**CCGC 5004 Database Systems**

**Lab Exercise 3 SELECT Statement on Multiple Tables**

**Overview**

**To receive credit for this lab you must be present in today’s class. Late submissions are deducted 10% per day up to 5 days. Submissions received after 5 days will be given a grade of 0.**

**SELF\_JOIN Sample**

**I showed you an additional table in class called EMPLOYEES. The script to create this table is loaded in BlackBoard. Download this script and run the script in MySQL Workbench.**

**Write the query shown in the example for a self-join, execute and include the output in today’s lab.**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

**Do it yourself portion:**

**In this exercise you will use the schema below to execute the queries listed.**

Graphical user interface, diagram

Description automatically generated

**You will write and execute queries as requested in the questions below.**

1. Write a SELECT statement that joins the Categories table to the Products table and returns these columns: category\_name, product\_name, list\_price.

Sort the result set by the category\_name column and then by the product\_name column in ascending sequence. (10 rows returned)

Screen Capture 1

1. Write a SELECT statement that joins the Customers table to the Addresses table and returns these columns: first\_name, last\_name, line1, city, state, zip\_code.

Return one row for each address for the customer with an email address of [allan.sherwood@yahoo.com](mailto:allan.sherwood@yahoo.com). (2 rows returned)

Screen Capture 2

1. Write a SELECT statement that joins the Customers table to the Addresses table and returns these columns: first\_name, last\_name, line1, city, state, zip\_code.

Return one row for each customer, but only return addresses that are the shipping address for a customer. (8 rows returned)

Screen Capture 3

1. Write a SELECT statement that joins the Customers, Orders, Order\_Items, and Products tables. This statement should return these columns: last\_name, first\_name, order\_date, product\_name, item\_price, discount\_amount, and quantity.

Use aliases for the tables.

Sort the final result set by the last\_name, order\_date, and product\_name columns. (12 rows returned)

Screen Capture 4

1. Write a SELECT statement that returns the product\_name and list\_price columns from the Products table.

Return one row for each product that has the same list price as another product.   
*Hint: Use a self-join to check that the product\_id columns aren’t equal but the list\_price columns are equal.*

Sort the result set by the product\_name column. (2 rows returned)

Screen Capture 5

1. Write a SELECT statement that returns these two columns:

category\_name The category\_name column from the Categories table

product\_id The product\_id column from the Products table

Return one row for each category that has never been used. *Hint: Use an outer join and only return rows where the product\_id column contains a null value. (1 row returned)*

Screen Capture 6