COMP 2350

Lab Report:

SELECTs Lab

By Yeseol (Sol) Kim

Submitted to: Patrick Guichon

Introduction:

Today, in the lab, I practiced creating MySQL databases and tables using commands. I was able to create new files using "create database" or "create table" and delete them using "drop database" or "drop table." It was very convenient as we could also specify the values and data types for each column all at once. Additionally, I learned to create backup files for the database, and I could later import these backup files. During this process, I was able to generate complex database Entity-Relationship Diagrams (ERDs) for a quick overview.

Next, I focused on scenarios, where we could manipulate data based on specific commands. For example, I could sort data in ascending or descending order, and combine sorting by last name and first name to display data by names. This allowed us to organize the data according to our needs using various commands.

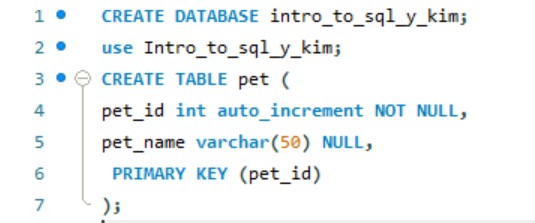
Finally, I connected MySQL with Render that I worked on last week, enabling us to directly send data to Render. However, modifying values in each EJS and JS file proved to be a bit challenging. Through multiple Git pushes, I identified and rectified errors in the process.

Screenshot #1:

This is how to create database and give the primary key and value of each column.

While commend can add datatype as well.

Filename: 01\_Create\_Pet\_Table\_Command.png



Screenshot #2:

*INSERT INTO person (person\_id, first\_name, last\_name, phone\_number, birth\_date)*

*VALUES*

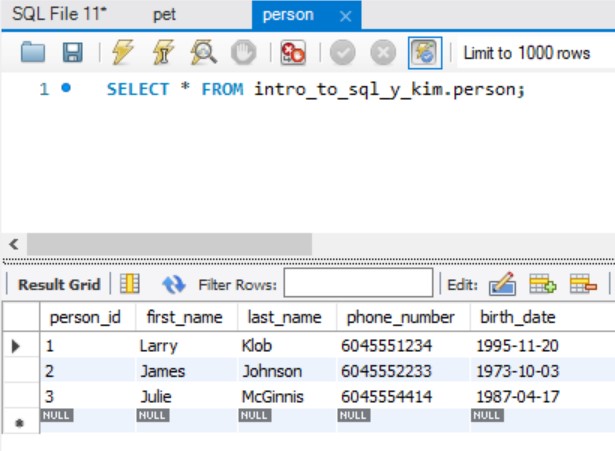
*(1, 'Larry', 'Klob', '6045551234', '1995-11-20'),*

*(2, 'James', 'Johnson', '6045552233', '1973-10-03'),*

*(3, 'Julie', 'McGinnis', '6045554414', '1987-04-17');*

This commend allowed to insert the value in columns in person table.

Filename: 02\_Select\_From\_Person.png



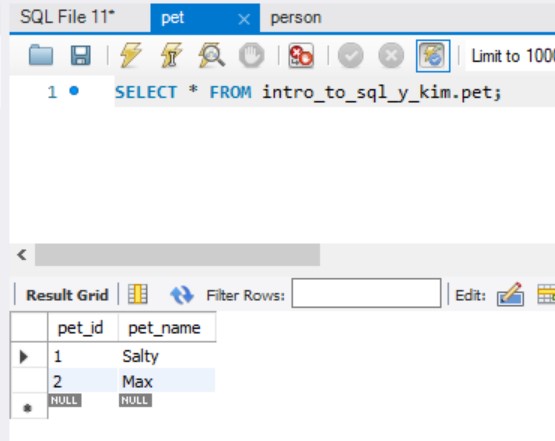
Screenshot #3:

The SQL SELECT command is used to retrieve data from one or more tables in a database. It does not select the table itself but rather selects specific columns or rows from the table.

*SELECT column1, column2 FROM your\_table;*

*SELECT \* FROM your\_table;*

Filename: 03\_Select\_From\_Pet.jpg



Screenshot #4:

This is the file I export the pet and person tables as a backup.

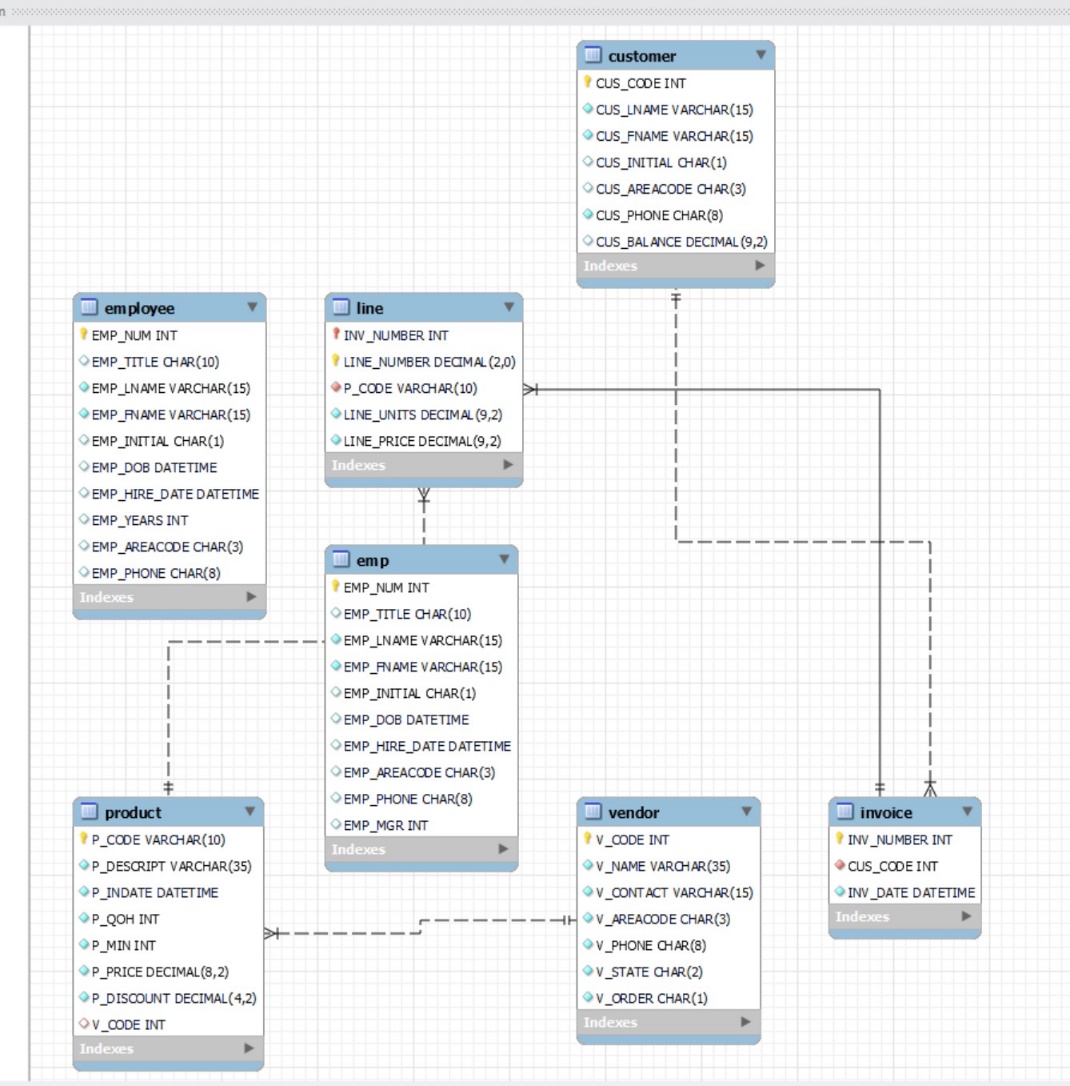
Filename: 04\_Person and Pet Backup (Logical).sql



Screenshot #5:

This is after import the sql file to Mysql workbench and show the ERD.

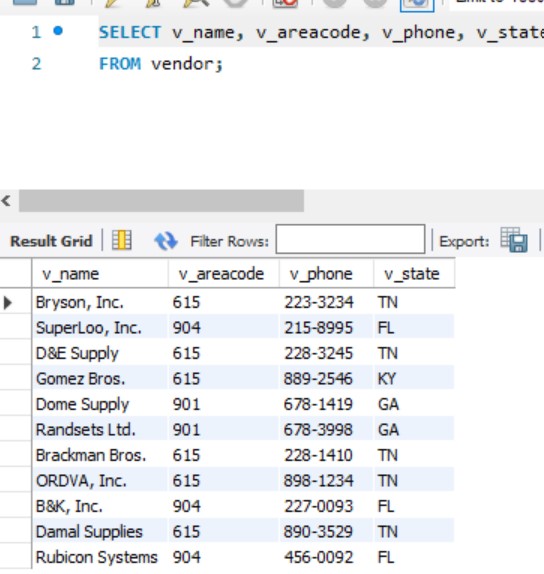
Filename: 05\_SaleCo\_ERD.jpg



Screenshot #6:

Screenshot 6 to 19, each representing different methods to retrieve data from an imported file, and you want to comment on the SQL SELECT statements with specific column names for displaying the data as a table.

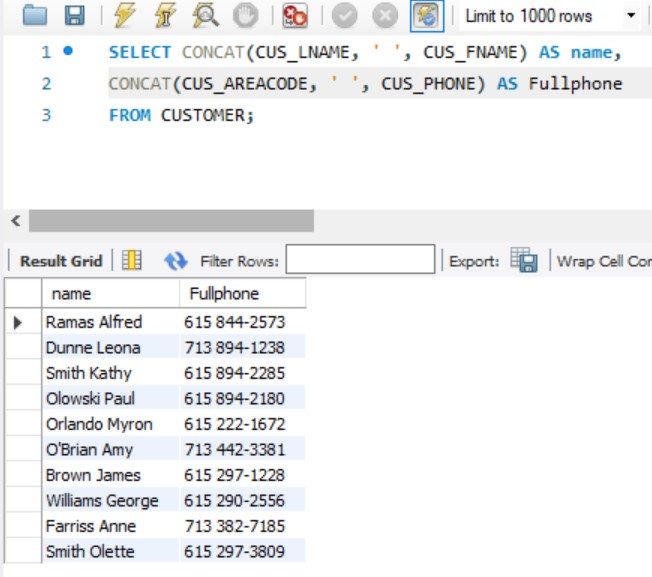
Filename: 06\_Scenario\_2.jpg



Screenshot #7:

The CONCAT function is used to combine multiple columns into a unified column. This results in a single column that incorporates the values from the specified columns.

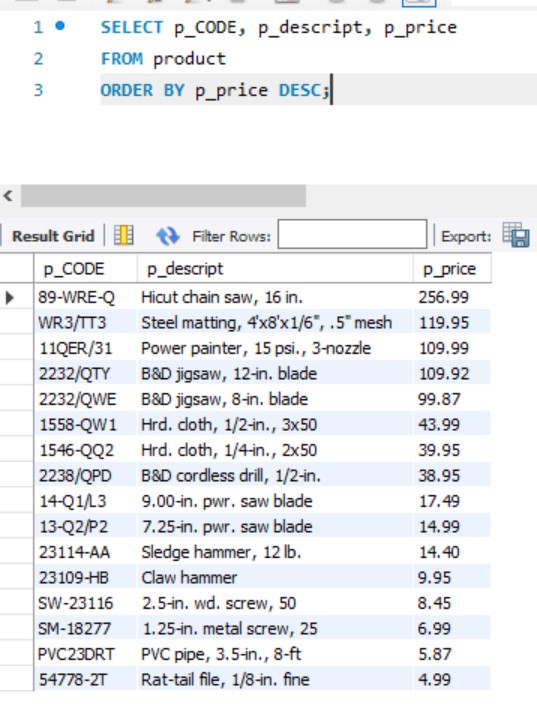
Filename: 07\_Scenario\_3.jpg



Screenshot #8:

It also has the capability to sort the data according to your preference. The default order is ascending. DESC means decending.

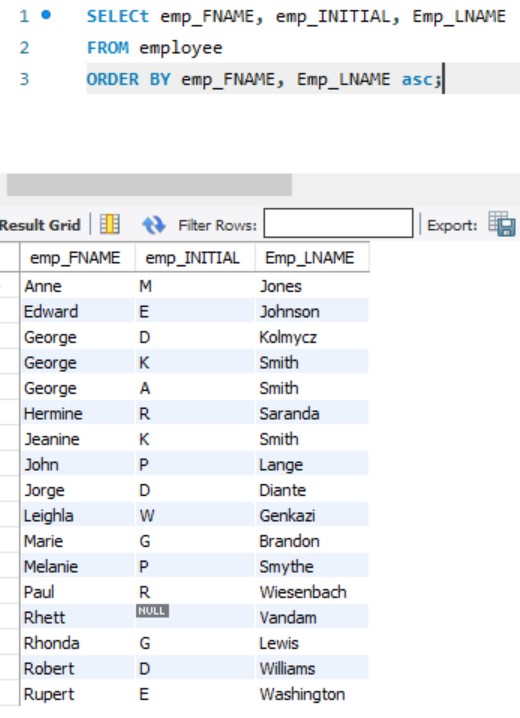
Filename: 08\_Scenario\_4.jpg



Screenshot #9:

I selected employee first names, initials, and employee last names to create a table, and sorted them in ascending order. The default in MySQL is ascending order, but I specified 'ASC' to organize them in ascending order explicitly.

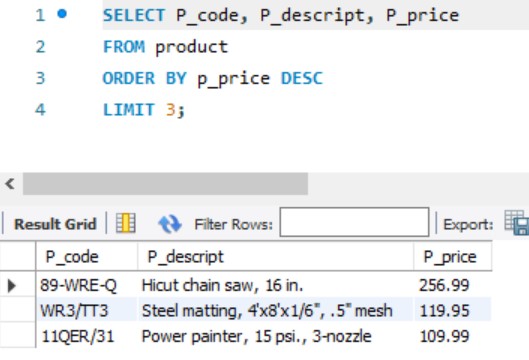
Filename: 09\_Scenario\_5.jpg



Screenshot #10:

I selected product code, product description, and product price from the product table, and instructed to sort the prices in descending order. Using the LIMIT, I displayed the top three products.

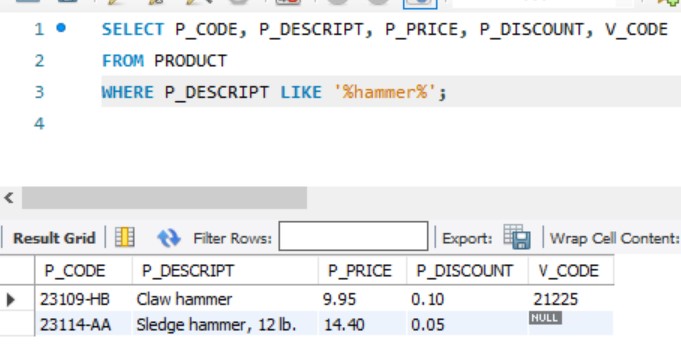
Filename: 10\_Scenario\_6.jpg



Screenshot #11:

Using WHERE allows for detailed filtering based on a specific column. By employing LIKE '% %', you can extract table values that include the specified pattern

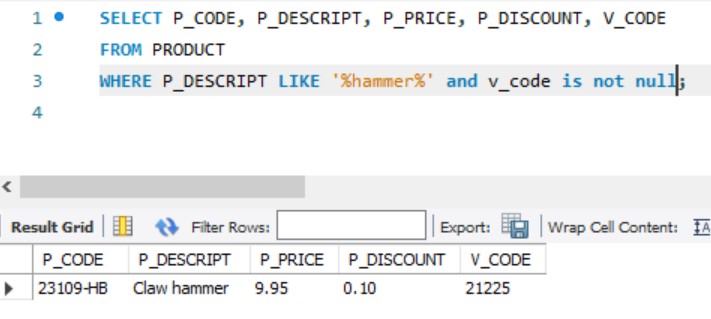
Filename: 11\_Scenario\_7A.jpg



Screenshot #12:

This is an additional item in the screenshot labeled 11. 'IS NOT NULL' indicates a table with existing values. It displays only the products where the V\_code is present, and the description includes the term 'hammer.

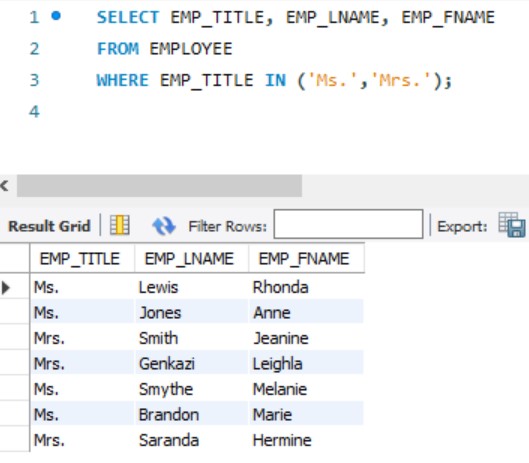
Filename: 12\_Scenario\_7B.jpg



Screenshot #13:

The utilization of the IN operator showcases the table that includes the specified values. It effectively filters and presents the relevant data based on the specified criteria.

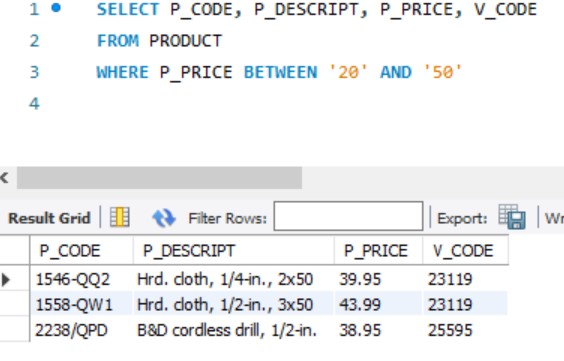
Filename: 13\_Scenario\_7C.jpg



Screenshot #14:

The BETWEEN operator defines a range of values by specifying a minimum and maximum. It displays values within that range, inclusive of both the minimum and maximum values.

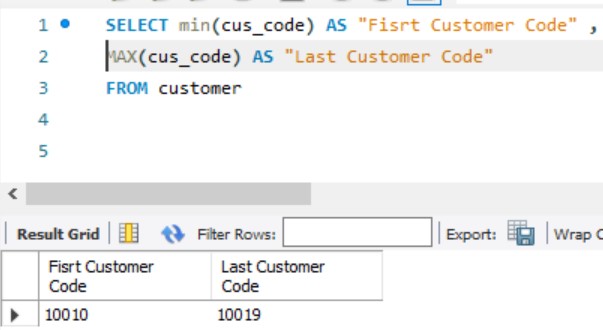
Filename: 14\_Scenario\_7D.jpg



Screenshot #15:

Alternatively, assigning the minimum and maximum values as new column values allows the creation of a new table. It looks advantageous when dealing with extensive datasets.

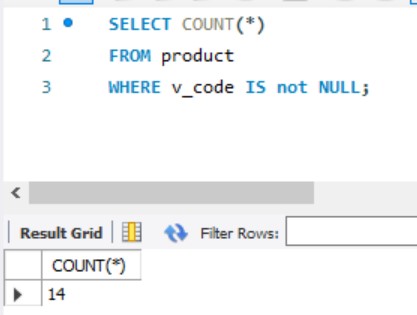
Filename: 15\_Scenario\_8A.jpg



Screenshot #16:

COUNT reveals the quantity of specific values. In this example, it counts the number of products where the V\_code is present.

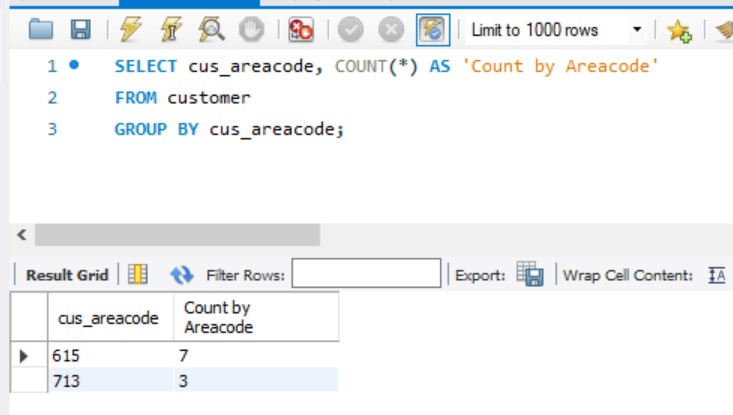
Filename: 16\_Scenario\_8B.jpg



Screenshot #17:

Using GROUP BY, you can count the occurrences of each value and present the counts for each group in individual rows.

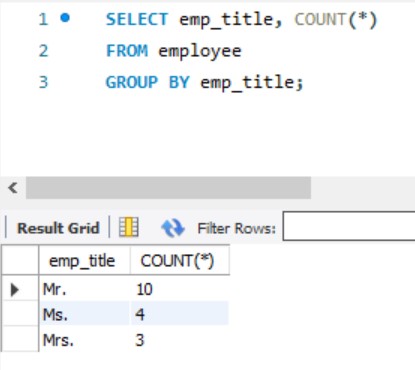
Filename: 17\_Scenario\_9A.jpg



Screenshot #18:

Alternatively, counting based on titles is also possible. This can be particularly useful when dealing with a table that contains repeated titles.

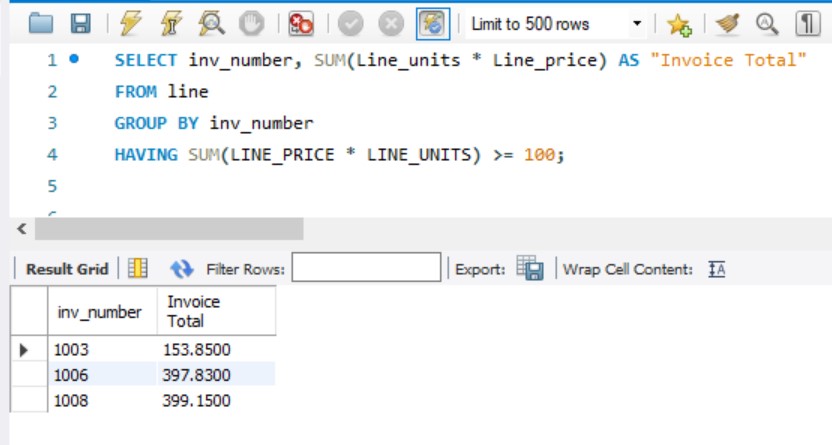
Filename: 18\_Scenario\_9B.jpg



Screenshot #19:

In the final screenshot, the total price for each invoice was calculated, automatically identifying values exceeding $100. This proved to be a much more concise and visually appealing method compared to calculating in Excel.

Filename: 19\_Scenario\_10.jpg



Screenshot #20:

This txt includes my Render site URL.



Filename: 20\_Render\_Web\_Users\_Site.txt

Screenshot #21:

I have modified all of my code to cater to web users transitioning from a to-do list

Filename: 21\_Qoddi\_Web\_Users\_Code.zip

Screenshot #22:

After changed to web users, I created a table in MySQL. This allows me to automatically update my contents and send them to the Render server.

Filename: 22\_Qoddi\_Web\_Users\_Screenshot.jpg