**FOR THE SQL PROBLEMS IN TASK THREE, CREATE A SEPARATE FLAT TEXT FILE THAT ENDS IN .sql.**

In this exercise, you will further familiarize yourself with basic SQL commands. In addition to the slides and examples provided, you are expected to use the tutorial at <https://www.w3schools.com/sql/default.asp> . You are encouraged to work with a partner. There are three tasks to accomplish:

**Task 1:** Complete the introduction to database lab tutorial on Moodle. Do this as an individual.

**Task 2:**

Read and practice the examples of the tutorial chapters in the following order (these are short tutorials and should not take long to accomplish). Also, take time to look at Chapter 17 in the Murach text. You do not need to submit anything for this task…it is to help you with task 3.

|  |  |
| --- | --- |
| Tutorial Topic | Link |
| Home | <https://www.w3schools.com/sql/default.asp> |
| Basic Introduction | <https://www.w3schools.com/sql/sql_intro.asp> |
| Basic Syntax | <https://www.w3schools.com/sql/sql_syntax.asp> |
| SQL Select | <https://www.w3schools.com/sql/sql_select.asp> |
| SQL Distinct | <https://www.w3schools.com/sql/sql_distinct.asp> |
| SQL Where | <https://www.w3schools.com/sql/sql_where.asp> |
| SQL And, Or, Not | <https://www.w3schools.com/sql/sql_and_or.asp> |
| SQL JOINS | <https://www.w3schools.com/sql/sql_join.asp> |
| SQL INNER JOIN | <https://www.w3schools.com/sql/sql_join_inner.asp> |
| SQL UNION | https://www.w3schools.com/sql/sql\_union.asp |

**Task 3:**

Now, try to write the following queries using the pubs database. Save each query you write into a flat text file and submit it to Moodle. Here are some specific considerations:

* Unless stated otherwise, each requirement must be done within a single SELECT query.
* In the flat text file, make sure each query has a comment above to indicate the problem number and the query should be formatted as follows (Note how the each portion is started on a new line):

--9

SELECT col1

,col 2

,col 3

FROM table1

WHERE condition1

AND/OR condition2

**Task 3 Deliverable:** Place all your queries into a single script. All the queries should be listed in the order of the questions asked. Make sure you test it to ensure it runs flawlessly without any form of pre-setup commands except for starting SQL Server Management Studio and logging in. Also, make sure you review the data and verify the expected results.

**Questions/ Problems:**

1. Write the command to change the database context to pubs.
   1. USE Pubs
2. Write a query that will list all information about the authors who live in California.
   1. select \* from authors where state = 'CA';
3. Write a query that will list the ID, title and the price of each title.
   1. select title\_id, title, price from titles;
4. Modify the previous query to add another column that will reflect what the price would be if increased by ten percent. The extra column should be titled “Adjusted Price”.
   1. alter table dbo.titles add adjusted\_price as (price + (price \* .10));
5. Now find out the title with the highest price (research the use of the MAX function).
   1. select max(titles.price) from titles;
6. Sales are important. Write a query that will list each title and all the information about each sale in the sales table.
   1. select title, sales.\* from titles, sales;
7. As a comment in your script, note what type of query the previous problem required. Was it a natural join, an inner join, a theta join or an outer join. Explain your answer.
   1. From what I can tell it’s a right outer join because I’m retuning matched records from the left table (titles) with the right and returning all records from the right (sales)
8. Revise that previous query to include the actual name of the store from the stores table and to restrict the information to that, the title, order dates and quantity ordered. The Headers should also not be the cryptic name from the table definition.

select title as Title, stor\_name as Store\_Name, ord\_date as Order\_Date,

qty as Quantity

from titles, stores, sales;