Lab 01 – Week 2 (Single-row Functions)

This week's lab continues using the SELECT command and learning the interfaces for both SQL Developer and introduces the use of single-line functions.

Submission

Your submission will be a single text-based SQL file with appropriate header and commenting. Please ensure your file runs when the entire file is executed in SQL Developer.

Create a new Worksheet in SQL Developer. Save the file as L01_ID#_LASTNAME.sql

Your submission needs to be commented and include the question, the solutions.

Do not comment the solutions (SQL Statements).

Example Submission

```
-- **********************
-- Name: Your Name
-- ID: #########
-- Date: The current date
-- Purpose: Lab 01 DBS311
-- **********************

-- Q1 SOLUTION --
SELECT * FROM TABLE;

-- Q2 Solution –
SELECT * FROM TABLE;
```

Style Guide

Your SQL should be written using the standard coding style:

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

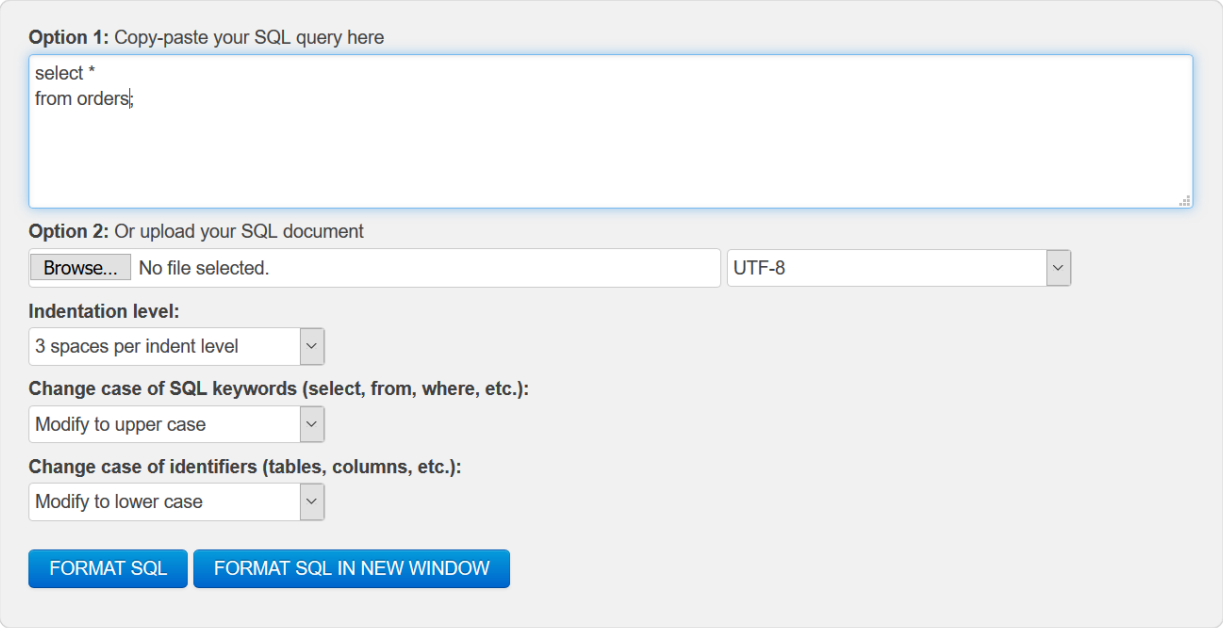
See the following sample:

```
SELECT columns
FROM tables
WHERE conditions
ORDER BY column1, column2;
```

To save time, you can write all SQL statement in your SQL developer. To make sure that your SQL statements style follows the standard SQL style guide, copy and paste your SQL statement onto the following website and click on "**FORMAT SQL**" or "**FORMAT SQL IN NEW WINDOW**".

https://www.freeformatter.com/sql-formatter.html#ad-output

You can also upload your SQL file. See the setting in the following image. Have SQL keywords (SELECT, INSERT, UPDATE, etc.) uppercase and user defined objects and identifiers (tables, columns, etc.) lowercase.



Tasks

-- **Q1:** Write a query to display the tomorrow's date in the following format:

```
January 10th of year 2019
```

the result will depend on the day when you RUN/EXECUTE this query. Label the column "Tomorrow".

***Advanced Option*:** Define an SQL variable called "tomorrow", assign it a value of tomorrow's date and use it in an SQL statement. Here the question is asking you to use a Substitution variable. Instead of using the constant values in your queries, you can use variables to store and reuse the values.

See the following example:

```
select *
from employees
where employee_id = 107;
```

You can also have the following code:

```
define emp_id number = 107;
select *
from employees
where employee_id = &emp_id;
```

After you use the variable, you can undefined the variable:

```
undefine emp_id;
```

Define a variable of type datetime:

```
define toay datetime = sysdate; -- Assigning current date to the today variable.
Or
define tomorrow = sysdate + 1;
```

-- **Q2:** For each product in category 2, 3, and 5, show product ID, product name, list price, and the new list price increased by 2%. Display a new list price as a whole number.

In your result, add a calculated column to show the difference of old and new list prices.

Sort the result according to category ID first and then based on product ID.

You output has to match the following result. This result is partially displayed as it has 158 rows.

See the result for the first 10 rows.

	Product ID	Product Name	LIST_PRICE	New Price	Price Difference
1	3	Corsair CB-9060011-WW	799.99	816	16.01
2	4	AMD 100-505989	2699.99	2754	54.01
3	5	PNY VCQK6000-PB	2290.79	2337	46.21
4	6	Zotac ZT-P10810A-10P	849.99	867	17.01
5	11	PNY VCQP5000-PB	2015.11	2055	39.89
6	12	Gigabyte GV-N108TAORUSX W-11GD	824.98	841	16.02
7	48	AMD FirePro S7000	1218.5	1243	24.5
8	58	Gigabyte GV-N108TAORUS X-11GD	784.98	801	16.02
9	83	Asus STRIX-GTX1080TI-O11G-GAMING	829.99	847	17.01
10	86	MSI GTX 1080 TI SEA HAWK X	804.98	821	16.02

-- **Q3:** For employees whose manager ID is 2, write a query that displays the employee's Full Name and Job Title in the following format:

Summer, Payne is Public Accountant.

Sort the result based on employee ID.

	Employee Info
1	Peterson, Mohammad is Finance Manager
2	Dunn, Annabelle is Administration Assistant
3	Spencer, Harper is Human Resources Representative
4	Gardner, Gracie is Public Relations Representative
5	Stephens, Rose is Accounting Manager

-- **Q4:** For each employee hired before October 2016, display the employee's last name, hire date and calculate the number of YEARS between TODAY and the date the employee was hired.

- Label the column Years worked.
- Order your results by the number of years employed. Round the number of years employed up to the closest whole number.

The output result includes 89 rows. See the partial result (The first 10 rows).

If you get the result in a different order, sort the result first based on the hire date column and then based on the number of years worked.

	Last Name	Hire Date	Years Worked
1	Richardson	03-JAN-16	4
2	Dixon	04-JAN-16	4
3	Wallace	05-JAN-16	4
4	Hawkins	13-JAN-16	4
5	Cooper	13-JAN-16	4
6	Patterson	14-JAN-16	4
7	Ramos	24-JAN-16	4
8	Wells	24-JAN-16	4
9	Hunter	24-JAN-16	4
10	Shaw	27-JAN-16	4

-- **Q5:** Display each employee's last name, hire date, and the review date, which is the first Tuesday after a year of service, but only for those hired after January 1, 2016.

- Label the column REVIEW DAY.
- Format the dates to appear in the format like:

  `TUESDAY, August the Thirty-First of year 2016`

  You can use ***ddspth*** to have the above format for the day.

- Sort by review date

The Query returns 107 rows. See the first 10 rows of the output result for comparision.

	Last Name	Hire Date	Review Date
1	Richardson	03-JAN-16	TUESDAY , January the tenth of year 2017
2	Dixon	04-JAN-16	TUESDAY , January the tenth of year 2017
3	Wallace	05-JAN-16	TUESDAY , January the tenth of year 2017
4	Hawkins	13-JAN-16	TUESDAY , January the seventeenth of year 2017
5	Patterson	14-JAN-16	TUESDAY , January the seventeenth of year 2017
6	Cooper	13-JAN-16	TUESDAY , January the seventeenth of year 2017
7	Ramos	24-JAN-16	TUESDAY , January the thirty-first of year 2017
8	Shaw	27-JAN-16	TUESDAY , January the thirty-first of year 2017
9	Jordan	30-JAN-16	TUESDAY , January the thirty-first of year 2017
10	Ford	29-JAN-16	TUESDAY , January the thirty-first of year 2017

-- **Q6:** For all warehouses, display warehouse id, warehouse name, city, and state. For warehouses with the null value for the state column, display "unknown". Sort the result based on the warehouse ID.

	Warehouse ID	Warehouse Name	City	State
1	1	Southlake, Texas	Southlake	Texas
2	2	San Francisco	South San Francisco	California
3	3	New Jersey	South Brunswick	New Jersey
4	4	Seattle, Washington	Seattle	Washington
5	5	Toronto	Toronto	Ontario
6	6	Sydney	Sydney	New South Wales
7	7	Mexico City	Mexico City	Distrito Federal,
8	8	Beijing	Beijing	Unknown
9	9	Bombay	Bombay	Maharashtra