**THE DATABASE YOU SHOULD BE USING, IS LISTED IN THE HOMEWORK 3 ASSIGNMENT DETAILS.**

**Basic SQL Instructions**

You will need to upload your script containing all of your SELECT queries. This can be a .txt, .sql, or .docx document.

Be sure to number each query in your script with two dashes in front of the number to comment it out (e.g. - -1)

Do not leave your queries commented out! Each script should run all 11 queries, in order, without having any of the queries commented out. You can have notes or the questions for each query commented out in the script. Failure to have all queries uncommented, will result in points lost.

1. Find customers (customer id) who have not placed any orders:

2. List the names and number of employees supervised (label this value HeadCount) for all the supervisors who supervise more than two employees:

3. Names of states where customers reside, but have no employees residing in that state: **Use the MINUS operator to solve this query:**

4. Display the customer ID, name, and order ID for all customer orders. For those customers who do not have any orders, include them in the display once, with a 0 value for OrderID. Hint: you will need to use UNION here.

5. Show the customer ID and name for all the customers who have ordered both products with IDs 5 and 4 on the same order:

6. List the IDs and names of all products that cost less than the average product price in their product line.

7. List the IDs and names of those sales territories that have at least 50 percent more customers as the average number of customers per territory.

8. List the order number and order quantity for all customer orders for which the order quantity is greater than the average order quantity of that product: (Hint: This involves a correlated subquery.)

9. For each product display in ascending order by product ID the product ID and description along with the customer ID and name for the customer who has bought the most of that product; also show the total quantity ordered by that customer (who has bought the most of that product). Use a correlated subquery:

**Result (a few rows to show you what it should look like) :**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |

*10. Write a query to get the product(s) whose total order is the lowest in the organization. Base the total order on revenue generated (standard price \* order quantity). Outout the product id and the revenue for this product.*

*11. Find the bottom 3 customers who generated the lowest amount of revenue. For these customers, output the customer name and customer state. There is a tie for 3rd place, so your query needs to be dynamic enough to return all 4 customers who match the criteria, without hardcoding anything like “4” in your query.*