Unit 9: Databases & The Modern Web:

Lab 1 – Northwind (SQL)

Assigned: 02/23/23

Due: N/A

Completed: 02/27/23

**LMS:** <https://lms.grandcircus.co/mod/assign/view.php?id=22855>

**Google Doc:** <https://docs.google.com/document/d/1gy1xtcWNmyVz5IAQq8HwImvRx6qkYgOwitfv4-NTW2g/preview>

**GitHub:** N/A

**Task:** Practice writing SQL statements on the Northwind database.

**Setup:**

* In pgAdmin, create a database called northwind.
* Open a SQL window. Copy-paste and run this file... <https://raw.githubusercontent.com/pthom/northwind_psql/master/northwind.sql>

**Build Specifications:**

Write SQL queries to do the following tasks. Record these queries in a text document and submit the document. You do not need to record the results of the queries.

1. Select all the records from the customers table.
   * SELECT \* FROM customers
2. Get distinct countries from the customers table.
   * SELECT DISTINCT country FROM customers
3. Get all the records from the table customers where the customer\_id starts with “BL”.
   * SELECT \* FROM customers WHERE customer\_id LIKE ‘BL%’
     1. The % acts as a wildcard, so it is important where you put this. Setup this way, the result starts with ‘BL’. If you wanted it to end with, then use ‘%BL’. If you want this to appear anywhere within, you would use ‘%BL%’.
4. Get the first 100 records of the orders table.
   * SELECT \* FROM orders LIMIT 100
5. Get all customers that live in the postal codes 1010, 3012, 12209, and 05023.
   * SELECT \* FROM customers WHERE postal\_code IN (‘1010’, ‘3012’, 12209’, ‘05023’)
6. Get all orders where the ShipRegion is not equal to NULL.
   * SELECT \* FROM orders WHERE ship\_region IS NOT NULL
7. Get all customers ordered by the country, then by the city.
   * SELECT \* FROM customers ORDER BY country, city
8. Add a new customer to the customers table. You can use whatever values
   * INSERT INTO customers(customer\_id, company\_name, contact\_name, etc.)

VALUES (‘1’, etc. )

1. Update all ship\_region to the value 'EuroZone' in the orders table, where the ship\_country is 'France'.
   * UPDATE orders SET ship\_region = 'EuroZone' WHERE ship\_country = 'France';
2. Delete all rows from order\_details that have a quantity of 1.
   * DELETE FROM order\_details WHERE quantity=’1’
3. Calculate the average, max, and min of the quantity in the order\_details table.
   * SELECT AVG(quantity) FROM order\_details
   * SELECT MIN(quantity) FROM order\_details
   * SELECT MAX(quantity) FROM order\_details
4. Calculate the average, max, and min of the quantity in the order\_details table, grouped by the order\_id.
   * SELECT AVG(quantity) FROM order\_details GROUP BY order\_id
   * SELECT MIN(quantity) FROM order\_details GROUP BY order\_id
   * SELECT MAX(quantity) FROM order\_details GROUP BY order\_id
5. Find the customer\_id that placed order 10290 (orders table)
   * SELECT customer\_id FROM orders WHERE order\_id= ‘10290’
6. Do an inner join, left join, right join on orders and customers tables.  (These are three separate queries, one for each type of join.)
   * SELECT \* FROM orders INNER JOIN customers ON orders.customer\_id = customers.customer\_id
   * SELECT \* FROM orders LEFT JOIN customers ON orders.customer\_id = customers.customer\_id
   * SELECT \* FROM orders RIGHT JOIN customers ON orders.customer\_id = customers.customer\_id
7. Use a join to get the ship\_city and ship\_country of all the orders which are associated with an employee who is in London.
   * SELECT ship\_city, ship\_country, FROM orders INNER JOIN employees ON employee\_id = employee\_id WHERE city = ‘London’
8. Use a join to get the ship\_name of all orders that include a discontinued product. (See orders, order\_details, and products. 1 means discontinued.)
   * SELECT DISTINCT(orders.ship\_name) FROM orders JOIN order\_details ON orders.orders\_id = order\_details.order\_id JOIN products ON order\_details.product\_id = products.product\_id WHERE products.discontinued = 1
9. Get first names of all employees who report to no one.
   * SELECT first\_name FROM employees WHERE reports\_to IS null
10. Get first names of all employees who report to 'Andrew'.
    * SELECT first\_name FROM employees WHERE reports\_to = (SELECT employee\_id FROM employees WHERE first\_name = ‘Andrew’)