Lab 02 – SQL Single Table Queries

# **Objectives**

The purpose of this lab is to start learning SQL by writing basic SELECT and DML statements involving a single table.

**Submission**

***Your submission will be a single SQL file with the solutions provided (with a .sql file extension). Submission in any other file format will get a 0.***

* Save your SQL statements as a script named: DBS211\_L02\_FirstName\_LastName.sql
* Locate, select, and submit the file to the Lab 02 link.

Using comments to number the question answers, write the SQL code to complete the following tasks.

Example Submission

|  |
| --- |
| -- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  -- Name: Your Name  -- ID: #########  -- Date: The current date  -- Purpose: Lab 02 DBS211  -- \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  -- Q1 SOLUTION --  SELECT \* FROM TABLE;  -- Q2 Solution –  SELECT \* FROM TABLE; |

## **Setup**

Create a new worksheet in SQL developer and add an appropriate comment header that includes your name, student id, the date and the purpose of the file (i.e. DBS211 – Lab 02).

## **Style Guide**

Your SQL should be written using the standard coding style:

* all keywords are to be in upper case
* all user-defined names are to be in lower case (example: table and field names)
* there should be a carriage return before each major part of the SQL statement (i.e. before SELECT, FROM, WHERE and ORDER BY)

See the following sample:

SELECT column1,

column2,

column3

FROM table\_name

WHERE conditions

ORDER BY column1,

column2;

To save time, you can write all SQL statements in your SQL developer.

**Marking Scheme**

|  |  |
| --- | --- |
| **Total Questions** | **8** |
| **Mark on each Question** | **0.5** |
| **Total Marks** | **4** |

**Grade Policy**

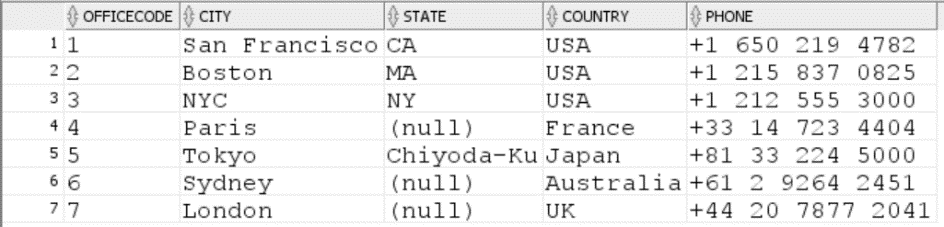
* Questions with errors do not get any marks (they get 0). Execute your *.sql* file using the “Run Script” button to make sure there are no errors in your file.
* You do not receive marks for the missing solutions.

**Tasks:**

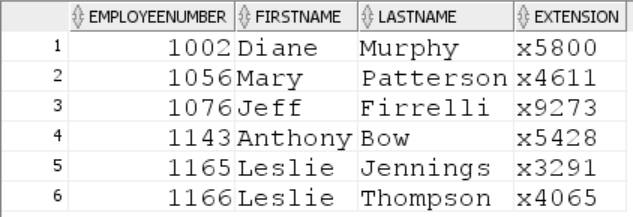
**SQL SELECT**

Remember to comment the question number for each solution.

1. Display the data for all offices. *Display office code, city, state, country and phone for all offices.*



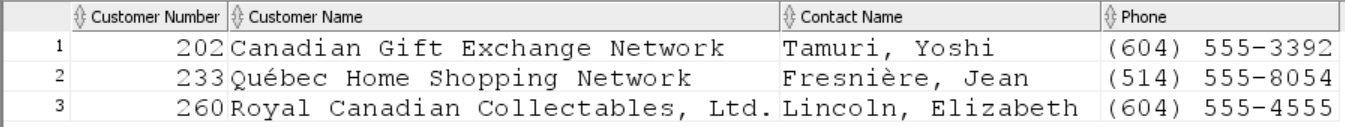
1. Display *employee number, first name, last name, and extension* for all employees whose office code is 1. *Sort the result based on the employee number.*



1. Display customer number, customer name, contact first name, contact last name and phone for all customers taking into account the below points:
   1. The contact’s first and last name should be in a single column in the below format.

*lastname, firstname*

* 1. Show customers who are in Canada
  2. The column names should look exactly like in the sample shown below
  3. *Sort the result based on the customer name.*

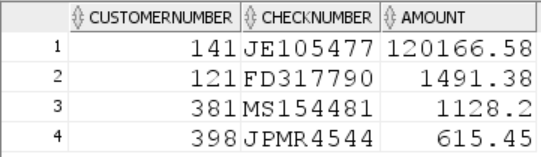


1. Display customer number for customers who have payments. Do not included any repeated   
   values. *Sort the result based on the customer number.* (**Hints**: How do you know a customer has made a payment? You will need to access only one table for this query)

*Only the first 10 rows of the output result is shown below but the query must return 98 rows.*



1. List customer numbers, check number, and amount for customers whose payment amount is not in the range of *$1,500* to *$120,000*. Sort the output by top payment amounts first.

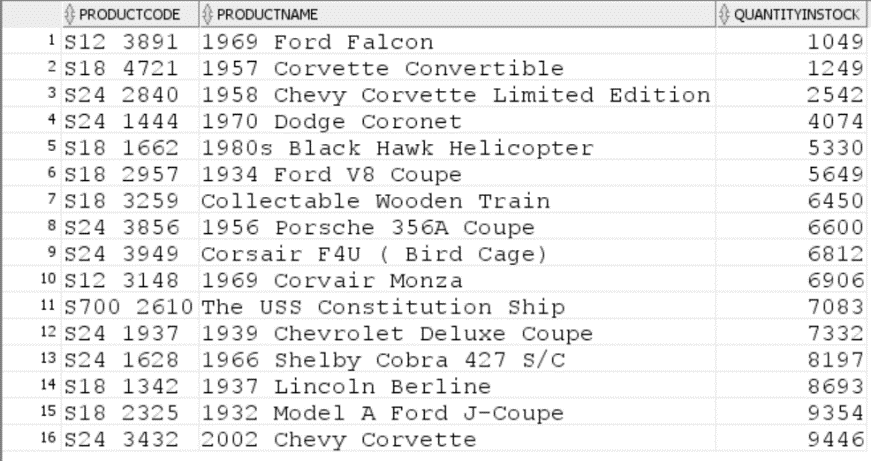


1. The company needs to know the percentage markup for each product sold. Produce a query that outputs the product code, product name, buy price, MSRP in addition to
   1. The difference between MSRP and BuyPrice (i.e. MSRP - BuyPrice) called *markup*
   2. The percentage markup (100 \* (*markup* / BuyPrice)) called *percmarkup*rounded to 1 decimal place.
   3. *Sort the result according to percmarkup.*
   4. *Show products with percmarkup greater than 140.*

Graphical user interface, text, application

Description automatically generated

1. Display *product code (in the format shown in the sample below), product name, and quantity in stock information* of all products with the string ‘***co’*** in their product name. (c and o can be lower or upper case). *Sort the result according to quantity in stock.*



1. Display *customer number, contact first name, contact last name* for all customers whose contact first name starts with letter ***s*** (both lowercase and uppercase) and includes letter ***e*** (both lowercase and uppercase). *Sort the result according to customer number.*

