How to do homework assignments.

CS197Q Intro to SQL

Homework in 197Q

- The homework assignments are based on a segment of the material covered in the text.
- Each homework assignment is an .sql file with a number of problems for which you will write SQL code.
- Each assignment references a database. That database will be listed in the assignment.
- If it is a new database, you will create and populate it using SQLite before doing the homework assignment.
- You will be able to execute and test your code before you submit it.

Important Points!

- Use the file provided to submit all of your sql code. It contains a set of questions in comments.
- **Do not modify** the file except to enter your sql code after each question. Include **only the SQL answer** to the question and **no other SQL.**
- **Do not put any additional** comments in the file. Contact an instructor with any comments or questions.
- Upload that file with your sql code.
- Be sure you upload a file with a .sql extension.
- Do not upload a file that contains non-standard characters or other markup text such as html. Use UTF-8 encoding.

Organization

I suggest creating something like this directory (folder) structure on your machine:

```
 … documents/CS197Q/
                      sqlite_database_scripts
                      sqlite_databases
                      homework
```

The first folder contains the database (.db) files SQLite/DB Browser can load.

The last folder contains the homework assignments (.sql files).

Preparatory Steps

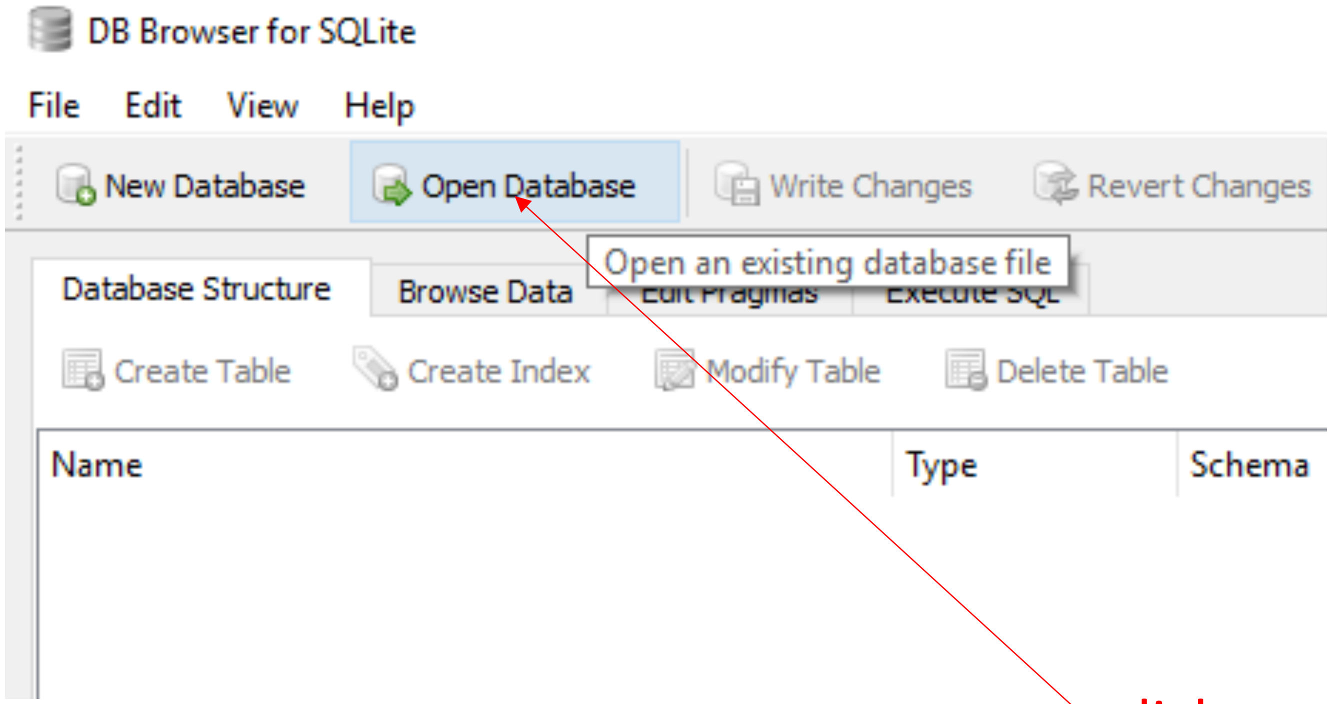
1. Download the assignment file from Moodle. The files will be named CS197QHwkX.X.sql.
2. Place the homework file in a well-named directory (folder) such as “homework”.
3. The homework will reference a database. The name of the database to use will be in the file and on the Moodle assignment.
4. The homework will use either an existing or a new database.
5. If necessary, create and populate the new database. The script files will be available on Moodle. Refer to the SQLite install guide for that procedure.

Steps for Doing the Work

1. Launch the DB Browser for SQLite.
2. Select “open database”, navigate to the database file and open it. The .db files should be in a directory such as “sqlite_databases”. The database tables should appear when you open the .db file.
3. Select the “Execute SQL” tab.
4. Select the “Open SQL File” Icon. Navigate to the homework file and open it.
5. Use the SQL editor to write your queries. Save and execute them to test. Use the “DB Schema” pane to explore the tables and columns of the database for reference.

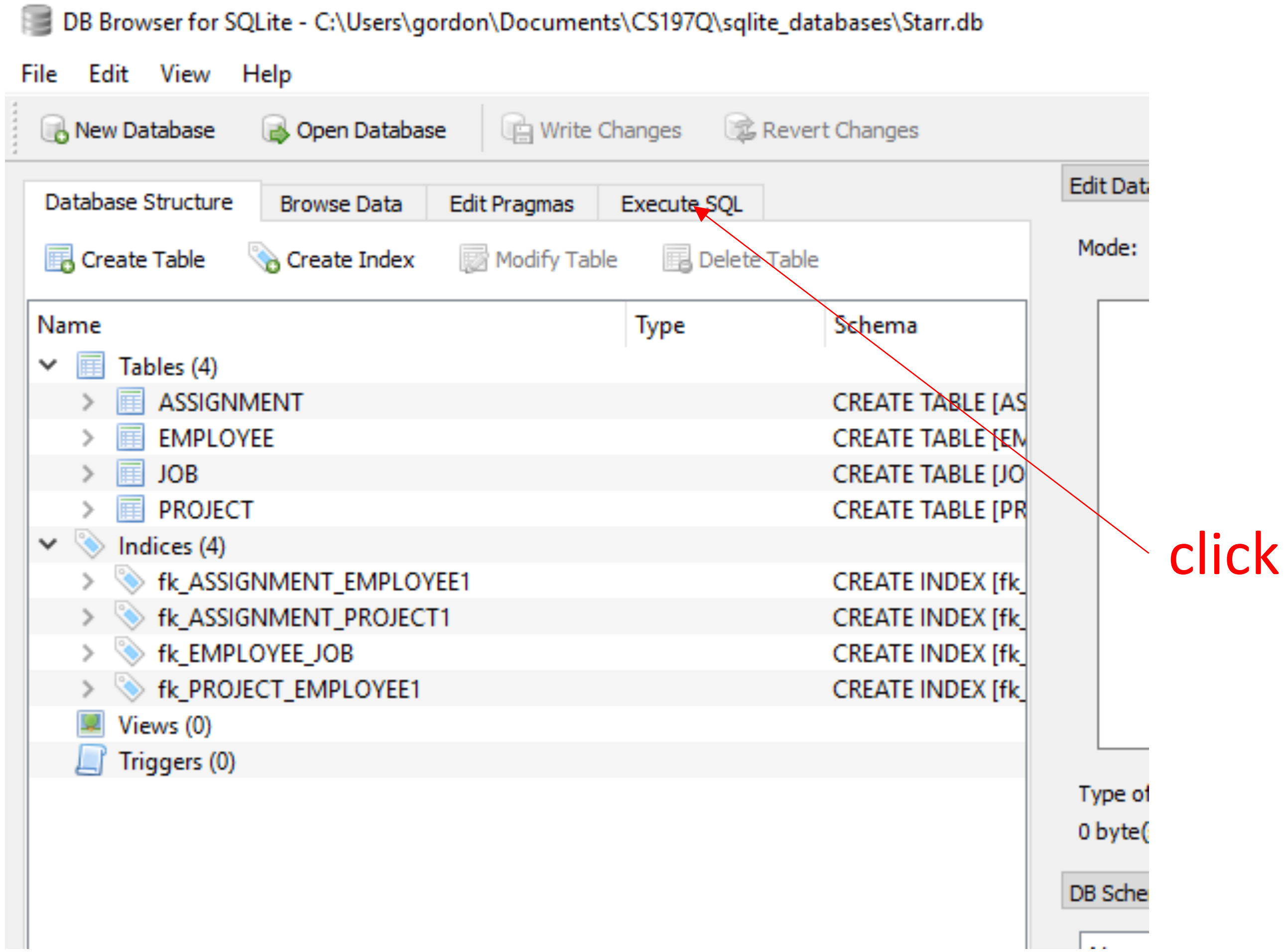
A Pictorial Guide using the DB Browser.

Select the Database



click

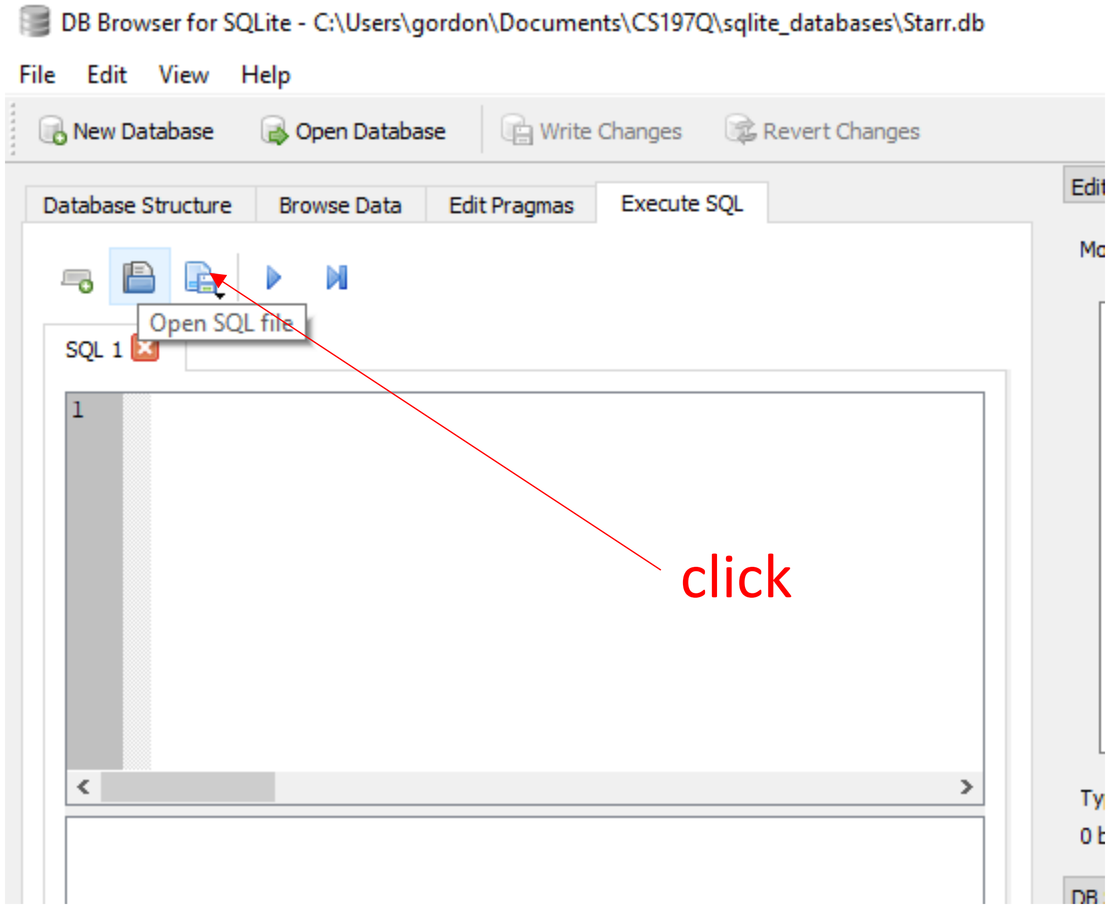
Database Opened, Select Execute SQL Tab



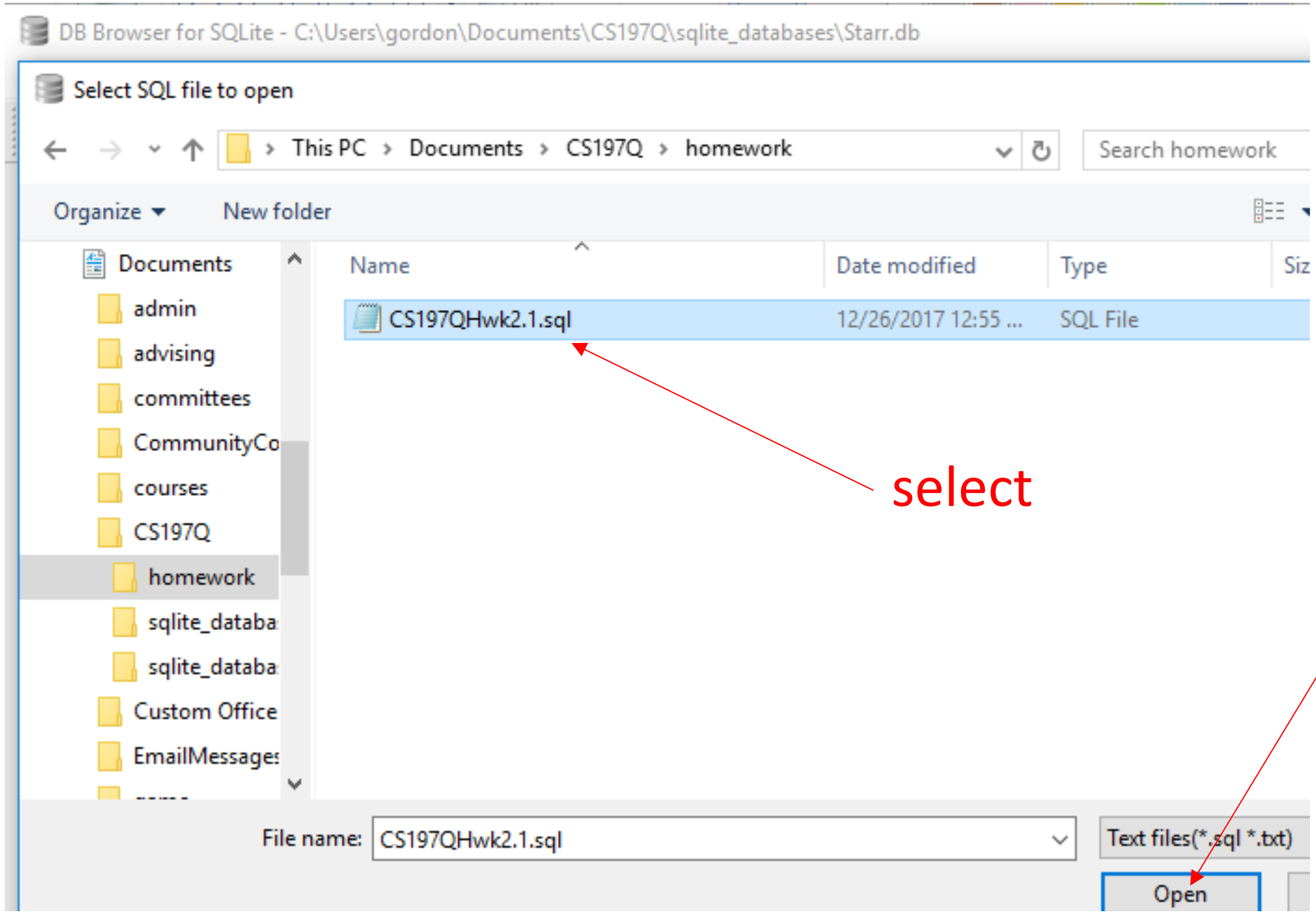
DB Browser for SQLite - C:\Users\gordon\Documents\CS197Q\sqlite_databases\Starr.db

Name	Type	Schema
Tables (4)		
ASSIGNMENT		CREATE TABLE [AS
EMPLOYEE		CREATE TABLE [EM
JOB		CREATE TABLE [JO
PROJECT		CREATE TABLE [PR
Indices (4)		
fk_ASSIGNMENT_EMPLOYEE1		CREATE INDEX [fk_
fk_ASSIGNMENT_PROJECT1		CREATE INDEX [fk_
fk_EMPLOYEE_JOB		CREATE INDEX [fk_
fk_PROJECT_EMPLOYEE1		CREATE INDEX [fk_
Views (0)		
Triggers (0)		

Open the Homework File



Navigate to File, Select and Open



Name	Date modified	Type
CS197QHwk2.1.sql	12/26/2017 12:55 ...	SQL File

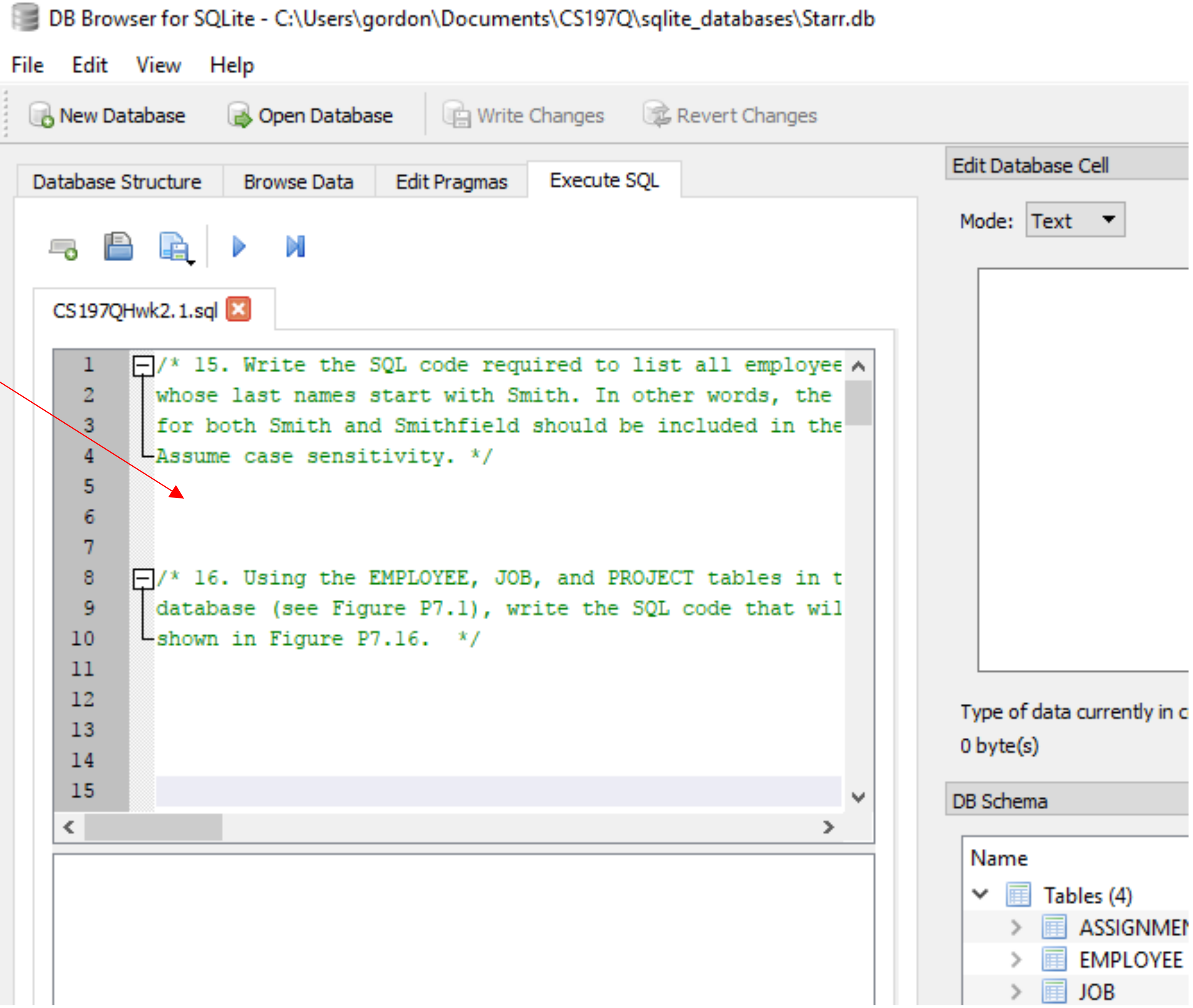
File name: CS197QHwk2.1.sql

select

click

SQL Editor

Enter your SQL code here



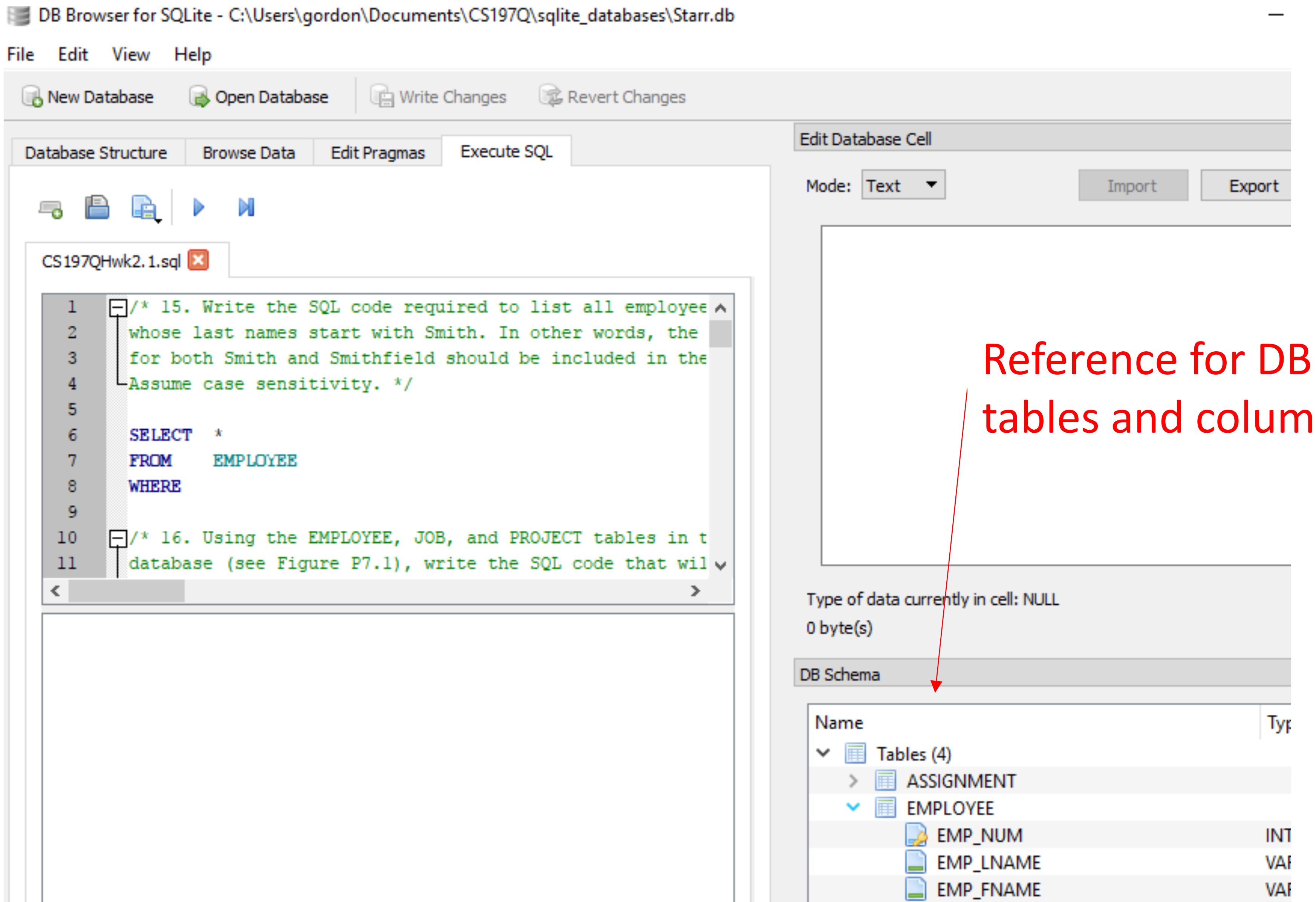
DB Browser for SQLite - C:\Users\gordon\Documents\CS197Q\sqlite_databases\Starr.db

Mode: Text

```
/* 15. Write the SQL code required to list all employee
whose last names start with Smith. In other words, the
for both Smith and Smithfield should be included in the
Assume case sensitivity. */



/* 16. Using the EMPLOYEE, JOB, and PROJECT tables in t
database (see Figure P7.1), write the SQL code that wil
shown in Figure P7.16.  */
```

Type of data currently in c
0 byte(s)

Query Writing

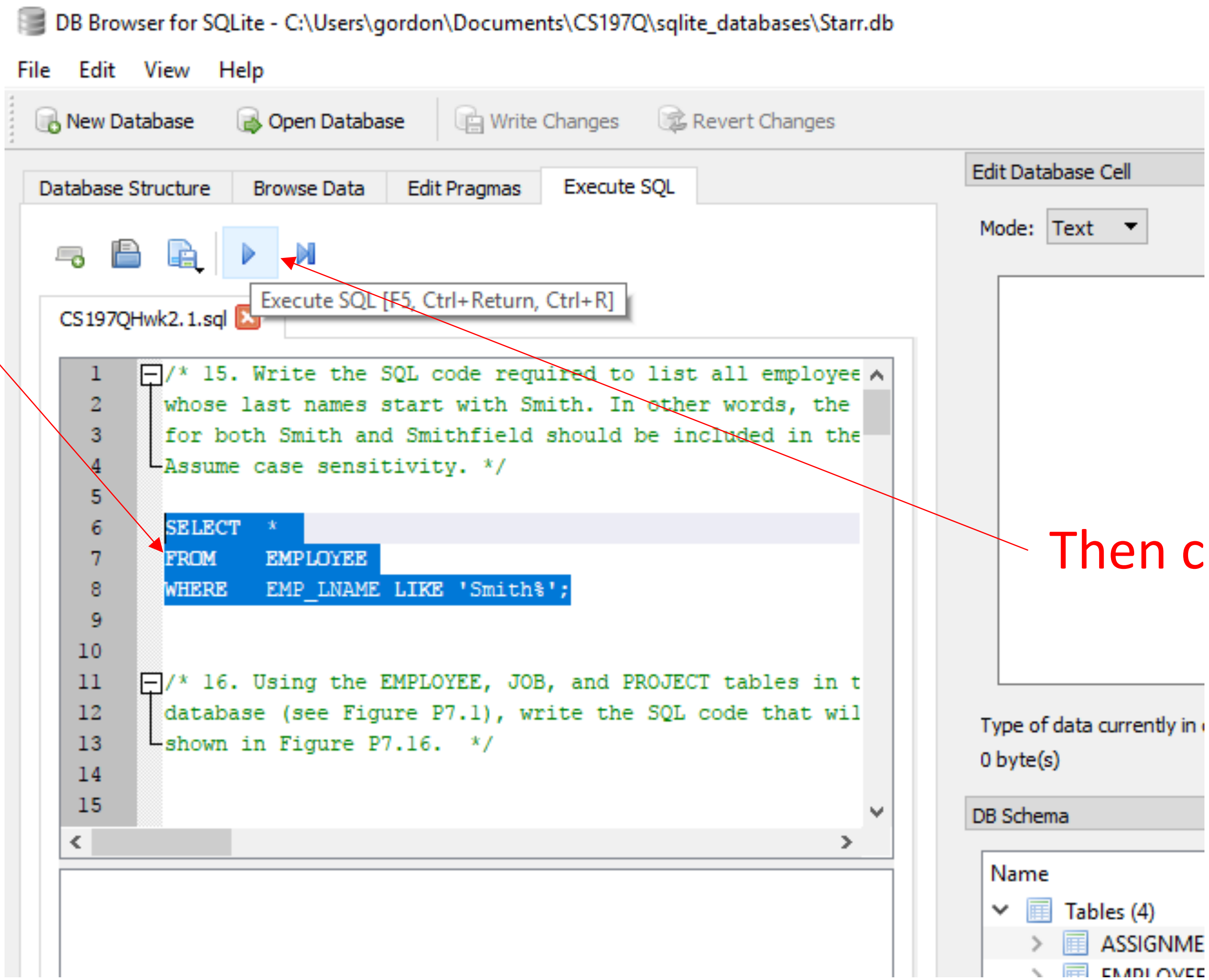


```
/* 15. Write the SQL code required to list all employee
whose last names start with Smith. In other words, the
for both Smith and Smithfield should be included in the
Assume case sensitivity. */

SELECT  *
FROM    EMPLOYEE
WHERE

/* 16. Using the EMPLOYEE, JOB, and PROJECT tables in t
database (see Figure P7.1), write the SQL code that wil
```

Reference for DB structure- tables and columns

Executing Your Query

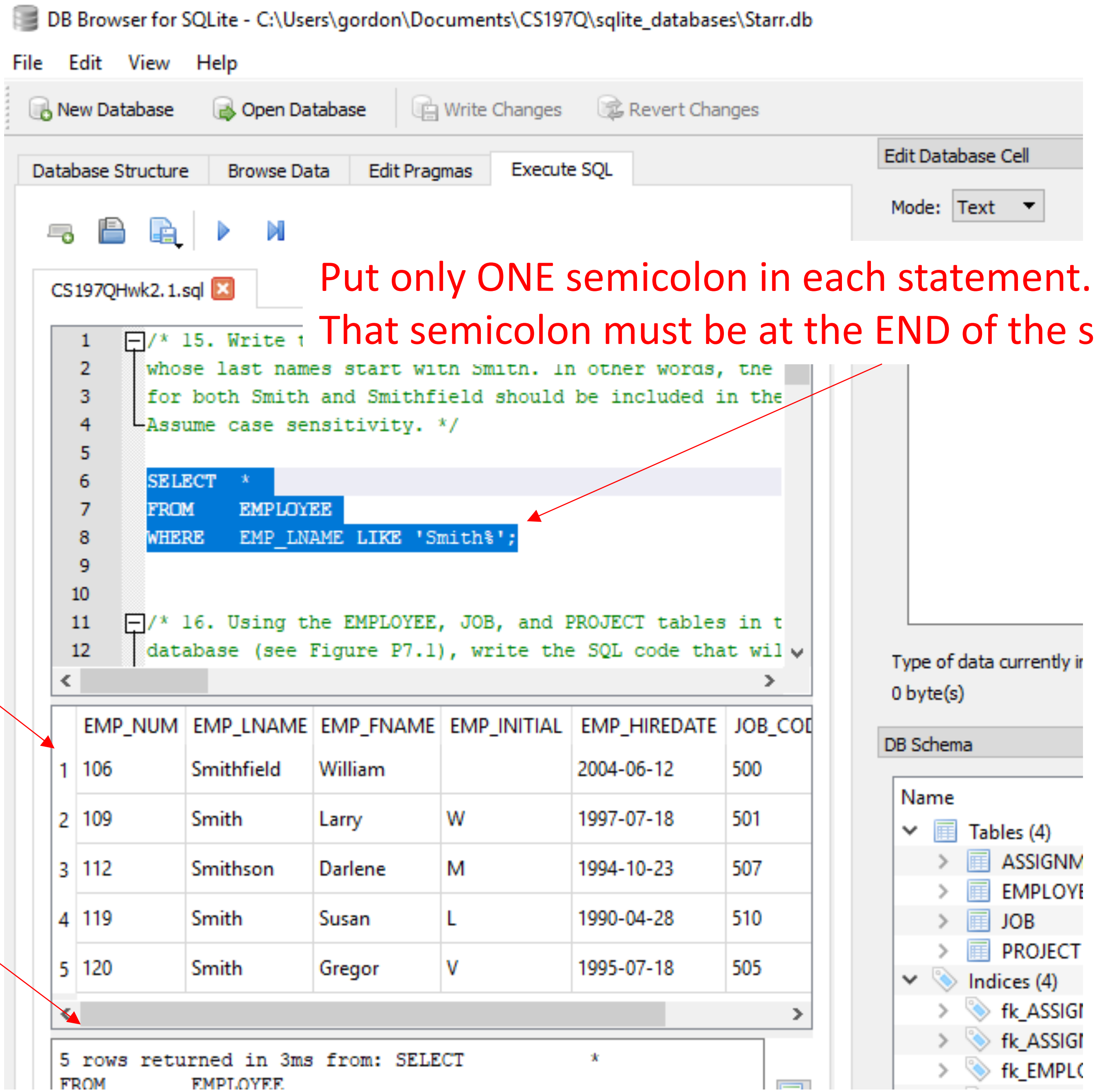
Select the code to be executed



```
/* 15. Write the SQL code required to list all employee
whose last names start with Smith. In other words, the
for both Smith and Smithfield should be included in the
Assume case sensitivity. */

SELECT   *
FROM     EMPLOYEE
WHERE    EMP_LNAME LIKE 'Smith%';


/* 16. Using the EMPLOYEE, JOB, and PROJECT tables in t
database (see Figure P7.1), write the SQL code that wil
shown in Figure P7.16.  */
```

Then click the blue arrow

Results



```
/* 15. Write t
whose last names start with Smith. In other words, the
for both Smith and Smithfield should be included in the
Assume case sensitivity. */

SELECT  *
FROM    EMPLOYEE
WHERE   EMP_LNAME LIKE 'Smith%';


/* 16. Using the EMPLOYEE, JOB, and PROJECT tables in t
database (see Figure P7.1), write the SQL code that wil
```

	EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_COD
1	106	Smithfield	William		2004-06-12	500
2	109	Smith	Larry	W	1997-07-18	501
3	112	Smithson	Darlene	M	1994-10-23	507
4	119	Smith	Susan	L	1990-04-28	510
5	120	Smith	Gregor	V	1995-07-18	505

```
5 rows returned in 3ms from: SELECT        *
FROM        EMPLOYEE
```

Put only ONE semicolon in each statement.
That semicolon must be at the END of the statement.

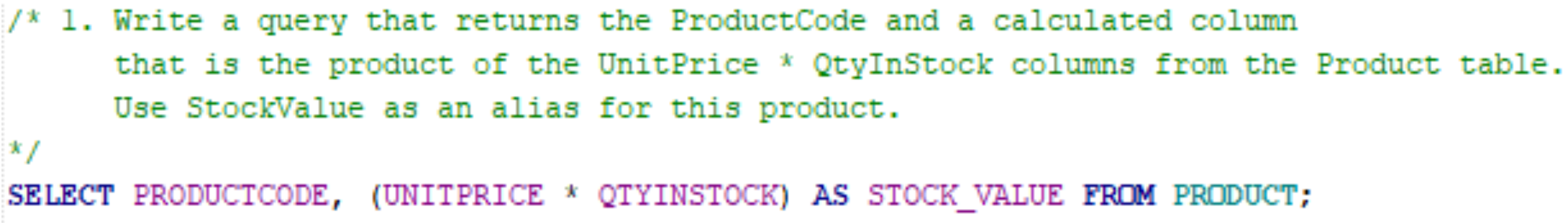
Result set

Message pane

Some “Do’s” and “Dont’s”

Please read the following examples.

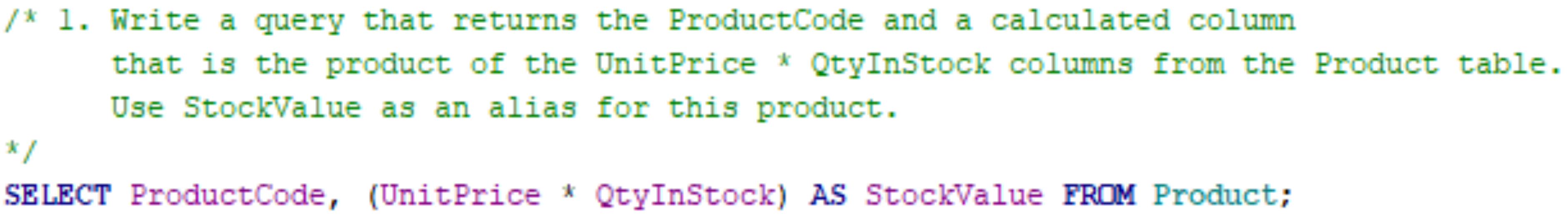
Do Not: (Naming Style)



```
/* 1. Write a query that returns the ProductCode and a calculated column
      that is the product of the UnitPrice * QtyInStock columns from the Product table.
      Use StockValue as an alias for this product.
*/
SELECT PRODUCTCODE, (UNITPRICE * QTYINSTOCK) AS STOCK_VALUE FROM PRODUCT;
```

Do not depart from the naming style of the database you are working with.

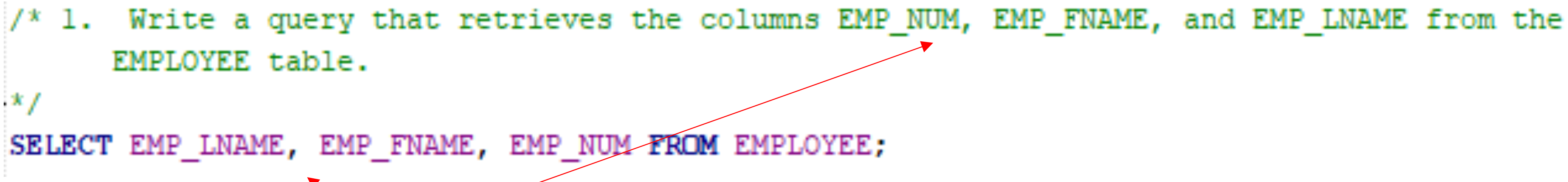
Do:



```
/* 1. Write a query that returns the ProductCode and a calculated column
      that is the product of the UnitPrice * QtyInStock columns from the Product table.
      Use StockValue as an alias for this product.
*/
SELECT ProductCode, (UnitPrice * QtyInStock) AS StockValue FROM Product;
```

Match the naming style of table and column names.

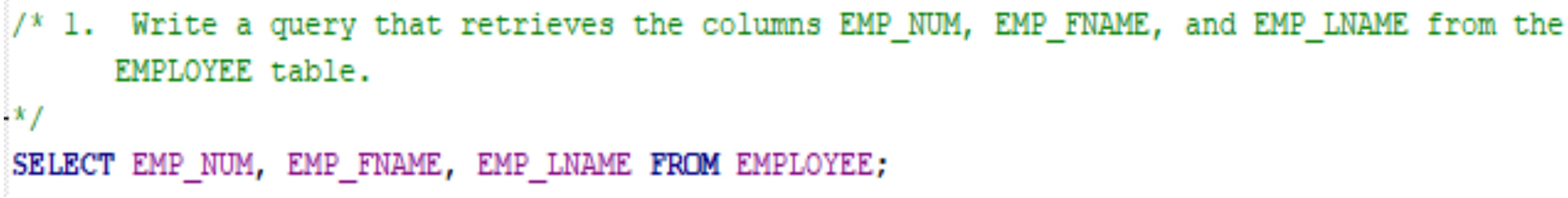
Do Not: (Column Order)



```
/* 1.  Write a query that retrieves the columns EMP_NUM, EMP_FNAME, and EMP_LNAME from the
      EMPLOYEE table.
*/
SELECT EMP_LNAME, EMP_FNAME, EMP_NUM FROM EMPLOYEE;
```

Do not use a different order for the columns in your query.

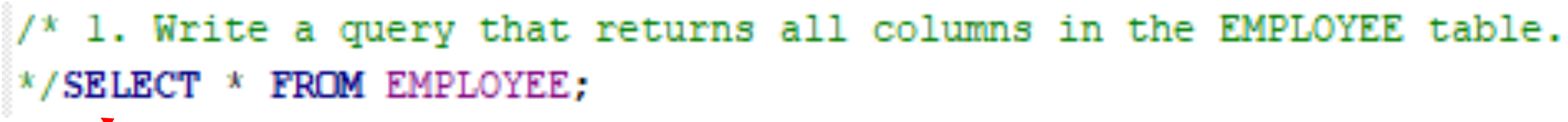
Do:



```
/* 1.  Write a query that retrieves the columns EMP_NUM, EMP_FNAME, and EMP_LNAME from the
      EMPLOYEE table.
*/
SELECT EMP_NUM, EMP_FNAME, EMP_LNAME FROM EMPLOYEE;
```

Match the order of columns.

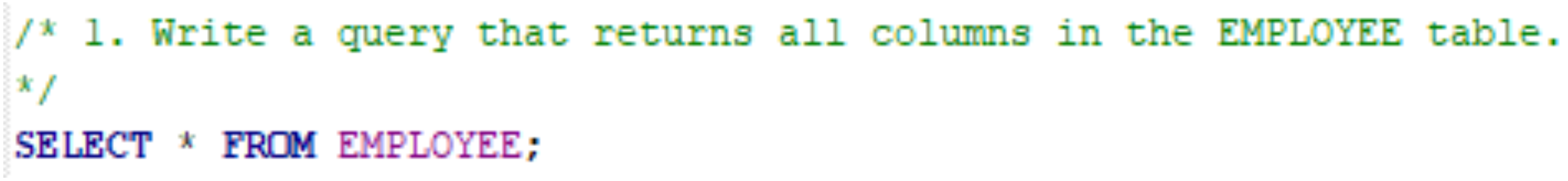
Do Not: (Comments and SQL)



```
/* 1. Write a query that returns all columns in the EMPLOYEE table.
*/SELECT * FROM EMPLOYEE;
```

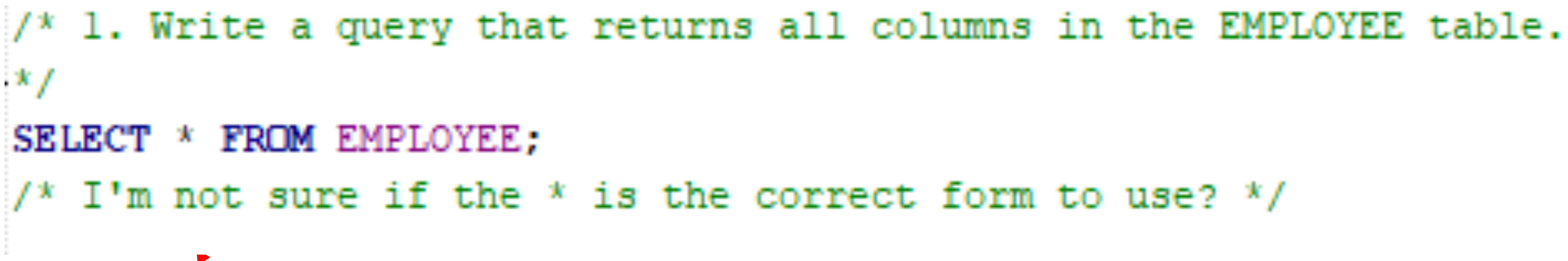
Do not place SQL on the same line as the closing comment.
This cannot be run as a script.

Do:



```
/* 1. Write a query that returns all columns in the EMPLOYEE table.
*/
SELECT * FROM EMPLOYEE;
```

Do Not: (Additional Comments)

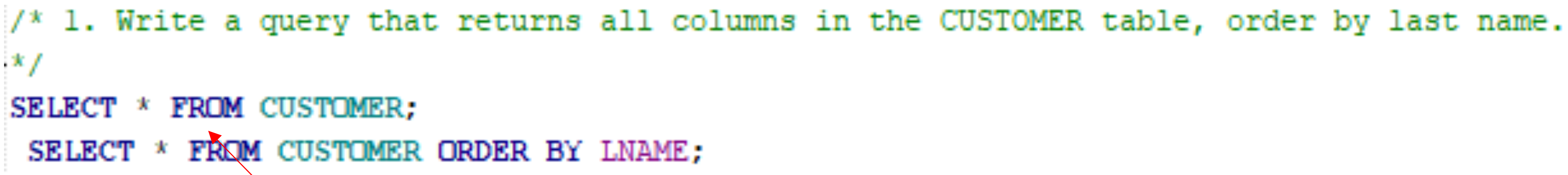


```
/* 1. Write a query that returns all columns in the EMPLOYEE table.
*/
SELECT * FROM EMPLOYEE;
/* I'm not sure if the * is the correct form to use? */
```

Do not enter any additional comments. If you have comments, post them to the course website/communications app.

NOTE: this pertains to the homework for this course. In general, additional comments are OK in script files as long as they are informative.

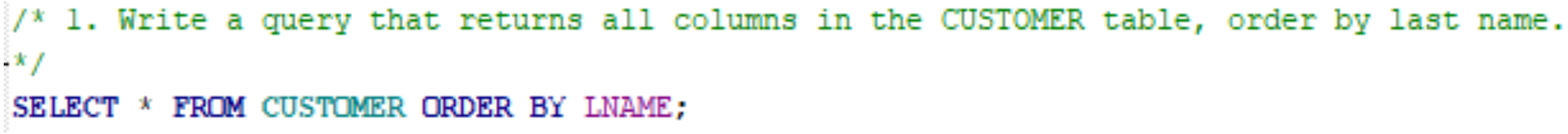
Do Not:



```
/* 1. Write a query that returns all columns in the CUSTOMER table, order by last name.
*/
SELECT * FROM CUSTOMER;
 SELECT * FROM CUSTOMER ORDER BY LNAME;
```

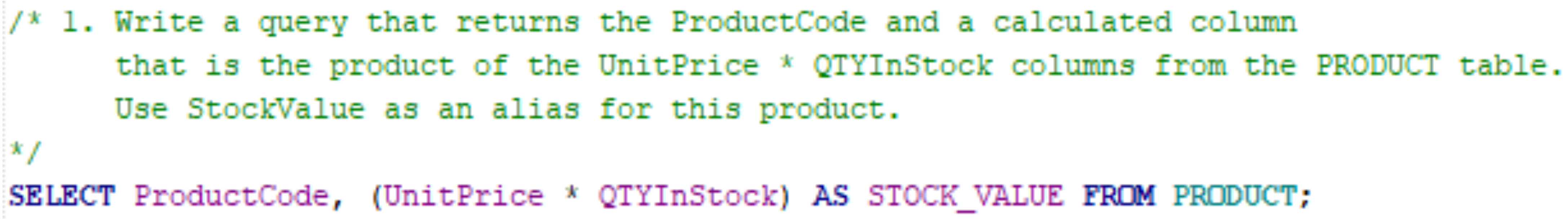
Do not enter any additional SQL. Enter the SQL that is your answer to the question only.

Do:



```
/* 1. Write a query that returns all columns in the CUSTOMER table, order by last name.
*/
SELECT * FROM CUSTOMER ORDER BY LNAME;
```

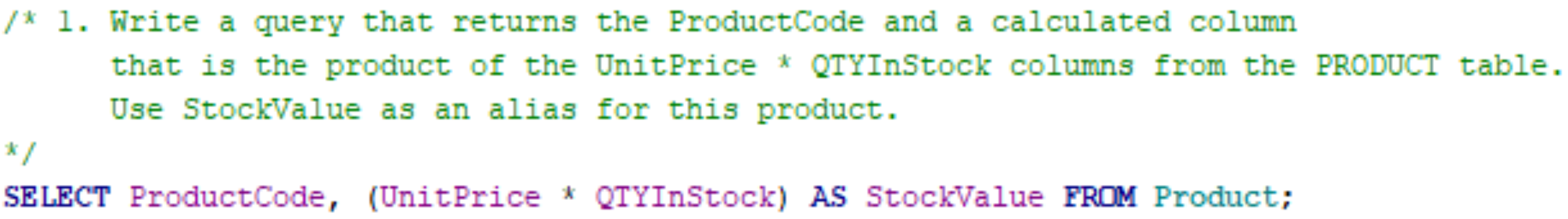
Do Not: (Alias Strings)



```
/* 1. Write a query that returns the ProductCode and a calculated column
      that is the product of the UnitPrice * QTYInStock columns from the PRODUCT table.
      Use StockValue as an alias for this product.
*/
SELECT ProductCode, (UnitPrice * QTYInStock) AS STOCK_VALUE FROM PRODUCT;
```

Do not use “random” alias strings.

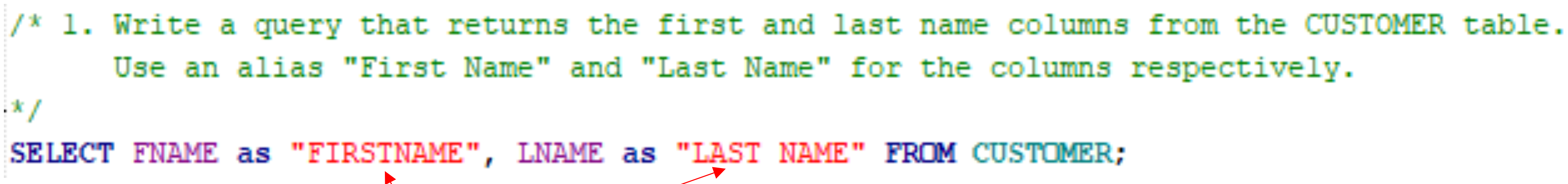
Do:



```
/* 1. Write a query that returns the ProductCode and a calculated column
      that is the product of the UnitPrice * QTYInStock columns from the PRODUCT table.
      Use StockValue as an alias for this product.
*/
SELECT ProductCode, (UnitPrice * QTYInStock) AS StockValue FROM Product;
```

Match the alias strings exactly.

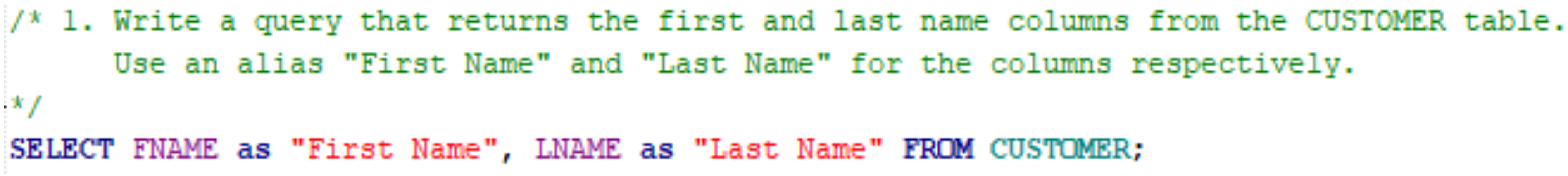
Do Not: (Alias Strings II)



```
/* 1. Write a query that returns the first and last name columns from the CUSTOMER table.
      Use an alias "First Name" and "Last Name" for the columns respectively.
*/
SELECT FNAME as "FIRSTNAME", LNAME as "LAST NAME" FROM CUSTOMER;
```

Do not use “random” alias strings.

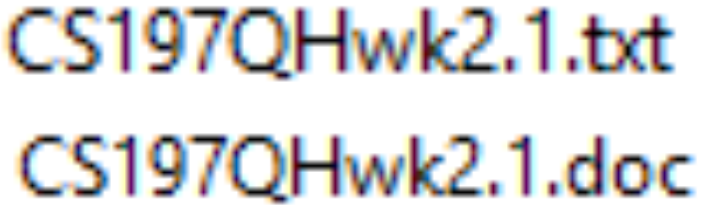
Do:



```
/* 1. Write a query that returns the first and last name columns from the CUSTOMER table.
      Use an alias "First Name" and "Last Name" for the columns respectively.
*/
SELECT FNAME as "First Name", LNAME as "Last Name" FROM CUSTOMER;
```

Match the alias strings exactly.

Do Not: (Homework File Extension)

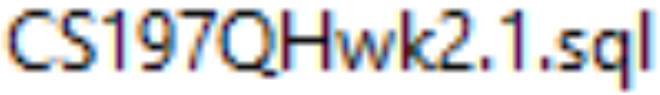


CS197QHwk2.1.txt

CS197QHwk2.1.doc

Do not upload files that do not have a .sql extension.

Do:



CS197QHwk2.1.sql

Upload a file with a .sql extension.

Finally…

- When you have finished the assignment, upload the .sql file to the Moodle assignment.

Homework Tips

- Write your query one small step at a time- execute, test, then go further. Don’t try to write it all at one go.
- Writing queries is an exploration of the data. You need to become “familiar” with the data and the database.
- Try to solve the problem before you ask.
- Try executing different approaches. (You can’t break it!!).
- Don’t forget to use the text as a reference.
- Use the web to search for ways of writing queries.
- Use the Moodle public forum as well if you get stuck.
- Please be sure to submit your own work.
- Make sure you understand the code you submit!

Remember:

- We are all here to learn.
- We are all at different levels of understanding.
- There are NO stupid questions.
- You will gain much if you put in the effort!