**ha1-20191-DBS301 – 12 questions**

**Due: end of week 6 before Friday midnight.**

**Anything after that will not be accepted. It will get zero**

SUBMISSION:

1 Change the file to read a1-your email id --- of one of the members

2 Email the result to me and CC all member in the group

3 Not doing the above will result in a 25% deduction

This assignment can be done in groups. Ideal might be 2 to 4, but is not limited to that number.

Being part of a group is the same as being a part of a team for these assignments. When you submitted your work as part of a group you are saying that you understood what was submitted and that you fully participated with ALL the group members in getting a solution to the questions. It does not mean letting others do your work for you. For your full participation you get a mark equal to all the others in the group. If on the test, which is very much like the assignment, you cannot answer it well then you didn’t participate and understand the assignment but depended on others for the mark you received. That is very much like submitting their work and claiming it is your work.

Members in group

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student ID | Email address | Oracle ID |
| Zhi Zhao | 109079178 | zzhao58@myseneca.ca | dbs301\_191e39 |
| Senal Goonetilaka | 136471174 | mgoonetilaka@myseneca.ca | dbs301\_191e08 |
| Wing Tung Lau | 059155135 | wtlau1@myseneca.ca | dbs301\_191e19 |
| Hyunjung Kim | 148297179 | hkim293@myseneca.ca | dbs301\_191e15 |
| Po Yi Wong | 130017189 | pywong11@myseneca.ca | dbs301\_191e36 |

WHICH ID was used to do the submission dbs301\_191e08

**ALL WORK MUST BE DONE ON SENECA’S ORACLE ACCOUNT**

**SQL that does not follow easily readable standard formats will not be marked.**

NOTE:

If any output goes on for more than 100 lines, only cut and paste the first 100 or so. I don’t want to get this word document too big.

1 Display the customer number, customer name and country code for all the customers that are in the BELGIUM. The country code for Belgium is BE.

Please note that the user is to enter the 2 digit country code AND that they can enter it as ---BE, be, bE and other combinations. Your SQL must allow for various user inputs of the code. They are not entering the country name.

SELECT cust\_no AS "Customer No", cname AS "Customer Name", country\_cd AS "CD"

FROM customers

WHERE UPPER(country\_cd) = UPPER('&2\_digital');

**OUTPUT**

Customer No Customer Name CD

----------- ------------------------------ --

1113 Wally Mart 3 BE

1040 Vacation Central 2 BE

1066 Wilderness Wonderment Ltd BE

1090 Rock Steady 4 BE

2. For any GO Outlets list the customer number, customer name and order number if they ordered any of these products -- 40303, 40301, 40306.

Put result in customer number order.

**SQL**

SELECT c.cust\_no AS "Customer No", c.cname AS "Customer Name", o.order\_no AS "Order No"

FROM customers c JOIN orders o

ON c.cust\_no = o.cust\_no

JOIN orderlines ol ON ol.order\_no = o.order\_no

WHERE upper(c.cust\_type) = UPPER('Go Outlet') AND ol.prod\_no IN (40303, 40301, 40306)

ORDER BY 1;

**OUTPUT**

Customer No Customer Name Order No

----------- ------------------------------ ----------

1009 GO Outlet Kista AB 137

1011 GO Outlet Singapore 200

1098 GO Outlet Madrid 27

1120 GO Outlet London 49

1121 GO Outlet Manchester 15

1122 GO Outlet New York 218

1122 GO Outlet New York 278

7 rows selected.

3. List the customer number, customer name and order number for customers that ordered products 40310, 40303 and 60101. Only show customers with customer number more than 1140

Put result in order number with lowest first to highest order number.

**SQL**

SELECT c.cust\_no AS "Customer No", c.cname AS "Customer Name", o.order\_no "Order No"

FROM customers c JOIN orders o

ON c.cust\_no = o.cust\_no

JOIN orderlines l

ON o.order\_no = l.order\_no

WHERE l.prod\_no in (40310,40303,60101)

AND c.cust\_no > 1140

ORDER BY 3;

**OUTPUT**

Customer No Customer Name Order No

----------- ------------------------------ ----------

1148 Juan's Sports 2 2

1145 Florida Sun Sports 3 71

1147 Juan's Sports 1 183

4. Display all orders for United Kingdom. The word entered is United Kingdom and not UK.

Show only cities that start with L.

Display the (1) customer number, (2) customer name, (3) order number, (4) product name, (5) the total dollars for that line.

Give that last column the name of SALES

Put the output into customer number order from highest to lowest.

Display only customer numbers less than 1000

**SQL**

SELECT C.Cust\_NO, C.CNAME, O.order\_NO, P.prod\_name, OL.price\*OL.qty AS Sales

FROM orders O

INNER JOIN customers C ON O.cust\_no = C.cust\_NO

INNER JOIN countries CO ON C.Country\_CD = CO.Country\_ID

INNER JOIN orderlines OL ON O.order\_NO = OL.order\_NO

INNER JOIN Products P ON OL.Prod\_NO = P.PROD\_NO

WHERE UPPER(CO.country\_name) = UPPER('&country') AND (C.city) LIKE 'L%' AND C.cust\_no < 1000

ORDER BY 1 DESC;

**OUTPUT**

no rows selected

5. DO NOT USE ALIAS NAMES

Find the total dollar value for all orders from London. Each row will show customer name, order number and total dollars for the order.

Sort by highest total

**SQL**

SELECT c.cname, o.order\_no, SUM(price\*qty)

FROM customers c JOIN orders o

ON c.cust\_no = o.cust\_no

JOIN orderlines l

ON o.order\_no = l.order\_no

WHERE c.city = 'London'

GROUP BY c.cname, o.order\_no

ORDER BY 3 DESC;

**OUTPUT**

CNAME ORDER\_NO TOTAL\_DOLLARS

------------------------------ ---------- -------------

GO Outlet London 107 28466

GO Outlet London 61 17739

GO Outlet London 29 6580

GO Outlet London 49 6446

Trees to Seas Ltd 170 5728

Trees to Seas Ltd 122 5056

6 rows selected.

6. For all orders in the orders table supply order date and count of the number of orders on that date. Only include those from 2015 and have more then 1 order on that date.

**SQL**

SELECT order\_dt AS "Order Date", count(\*) AS "Number Of Orders"

FROM orders

WHERE substr(order\_dt,-4,4)='2015'

GROUP BY order\_dt

HAVING count(\*)>1

**OUTPUT**

Order Date Number Of Orders

----------- ----------------

02-Apr-2015 3

28-Jan-2015 2

28-Feb-2015 2

29-Mar-2015 2

23-Feb-2015 2

09-Mar-2015 2

20-Mar-2015 2

26-Mar-2015 2

10-Feb-2015 3

12-Feb-2015 2

01-Apr-2015 2

11 rows selected.

7. Mr.King would like to see all orders in 2014 from United Kingdom and Canada.

Show the (1) customer number, (2) customer name and (3)country name

**SQL**

SELECT c.cust\_no AS "Customer No", c.cname AS "Customer Name", ct.country\_name AS "Country Name"

FROM customers c JOIN orders o ON c.cust\_no = o.cust\_no

JOIN countries ct ON c.country\_cd = ct.country\_id

WHERE substr(order\_dt,-4,4)='2014'

AND ct.country\_name IN ('United Kingdom','Canada');

**OUTPUT**

Customer No Customer Name Country Name

----------- ------------------------------ ----------------------------------------

1111 Wally Mart 1 United Kingdom

1127 Fresh Air Co 1 United Kingdom

1128 Fresh Air Co 2 United Kingdom

1128 Fresh Air Co 2 United Kingdom

1128 Fresh Air Co 2 United Kingdom

1129 Fresh Air Co 3 United Kingdom

1129 Fresh Air Co 3 United Kingdom

1132 Excellence en Montagne Canada

1025 Fresh Air Co 5 United Kingdom

1033 Supras Camping Suplies 3 Canada

1042 Vacation Central 4 Canada

Customer No Customer Name Country Name

----------- ------------------------------ ----------------------------------------

1042 Vacation Central 4 Canada

1048 Supras Camping Supplies 5 Canada

1048 Supras Camping Supplies 5 Canada

1048 Supras Camping Supplies 5 Canada

1065 Trees to Seas Ltd United Kingdom

1065 Trees to Seas Ltd United Kingdom

1067 Andes Camping Supplies 1 Canada

1067 Andes Camping Supplies 1 Canada

1069 Andes Camping Supplies 3 Canada

1069 Andes Camping Supplies 3 Canada

1092 Clear Valley Waters 2 Canada

Customer No Customer Name Country Name

----------- ------------------------------ ----------------------------------------

1092 Clear Valley Waters 2 Canada

23 rows selected.

8. The President wants to know out of the 150 to 155 customers that are on file, how many customers have not placed an order?

**SQL**

**Solution1**

SELECT count(\*)

FROM customers c

WHERE c.cust\_no NOT IN (SELECT DISTINCT cust\_no

FROM orders)

**Solution 2**

SELECT count(c.cname)

FROM customers

LEFT JOIN orders using (suct\_no)

WHERE order\_no is null

**Solution 3**

Select count(\*)

FROM (

Select cust\_no

FROM customers

Minus

Select cust\_no

From orders)

**OUTPUT**

COUNT(\*)

-----------------

47

9. Show what customers (number and name) along with the country name for all customers that are in the same countries as the Supra customers. Limit the list to any customer that starts with the letters A or B.

**SQL**

SELECT c.cust\_no AS "Customer No", c.cname AS "Customer Name", ct.country\_name AS "Country"

FROM customers c JOIN countries ct

ON c.country\_cd = ct.country\_id

WHERE ct.country\_name IN (SELECT DISTINCT ct.country\_name

FROM customers c

JOIN countries ct

ON c.country\_cd = ct.country\_id

WHERE c.cname like '%Supra%')

AND c.cname like 'A%' OR c.cname like 'B%'

**OUTPUT**

Customer No Customer Name Country

----------- ------------------------------ ----------------------------------------

1133 Back woods up front Ltd. Canada

1134 Breathe Free ltd United Kingdom

1006 Bergsteiger G.m.b.H. Germany

1007 Backwoods Equipment(Rom) Australia

1008 Botanechi K.K. Japan

1010 Andes Camping Supplies 6 Canada

1012 Act'N'Up Fitness 5 United States of America

1059 Botanechi K.K. 1 Japan

1067 Andes Camping Supplies 1 Canada

1068 Andes Camping Supplies 2 Canada

1069 Andes Camping Supplies 3 Canada

Customer No Customer Name Country

----------- ------------------------------ ----------------------------------------

1070 Andes Camping Supplies 4 Canada

1072 Advanced Climbing Ltd United States of America

1075 Act'N'Up Fitness 1 United States of America

1076 Act'N'Up Fitness 2 United States of America

1077 Act'N'Up Fitness 3 United States of America

1078 Act'N'Up Fitness 4 United States of America

1086 All season camping goods United States of America

18 rows selected.

10. List the (1) employee number, (2) last name (3) job id and (4) the modified or not modified salary for all employees.

Show only employees -- If the salary without the increase is outside the range $6,000 – $11,000

- and who are employed as a Vice Presidents or Managers (President is not counted here).

- You should use Wild Card characters for this.

- the modified salary for a VP will be 30% higher

- and managers a 20% salary increase.

- Sort the output by the top salaries (before this increase).

The output lines should look like this sample line:

205 Higgins 14400

**SQL**

SELECT EMPLOYEE\_ID, LAST\_NAME, JOB\_ID,

CASE WHEN JOB\_ID like '%VP%' THEN SALARY\*1.3

WHEN JOB\_ID like '%MAN%' THEN SALARY\*1.2

END "Modified\_SALARY"

FROM EMPLOYEES

WHERE SALARY NOT BETWEEN 6000 AND 11000

AND (JOB\_ID like '%VP%' OR JOB\_ID like '%MAN%')

ORDER BY salary desc

**OUTPUT**

EMPLOYEE\_ID LAST\_NAME JOB\_ID Modified\_SALARY

----------- ------------------------- ---------- ----------------------------------------------------

101 Kochhar AD\_VP 22100

102 De Haan AD\_VP 22100

201 Hartstein MK\_MAN 15600

124 Mourgos ST\_MAN 6960

11. Display (1) Department\_id, (2) Job\_id and the (3) Lowest salary for this combination but only if that Lowest Pay falls in the range $6000 - $18000.

Exclude people who

(a) work as some kind of Representative (REP)job from this query and

(b) departments IT and SALES

Sort the output according to the Department\_id and then by Job\_id.

          You MUST NOT use the Subquery method.

**SQL**

SELECT e.department\_id AS "Department ID",e.job\_id AS "JOB ID", min(e.salary) AS "Salary"

FROM employees e

INNER JOIN departments d ON e.department\_id = d.department\_id

WHERE e.job\_id NOT like('%REP%')

AND d.department\_name NOT in ('IT','SALES')

GROUP BY e.department\_id, e.job\_id

HAVING min(e.salary) BETWEEN 6000 AND 18000

ORDER BY 1,2;

**OUTPUT**

Department ID JOB ID Salary

------------- ---------- ----------

20 MK\_MAN 13000

80 SA\_MAN 10500

90 AD\_VP 17000

110 AC\_ACCOUNT 8300

110 AC\_MGR 12000

12. Display (1) last\_name, (2) salary and (3) job for all employees who earn more than all lowest paid employees per department outside the US locations.

Exclude President and Vice Presidents from this query.

Sort the output by job title ascending.

          You need to use a Subquery and Joining with the NEWER method. (USING/JOIN)

**SQL**

SELECT e.last\_name AS "Last Name", e.salary AS "Salary", e.job\_id AS "JOB"

FROM employees e

WHERE salary > ALL(SELECT min(salary)

FROM employees e

INNER JOIN departments d

ON e.department\_id = d.department\_id

INNER JOIN locations l ON d.location\_id = l.location\_id

WHERE country\_id NOT IN 'US'

GROUP BY d.department\_id)

AND e.job\_id NOT LIKE '%VP%' AND e.job\_id NOT LIKE '%PRES%'

ORDER BY 3;

**OUTPUT**

Last Name Salary JOB

------------------------- ---------- ----------

Gietz 8300 AC\_ACCOUNT

Higgins 12000 AC\_MGR

Flertjan 17000 AC\_REP

Hunold 9000 IT\_PROG

Hartstein 13000 MK\_MAN

Zlotkey 10500 SA\_MAN

Strandherst 9000 SA\_REP

Litrand 10000 SA\_REP

Young 10000 SA\_REP

Testorok 11000 SA\_REP

Sanchez 11000 SA\_REP

Last Name Salary JOB

------------------------- ---------- ----------

Montoya 11000 SA\_REP

Mustaine 11000 SA\_REP

Smertal 11000 SA\_REP

Grovlin 11000 SA\_REP

Whiteduck 11000 SA\_REP

Abel 11000 SA\_REP

Termede 11000 SA\_REP

Krain 11000 SA\_REP

Gregson 11000 SA\_REP

Torson 11000 SA\_REP

Cornel 11000 SA\_REP

Last Name Salary JOB

------------------------- ---------- ----------

Gibbons 11000 SA\_REP

Pallomine 11000 SA\_REP

Jacobs 11000 SA\_REP

Brigade 11000 SA\_REP

Armarillo 11000 SA\_REP

Mot 11000 SA\_REP

Turcotte 11000 SA\_REP

LeBlanc 11000 SA\_REP

Rodriguez 11000 SA\_REP

Loo Nam 11000 SA\_REP

Chan 11000 SA\_REP

Last Name Salary JOB

------------------------- ---------- ----------

Wandiko 11000 SA\_REP

O'Brian 8600 SA\_REP

Chancevente 12000 SA\_REP

Bergsteige 8000 SA\_REP

Gruber 9000 SA\_REP

Harvey 10000 SA\_REP

39 rows selected.

**NOTE:**

**The assignment covers just some of the questions, but it is your ability to do the logic on your own that will result in a great test mark.**

**Major topics are functions, joins and subqueries**

**We did not test on returning 2 items from a subquery.**