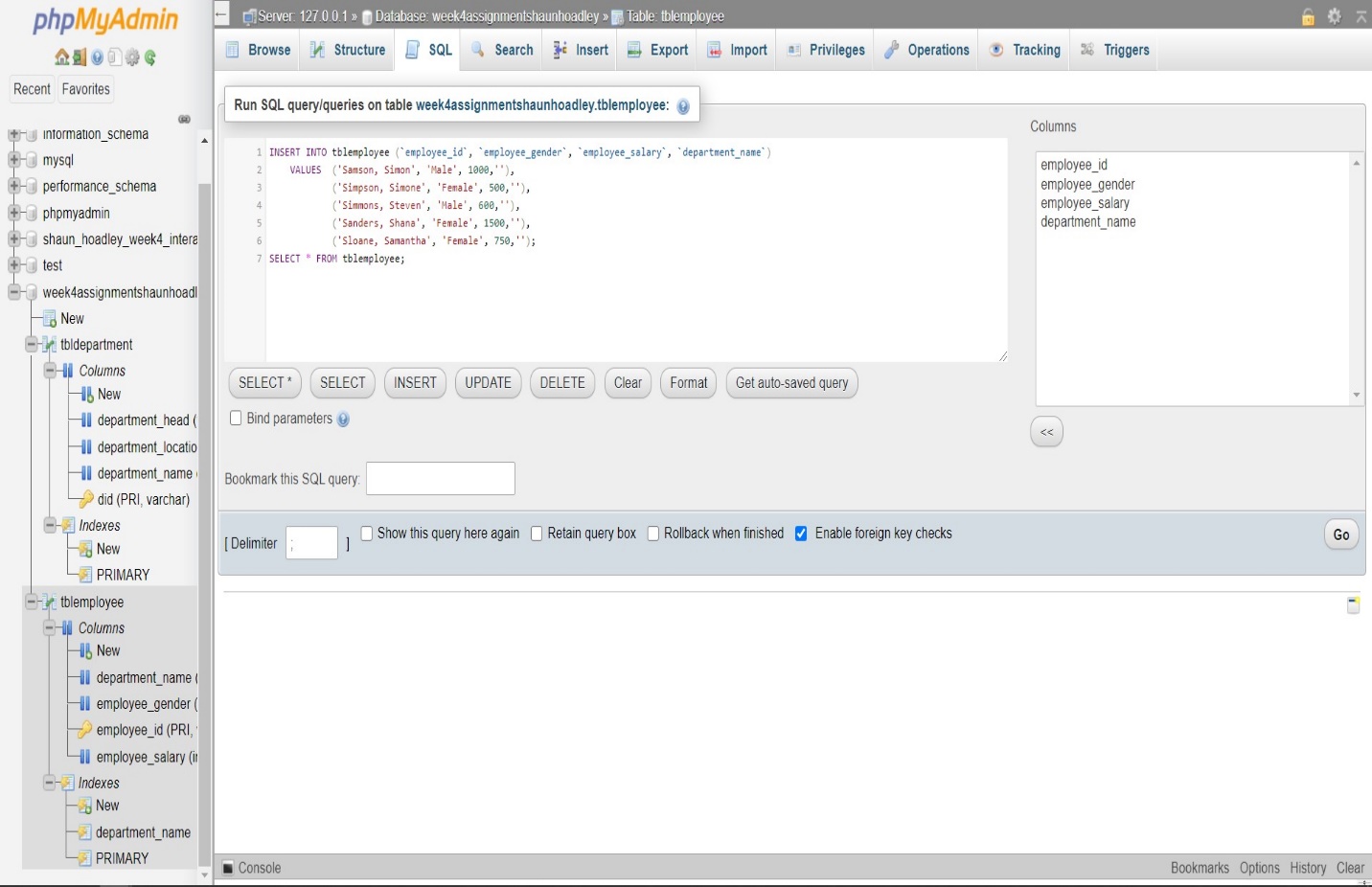


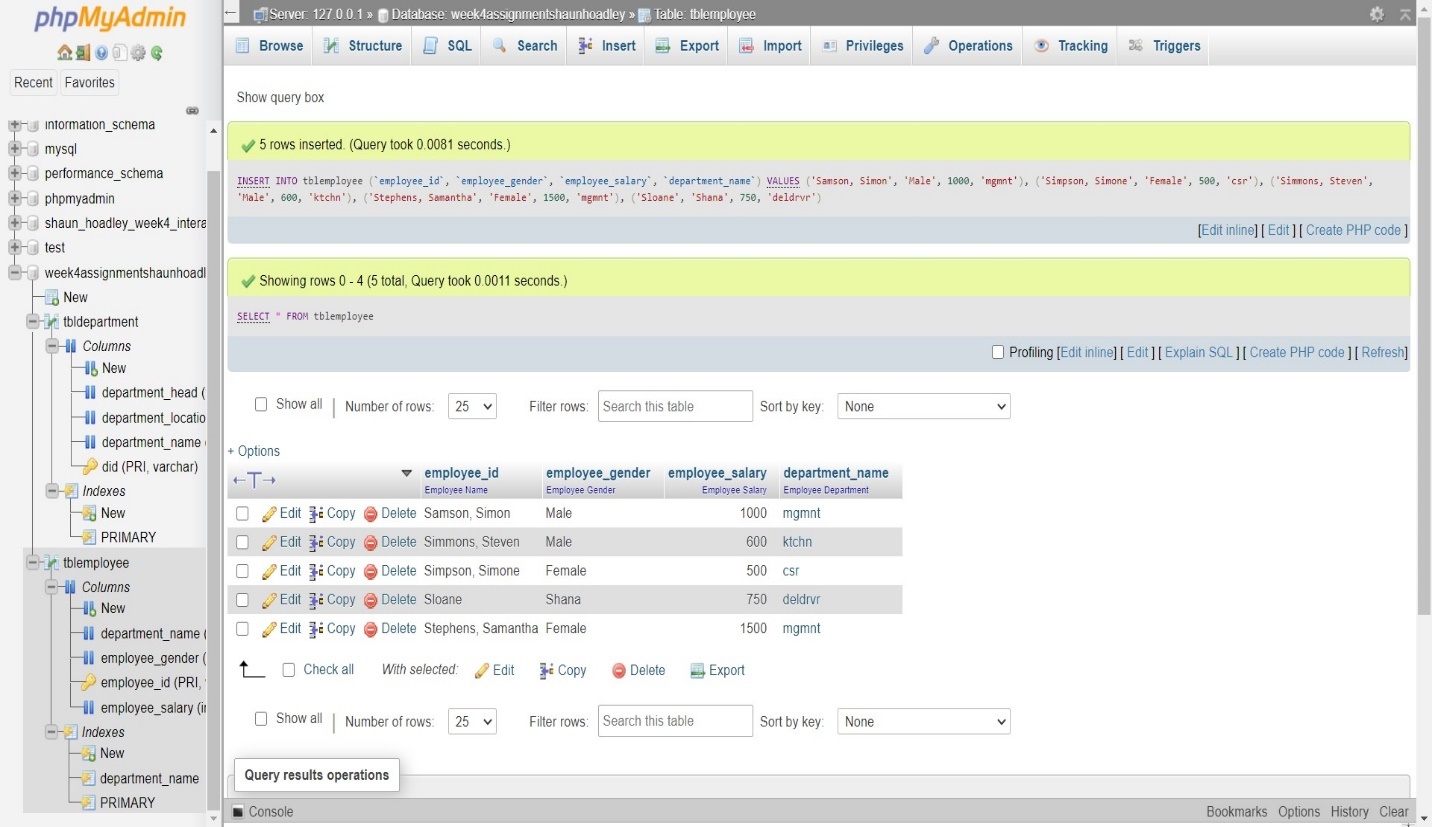




**Image 5: Insert Department Data Via SQL Commands**

**Image 6: Populated Department Table**

 **Image 7: Insert Employee Data Via SQL Commands**

 **Image 8: Populated Employee Table**

**Data Definition Language (DDL) and Data Manipulation Language (DML)**

Data Definition Language (DDL) is a part of the Structured Query Language (SQL) that we use in programming computer relational databases. Data definition language is, according to Coronel and Morris, “The language that allows a database administrator to define the database structure, schema, and subschema.” (2019). The instructions for this paper say “Describe Data Definition Language (DDL) insert, update, delete.” Insert, update, and delete are, however, DML (Data Manipulation Language) commands. Coronel and Morris describe data manipulation language as a “set of commands that allow an end user to manipulate the data in a database.”

The first command we will look at is the INSERT command. The insert command is used to add rows to a table. The format for using the insert command is “INSERT INTO *tablename* VALUES (value1, value2,…, valueN).” The “valueN” labels correspond to each of the columns in the table. For example, you wish to populate a row of data for the table *cars* that has the columns *year, make, model, and color*. To add a row for a 2011 silver Chevrolet HHR into the *cars* table would look like “INSERT INTO cars VALUES (2011, Chevrolet, HHR, silver);”

The next command to look at is the UPDATE command. The update command is for updating existing information in a table. The format to use the update command is “UPDATE *tablename* SET *coulmnname = newdata* WHERE *conditionlist*;”. Additional SETs can be added by separating them with a comma as in the following example. You receive a delivery order of items already listed in your movie inventory database and one of them, six DVDs of Ghostbusters, also has a price reduction from $9.99 to $7.99. The proper format to enter the changes would be “UPDATE movie\_inv SET qty = qty + 6, price = 7.99 WHERE movie\_id = GhostbustersDVD;”. It is vital to note, if you do not include the WHERE, it will change the qty and price of ALL of the rows in the table.

Next we come to the DELETE command. As the name implies, the DELETE command removes a row from a table. The DELETE command will not remove the table even if you remove all of the rows, that would require the DROP command. The format for deleting a row is “DELETE FROM *tablename* WHERE *conditionlist*;”. The WHERE is extremely important unless you are deleting all of the rows of the table. While some RDBMSs will alert you to be certain that you are sure you wish to proceed, not all do or the option may be turned off. As an example, you are a teacher and are cleaning up your student\_list database. You entered all of the students you are supposed to have before school starts. When school begins you discover a few students were added and removed for various reasons. You want to remove Billy Madison from your table, so you use “DELETE FROM active\_students WHERE stu\_name = ‘Billy Madison’;”.

The next command we will look at is the INNER JOIN. The INNER JOIN is a query that joins all of the columns between tables that match. Any columns that do not match will not be shown. The format for the inner join command is “SELECT *column names* FROM *table1* INNER JOIN *table2* ON *table1.columnname* = *table2.columnname;”.*

The SUM command is used to add all of the values of a column. The syntax for using SUM is “SUM (*expression*)”. As an example, you wish to get the sum of the column qty from the order\_details table and display it as total\_items\_ordered, “SELECT SUM (qty) AS total\_items\_ordered FROM order\_details;”.

Lastly, we have the ORDER BY command. The order by keyword in SQL is used to sort the results of a query. The results can be sorted in ascending or descending order, but the default is ascending order. The syntax for the use of ORDER BY is “ORDER BY *column1, column2, columnN* ASC|DESC;” For example, you have a customer database and you wish to see a list of customers in order by state from z to a. This query would look something like this, “SELECT \* FROM customers ORDER BY state DESC;”.

**References**

Coronel, C., & Morris, S. (2019). [Database systems: Design, implementation, and management](https://ashford.instructure.com/courses/74798/modules/items/3783232) (13th ed.). Retrieved from <https://www.vitalsource.com>