Database Design II and SQL Using Oracle DBS301SEE.09112.2191 Course Documents

Review Test Submission: L4-functions

Review Test Submission: L4-functions

User	Mailanachchige Don Senal Goonetilaka
Course	Database Design II and SQL Using Oracle
Test	L4-functions
Started	1/31/19 2:52 PM
Submitted	2/1/19 10:09 PM
Due Date	2/1/19 11:59 PM
Status	Needs Grading
Attempt Score	Grade not available.
Time Elapsed	31 hours, 17 minutes
Instructions	Remember .... SQL and output whenever question allows for that response.

Remember .... SQL and output whenever question allows for that response. Also Oracle syntax only.

This should be completed in *week 4* if you are keeping up

Results Displayed	All Answers, Submitted Answers, Correct Answers

Question 1

Needs Grading

-- List all the countries that start with a letter entered by the user (prompt). BUT .. if user enters something else it should still work.

NOTE: The user must enter a lowercase letter

Selected Answer:

```
SELECT *
FROM countries
WHERE LOWER(country_name) like LOWER('&EnterLetter%');
```

Correct Answer:

```
old:SELECT *
FROM countries
WHERE country_name LIKE '&EnterrLetter%'
new:SELECT *
FROM countries
WHERE country_name LIKE 'a%'   -- used lowercase a
no rows selected
Because it didn't handle lowercase, then you need to
write it so it handles any case
```

Question 2

Needs Grading

-- let's make the previous question easier for the user !!!

SELECT * FROM countries

WHERE UPPER(country_name) LIKE UPPER('&EnterLetter%');

Try it by entering an UPPER and a LOWER value to see the effect. It is very different when compared to the previous questions results.

JUST ANSWER YES if you tried it.

Selected Answer: YES
Correct Answer: Hopefully ... YES

Question 3

Needs Grading

Calculate how many letters each country name has in it

and list them from most letters to least letters

Selected Answer:

```
SELECT country_name,LENGTH("COUNTRY_NAME")
FROM countries
ORDER BY LENGTH("COUNTRY_NAME") DESC;

COUNTRY_NAME                             LENGTH("COUNTRY_NAME")
---------------------------------------- ----------------------
United States of America                                   24
United Kingdom                                             14
Netherlands                                                11
Switzerland                                                11
Argentina                                                   9
Singapore                                                   9
Australia                                                   9
Zimbabwe                                                    8
HongKong                                                    8
Nigeria                                                     7
Belgium                                                     7

COUNTRY_NAME                             LENGTH("COUNTRY_NAME")
---------------------------------------- ----------------------
Denmark                                                     7
Germany                                                     7
Mexico                                                      6
Kuwait                                                      6
France                                                      6
Canada                                                      6
Brazil                                                      6
Israel                                                      6
Zambia                                                      6
India                                                       5
Egypt                                                       5

COUNTRY_NAME                             LENGTH("COUNTRY_NAME")
---------------------------------------- ----------------------
China                                                       5
Italy                                                       5
Japan                                                       5

25 rows selected.
```

Correct Answer:

```
SELECT  country_name, LENGTH(country_name) as "No. of Characters"
FROM countries
ORDER BY "No. of Characters" DESC;
COUNTRY_NAME                               No. of
Characters
---------------------------------------- ----------------
-
United States of America
24
United Kingdom
14
Netherlands
11
Switzerland
11
Argentina
9
Singapore
9
Australia
9
Zimbabwe
8
HongKong
8
Nigeria
7
Belgium
7
Denmark
7
Germany
7
Mexico
6
Kuwait
6
France
6
Canada
6
Brazil
6
Israel
6
Zambia
6
India
5
Egypt
5
China
5
Italy
5
Japan
5
 25 rows selected
```

Question 4

Needs Grading

Show the date 10 days before today (today is not hard coded but is supplied by the system)

Selected Answer:
```
SELECT SYSDATE "Today", SYSDATE -10 as "10DaysAgo"
FROM dual;

Today     10DaysAgo
---------  ---------
01-FEB-19  22-JAN-19
```

Correct Answer:
```
SELECT sysdate "to show today"
       sysdate - 10 AS "to show 10 Days Ago"
FROM dual;
```

Question 5

Needs Grading

What date is the next Saturday from now

Selected Answer:
```
SELECT  NEXT_DAY(sysdate,'Saturday') AS "Next Sat"
FROM dual;

Next Sat
---------
02-FEB-19
```

Correct Answer:
```
SELECT  NEXT_DAY(sysdate,'Saturday') AS "Next Saturday"
FROM dual
```

Question 6

Needs Grading

Display the difference between the Average pay and Lowest pay in the company.

Name this result *The gap*

Selected Answer:
```
SELECT AVG(salary)-MIN(salary) AS "The gap"
FROM employees;

The gap
----------
7472.22222
```

Correct Answer:
```
SELECT AVG(salary)-MIN(salary) AS "The gaP"
FROM employees;

   The gap
----------
7361.111111   --- you should round it for the user
```

Question 7

Needs Grading

Display the (1) department number and (2) Highest, (3) Lowest and (4)Average pay per each department. Do not label the columns .Round the average.

Sort the output so that the department with highest average salary is shown first.

Selected Answer:

```
SELECT DEPARTMENT_ID AS "Department Number", MAX(SALARY) AS
"Highest Pay", MIN(SALARY) AS "Lowest Pay", ROUND(AVG(SALARY),0)
AS "Average Salary"
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID
ORDER BY 4 DESC;
```

Department Number	Highest Pay	Lowest Pay	Average Salary
90	24000	17000	19333
10	17000	4400	10700
80	12000	7000	10546
110	12000	8300	10150
20	13000	6000	9500
	7000	7000	7000
60	9000	4200	6400
50	5800	2500	3500

8 rows selected.

Correct Answer:

```
SELECT department_id, max(salary), min(salary),
round(avg (salary),0)
FROM employees
GROUP BY department_id
ORDER BY max(salary)DESC;
DEPARTMENT_ID MAX(SALARY) MIN(SALARY)
ROUND(AVG(SALARY),0)
------------- ----------- ----------- ---------------
-----
           90       24000       17000
19333
           10       17000        4400
10700
           20       13000        6000
9500
          110       12000        8300
10150
           80       12000        7000
10546
           60        9000        4200
6400
                     7000        7000
7000
           50        5800        2500
3500
 8 rows selected
```

Question 8

Needs Grading

Display how many people work the same job in the same department.

Name these headings results as .... No., Job, How Many.

Include only jobs that involve more than one person.

Sort the output so that jobs with the most people involved are shown first.

Selected Answer:

```
SELECT DEPARTMENT_ID AS "NO", JOB_ID AS
"JOB",COUNT(EMPLOYEE_ID)AS "HOW MANY"
FROM EMPLOYEES
GROUP BY DEPARTMENT_ID,JOB_ID
HAVING COUNT(EMPLOYEE_ID)>1
ORDER BY 3 DESC;
```

NO	JOB	HOW MANY
80	SA_REP	34
50	ST_CLERK	4
60	IT_PROG	3
	SA_REP	2
90	AD_VP	2

Correct Answer:

Will have different alias

50	5	5
80	3	3
90	3	3
60	3	3
20	2	2
110	2	2

Question 9

Needs Grading

Remember .... SQL and output whenever question allows for that response. Also Oracle syntax only.

This should be completed in *week 4* if you are keeping up.

For each job ID display the job iD and total amount paid each month for this type of the job. Exclude titles *AD_PRES* and *AD_VP* and also include only jobs that require or exceed more than $15,000.

Sort the output so that top paid jobs are shown first.

Selected Answer:

```
SELECT JOB_ID, SUM(SALARY)
FROM EMPLOYEES
WHERE JOB_ID != 'AD_PRES' AND JOB_ID !='AD_VP'
GROUP BY JOB_ID HAVING SUM(SALARY)>=15000
ORDER BY SUM(SALARY) DESC;

JOB_ID    SUM(SALARY)
----------   -----------
SA_REP        372600
IT_PROG        19200
AC_REP         17000
```

Correct Answer:

```
SELECT JOB_ID, SUM(SALARY) AS "Monthly Total"
FROM EMPLOYEES
GROUP BY JOB_ID
HAVING JOB_ID != 'AD_VP'
  AND JOB_ID != 'AD_PRES'
  AND SUM(SALARY) > 15000
ORDER BY STDDEV(SALARY);



JOB_ID            Sum
---------- ----------
SA_REP         383600
IT_PROG         19200
```

Question 10

Needs Grading

For each department show the latest and earliest hire date, BUT
- exclude departments 10, 30 and 40

- also exclude those departments where the last person was hired in this century (2000 plus).

- Sort the output so that the most recent, meaning latest hire dates, are shown first.

Selected Answer:

```
SELECT department_id, MIN(hire_date) AS "Min", MAX(hire_date) AS
"MAX"
FROM employees
WHERE department_id NOT IN (10, 30, 40)
GROUP BY department_id;

DEPARTMENT_ID         Min              MAX
-------------         ---------        ---------
           20          17-FEB-96        17-AUG-97
           50          17-OCT-95        16-NOV-99
           60          03-JAN-90        07-FEB-99
           80          11-MAY-96       27-JUL-17
           90          17-JUN-87        13-JAN-93
          110          07-JUN-94        07-JUN-94

6 rows selected.
```

Correct Answer:

Output needs fixing to handle --> hired in this century (2000 plus

```
select department_id, min(hire_date), max(hire_date)
from employees
where department_id NOT IN (10, 30, 40)
group by department_id;
```

select department_id, min(hire_date), max(hire_date) from employees where department_id NOT IN (10, 30, 40) group by department_id;

```
DEPARTMENT_ID MIN(HIRE_DATE) MAX(HIRE_DATE)
------------- -------------- --------------
           20 17-FEB-96      17-AUG-97
           50 17-OCT-95      16-NOV-99
           60 03-JAN-90      07-FEB-99
           80 11-MAY-96      27-JUL-17
           90 17-JUN-87      13-JAN-93
          110 07-JUN-94      07-JUN-94
  6 rows selected
```

Question 11

Needs Grading

List all the countries and replace all letter "a"'s with a space.

Selected Answer:

```
SELECT REPLACE(COUNTRY_NAME,'a',' ')
FROM COUNTRIES;

REPLACE(COUNTRY_NAME,'A','')
----------------------------------------
Argentin
Austr li
Belgium
Br zil
C n d
Switzerl nd
Chin
Germ ny
Denm rk
Egypt
Fr nce

REPLACE(COUNTRY_NAME,'A','')
----------------------------------------
HongKong
Isr el
Indi
It ly
J p n
Kuw it
Mexico
Nigeri
Netherl nds
Sing pore
United Kingdom

REPLACE(COUNTRY_NAME,'A','')
----------------------------------------
United St tes of Americ
Z mbi
Zimb bwe

25 rows selected.
```

Correct Answer:

```
SELECT  country_name,
REPLACE(country_name,'a',' ') AS  "New"
FROM countries;

Kuwait                                   Kuw
it
Mexico
 Mexico
Nigeria
Nigeri
Netherlands                              Netherl
nds
Singapore                                Sing
pore
United Kingdom                           United
Kingdom
United States of America                 United St tes
of Americ
Zambia                                   Z
mbi
Zimbabwe                                 Zimb
bwe
```

Question 12

Needs Grading

For each manager number display how many persons he / she supervises.

-- Exclude managers with numbers 100, 101 and 102 and

-- include only those managers that supervise more than 2 persons.

-- Sort the output so that manager numbers with the most supervised persons are shown first.

This is often on a test or a question like it.

Selected Answer:

```
SELECT MANAGER_ID AS "Manager ID",COUNT(EMPLOYEE_ID) AS
"Number Of Employees"
FROM EMPLOYEES
WHERE EMPLOYEE_ID !=100 AND EMPLOYEE_ID !=101 AND
EMPLOYEE_ID !=102
GROUP BY MANAGER_ID HAVING COUNT(EMPLOYEE_ID)>2
ORDER BY 2 DESC;
```

Manager ID	Number Of Employees
149	37
124	4
100	3

Correct Answer: no answer provided as this is often on a test or assignment

Question 13

Needs Grading

Select dept. ID, job and count of number employees as long as there are more than 2 employees with that job in a department.
Sort by department then by job within department
EXTRA if you want to do it:
Display each department id with department name and highest salary in that department

Selected Answer:

```
SELECT DEPARTMENT_ID, JOB_ID, COUNT(EMPLOYEE_ID)
FROM EMPLOYEES
GROUP BY JOB_ID, DEPARTMENT_ID HAVING
COUNT(EMPLOYEE_ID)>2
ORDER BY DEPARTMENT_ID;

DEPARTMENT_ID          JOB_ID     COUNT(EMPLOYEE_ID)
-------------          ----------          ------------------
         50                    ST_CLERK              4
         60                    IT_PROG               3
         80                    SA_REP               34
```

Correct Answer:

```
select  department_id, job_id, count(*)
 from employees
 group by job_id, department_id
 having count(*) > 2
order by 1,2
DEPARTMENT_ID JOB_ID       COUNT(*)
------------- ---------- ----------
           50 ST_CLERK           4
           60 IT_PROG            3
           80 SA_REP            34
```

Question 14

Needs Grading

TRY THIS ONE

List the customer number and how many orders they have placed. Only show those customers that have more than4 orders. List the customers based on the highest number of orders first down to the lowest.

Selected Answer:

```
SELECT cust_no, COUNT(channel)
FROM "ORDERS"
GROUP BY cust_no HAVING COUNT (channel)>4
ORDER BY 2 DESC;

  CUST_NO COUNT(CHANNEL)
---------- --------------
      1008             12
      1056             11
      1022              9
      1011              9
      1148              9
      1066              8
      1095              8
      1036              6
      1092              6
      1041              6
      1130              6

  CUST_NO COUNT(CHANNEL)
---------- --------------
      1085              5
      1102              5
      1038              5

14 rows selected.
```

Correct Answer:

```
SELECT cust_no, count(order_no)
FROM orders
GROUP BY cust_no
having count (order_no) > 4
order by 1;
   CUST_NO COUNT(ORDER_NO)
---------- ---------------
      1008              12
      1011               9
      1022               9
      1036               6
      1038               5
      1041               6
      1056              11
      1066               8
      1085               5
      1092               6
      1095               8
      1102               5
      1130               6
      1148               9
 14 rows selected
```

Friday, February 1, 2019 10:13:01 PM EST

← **OK**