# CS338: Assignment 3

## Notes:

Assignments are essential in learning the material and preparing for exams. You can ask for help but make sure you write the answers in your own words. We will check for copies. You should submit your work to crowdmark by **11:59 PM on July 22nd**. Late submissions will NOT be accepted. **Only use SQL commands that are discussed in lectures to answer these questions. Questions answered with SQL commands outside of lecture material will be MARKED ZERO.** One of the main objectives of this course (and the only objective of this assignment) is to help students learn and master the fundamentals of SQL. **You can use VIEW as an intermediate step.**

## Office Hours:

You can find **Mattie** in **DC2555** on **Tuesdays** and **Thursdays**, **from 16:00 to 17:30**.

## Submission

The submission deadline is **July 22nd, 11:59 PM** on **Learn Drop box**. To submit your answers, type them in a file named **Assignment3.md**. Make it easy to find your answers for each question. In your submitted file, use the following pattern on a separate line before the answer to each question, replacing [N] with the question number.

```
# Q.[N].
```

For example, to separate the answer to question 4, you would have to use the following line before your answer with empty lines before and after it:

```
# Q.4.
```

## Marking

Your queries will be tested against the provided sample dataset in the order of the questions. Full marks are only given to a correct output. If the output is incorrect, the TAs will decide how much of the correct answer has been submitted for the question.

## Database Description

In this exercise, we intend to become familiar with interacting with a relational database through the SQL language. To this end, we will use the Northwind database. The Northwind database is a sample created by Microsoft for their database tutorials. It holds sales data for a fictional company, "Northwind Traders," which deals in specialty foods globally. This database is a great learning tool for small-business systems, featuring customers, orders, inventory, purchasing, suppliers, shipping, employees, and accounting. The schema of this database can be found in this GitHub repository. For this assignment, we only consider a

subset of tables. These tables are listed below:

- `Suppliers`: Suppliers and vendors of Northwind.
- `Customers`: Customers who buy products from Northwind.
- `Employees`: Employee details of Northwind traders.
- `Products`: Product information.
- `Shippers`: The details of the shippers who ship the products from the traders to the end customers.
- `Orders` and `Order_Details`: Sales Order transactions taking place between the customers & the company.

A sample database is also provided (`northwind.db`). You can simply use this website if you do not want to install a DBMS. This online tool is quite easy to use and you can find tutorials online, such as this one. To load the provided database, go to the `File > Open DB` and select the `northwind.db` file to upload. Your answers will be run against this dataset in order to verify the correctness of your queries.

# Questions

1. Write SQL queries to delete the `demo`, `CustomerCustomerDemo` and `CustomerDemographics` tables. Make sure the query does not throw an error if the tables do not exist in the database.

2. Write an SQL command that creates a table named `ArchivedOrders`. Make sure the table does not already exist before creating the table. Note:
   - An `ArchivedDate` column automatically stores the date and time an order was inserted into the `ArchivedOrders` table.
   - The table is the result of natural join between two tables and include all the columns from them as follows:
     - all the data from the table `Orders` table with the same column names and types.
     - all the data from the table `OrderDetails` table with the same column names and types.
   - The PK in the table is the combination of "`OrderID`" and "`ProductID`" columns
   - Note that `OrderID` is not FK in this table. Also, note that there is only one column named `OrderID`. (This means that you should join the `Orders` and `OrderDetails` tables on the `OrderID` column when inserting an order into this table in Question 3.)
   - The constraints on the columns in this table are:
     - `Freight` has a default value of 0
     - `UnitPrice` has a default value of 0 and its value is always greater than or equal to 0
     - `Quantity` has a default value of 1 and its value is always greater than 0 (not equal)
     - `Discount` has a default value of 0 and its value is between 0 and 1(inclusive)
     - `ProductID` is a foreign key that references `Products.ProductID`
     - `EmployeeID` is a foreign key that references `Employees.EmployeeID`
     - `CustomerID` is a foreign key that references `Customers.CustomerID`
     - `ShipVia` is a foreign key that references `Shippers.ShipperID`
     - The foreign keys do nothing upon deletion and update

3. Write SQL queries that move the information for any order before the date `2016-08-01` (exclusive) from the `Orders` and `OrderDetails` tables to the `ArchivedOrders` table. Your queries must:
    - Ensure that after execution, none of the orders in the `ArchivedOrders` table exist in the `Orders` and `OrderDetails` tables.
    - After the action is complete, write SQL queries that print the number of records in the `Orders`, `OrderDetails`, and `ArchivedOrders` tables in that order.

4. Write SQL query to get the number of products the supplier "`Ma Maison`" sells. The output is an integer.

5. Write SQL query to get the average number of products per Category. The output is a number.

6. Write a SQL query to get all the information on the ID, description, and the number of employees in the region (in the same order) with the largest number of employees. The output is a table with a single record. You can use Views in this question.

7. Write an SQL query to get the number of customers who have made more than 10 orders. The output is an integer.

8. Write a SQL query to get the company name, city, and country of the customers who received their orders through the "United Package" shipper.

9. Write a SQL query that shows the ID of ALL employees, the ID of their manager, and the numbers of territories in `EmployeeTerritories` that their manager is responsible for. A manager is an employee to whom at least one other employee reports.

10. Add the Waterloo Shipping company using the below Query, and then write a select statement that shows the name of ALL shipper companies, the number of orders they delivered, and the average of the total cost of these orders, such that all orders have more than 5 products on them. Please note that the cost of each product in an order is (`UnitPrice * Quantity - Discount`) and the total cost of an order is the sum of the above values calculated for all products in the order.

```
INSERT INTO Shippers (ShipperID, CompanyName, Phone)
       VALUES (4, 'Waterloo Shipping', '11111111');
```