Fatima Jinnah Women University
Department of Software Engineering

LAB 6

Name: Raifa Khalid

Reg. no: 2020-BSE-024

Section: A

Semester: Fourth

Course: Data Base (LAB)

EXAMPLES:

NVL Function:

```
SQL> select ename, sal, nvl(comm,0) from emp;

ENAME             SAL NVL(COMM,0)
---------- ---------- -----------
SMITH             800           0
ALLEN            1600         300
WARD             1250         500
JONES            2975           0
MARTIN           1250        1400
BLAKE            2850           0
CLARK            2450           0
SCOTT            3000           0
KING             5000           0
TURNER           1500           0
ADAMS            1100           0

ENAME             SAL NVL(COMM,0)
---------- ---------- -----------
JAMES             950           0
FORD             3000           0
MILLER           1300           0

14 rows selected.
```

```
SQL> select ename, sal, comm,nvl2(comm,'sal+comm','sal') from emp;

ENAME             SAL       COMM NVL2(COM
---------- ---------- ---------- --------
SMITH             800            sal
ALLEN            1600        300 sal+comm
WARD             1250        500 sal+comm
JONES            2975            sal
MARTIN           1250       1400 sal+comm
BLAKE            2850            sal
CLARK            2450            sal
SCOTT            3000            sal
KING             5000            sal
TURNER           1500          0 sal+comm
ADAMS            1100            sal

ENAME             SAL       COMM NVL2(COM
---------- ---------- ---------- --------
JAMES             950            sal
FORD             3000            sal
MILLER           1300            sal

14 rows selected.
```

COALESCE Function:

```
SQL> select ename, coalesce(mgr,comm,-1) from emp;

ENAME      COALESCE(MGR,COMM,-1)
---------- ---------------------
SMITH                       7902
ALLEN                       7698
WARD                        7698
JONES                       7839
MARTIN                      7698
BLAKE                       7839
CLARK                       7839
SCOTT                       7566
KING                          -1
TURNER                      7698
ADAMS                       7788

ENAME      COALESCE(MGR,COMM,-1)
---------- ---------------------
JAMES                       7698
FORD                        7566
MILLER                      7782

14 rows selected.
```

CASE expression:

```
SQL> select ename, job, sal, case job when 'CLERK' then 1.10*sal when 'ANALYST' then 1.15*sal else sal end "Revised salary" from emp;

ENAME      JOB              SAL Revised salary
---------- --------- ---------- --------------
SMITH      CLERK            800            880
ALLEN      SALESMAN        1600           1600
WARD       SALESMAN        1250           1250
JONES      MANAGER         2975           2975
MARTIN     SALESMAN        1250           1250
BLAKE      MANAGER         2850           2850
CLARK      MANAGER         2450           2450
SCOTT      ANALYST         3000           3450
KING       PRESIDENT       5000           5000
TURNER     SALESMAN        1500           1500
ADAMS      CLERK           1100           1210

ENAME      JOB              SAL Revised salary
---------- --------- ---------- --------------
JAMES      CLERK            950           1045
FORD       ANALYST         3000           3450
MILLER     CLERK           1300           1430

14 rows selected.
```

DECODE function:

```
SQL> select ename, job, sal, decode(job,'CLERK',1.10*sal,'ANALYST',1.15*sal)  "Revised salary" from emp;

ENAME      JOB              SAL Revised salary
---------- --------- ---------- --------------
SMITH      CLERK            800            880
ALLEN      SALESMAN        1600
WARD       SALESMAN        1250
JONES      MANAGER         2975
MARTIN     SALESMAN        1250
BLAKE      MANAGER         2850
CLARK      MANAGER         2450
SCOTT      ANALYST         3000           3450
KING       PRESIDENT       5000
TURNER     SALESMAN        1500
ADAMS      CLERK           1100           1210

ENAME      JOB              SAL Revised salary
---------- --------- ---------- --------------
JAMES      CLERK            950           1045
FORD       ANALYST         3000           3450
MILLER     CLERK           1300           1430

14 rows selected.
```

AVG Function:

```
SQL> select avg(sal) from emp;

  AVG(SAL)
----------
2073.21429
```

SUM Function:

```
SQL> select sum(sal) from emp;

  SUM(SAL)
----------
     29025
```

MIN Function:

```
SQL> select min(sal) from emp;

  MIN(SAL)
----------
       800
```

MAX Function:

```
SQL> select min(hiredate) from emp;

MIN(HIRED
---------
17-DEC-80

SQL> select min(hiredate),max(hiredate) from emp;

MIN(HIRED MAX(HIRED
--------- ---------
17-DEC-80 23-MAY-87
```

COUNT Function:

```
SQL> select count(*) from emp where deptno=10;

  COUNT(*)
----------
         3
```

```
SQL> select count(comm) from emp where deptno=30;

COUNT(COMM)
-----------
          4
```

DISTINCT Keyword:

```
SQL> select count(distinct deptno) from emp;

COUNT(DISTINCTDEPTNO)
---------------------
                    3
```

NVL Function:

```
SQL> select avg(comm) from emp;

 AVG(COMM)
----------
       550

SQL> select avg(nvl(comm,0)) from emp;

AVG(NVL(COMM,0))
----------------
      157.142857
```

GROUP BY Clause:

```
SQL> select deptno, avg(sal) from emp group by deptno;

    DEPTNO   AVG(SAL)
---------- ----------
        30 1566.66667
        20       2175
        10 2916.66667
```

```
SQL> select avg(sal) from emp group by deptno;

  AVG(SAL)
----------
1566.66667
      2175
2916.66667
```

```
SQL> select deptno, job, sum(sal) from emp group by deptno,job;

    DEPTNO JOB         SUM(SAL)
---------- --------- ----------
        20 CLERK           1900
        30 SALESMAN        5600
        20 MANAGER         2975
        30 CLERK            950
        10 PRESIDENT       5000
        30 MANAGER         2850
        10 CLERK           1300
        10 MANAGER         2450
        20 ANALYST         6000

9 rows selected.
```

HAVING Clause:

```
SQL> select deptno, max(sal) from emp group by deptno having max(sal)>1000;

    DEPTNO   MAX(SAL)
---------- ----------
        30       2850
        20       3000
        10       5000

SQL> select job, sum(sal) from emp where job not like 'CLERK' group by job having sum(sal)>1000 order by sum(sal);

JOB         SUM(SAL)
--------- ----------
PRESIDENT       5000
SALESMAN        5600
ANALYST         6000
MANAGER         8275
```

NESTING GROUP FUNCTIONS:

```
SQL> select max(avg(sal)) from emp group by deptno;

MAX(AVG(SAL))
-------------
   2916.66667
```

TASKS

1. **Create a query that displays the employee's name and commission amounts. If an employee does not earn commission, put "No Commission". Label the column COMM.**

```
SQL> select ename,comm, nvl2(comm,'commission','No commission') "COMM" from emp;

ENAME            COMM COMM
---------- ---------- -------------
SMITH                 No commission
ALLEN              300 commission
WARD               500 commission
JONES                 No commission
MARTIN            1400 commission
BLAKE                 No commission
CLARK                 No commission
SCOTT                 No commission
KING                  No commission
TURNER               0 commission
ADAMS                 No commission

ENAME            COMM COMM
---------- ---------- -------------
JAMES                 No commission
FORD                  No commission
MILLER                No commission

14 rows selected.
```

2. **Using the DECODE function, write a query that displays the grade of all employees based on the value of the column JOB_ID, as per the following data**

```
SQL> select job,decode(job,'PRESIDENT','A','MANAGER','B','ANALYST','C','SALESMAN','D','CLERK','E') as grade from emp;

JOB       G
--------- -
CLERK     E
SALESMAN  D
SALESMAN  D
MANAGER   B
SALESMAN  D
MANAGER   B
MANAGER   B
ANALYST   C
PRESIDENT A
SALESMAN  D
CLERK     E

JOB       G
--------- -
CLERK     E
ANALYST   C
CLERK     E

14 rows selected.
```

3. **Rewrite the statement in the preceding question using the CASE syntax.**

```
SQL> select job, case job when 'PRESIDENT' then 'A' when 'MANAGER' then 'B' when 'ANALYST' then 'C' when 'SALESMAN' then
'D' when 'CLERK' then 'E' end "grade" from emp;

JOB       g
--------- -
CLERK     E
SALESMAN  D
SALESMAN  D
MANAGER   B
SALESMAN  D
MANAGER   B
MANAGER   B
ANALYST   C
PRESIDENT A
SALESMAN  D
CLERK     E

JOB       g
--------- -
CLERK     E
ANALYST   C
CLERK     E

14 rows selected.
```

4. **Display the highest, lowest, sum, and average salary of all employees. Label the column Maximum, Minimum, Sum and Average respectively.**

```
SQL> select max(sal)"Maximum",min(sal)"Minimum", sum(sal)"SUM", avg(sal) "Average" from emp;

   Maximum    Minimum        SUM    Average
---------- ---------- ---------- ----------
      5000        800      29025 2073.21429
```

5. **Modify the above query to display maximum, minimum, sum and average salary for each job type.**

```
SQL> select job, max(sal)"Maximum",min(sal)"Minimum", sum(sal)"SUM", avg(sal) "Average" from emp group by job;

JOB          Maximum    Minimum        SUM    Average
--------- ---------- ---------- ---------- ----------
CLERK           1300        800       4150     1037.5
SALESMAN        1600       1250       5600       1400
PRESIDENT       5000       5000       5000       5000
MANAGER         2975       2450       8275 2758.33333
ANALYST         3000       3000       6000       3000
```

6. **Write a query to display the number of people with the same job.**

```
SQL> select count (*) from emp group by job;

  COUNT(*)
----------
         4
         4
         1
         3
         2
```

7. **Determine the number of managers without listing them. Label the column "Number of Columns".**

```
SQL> select count(job)"Number of columns" from emp where job='MANAGER';

Number of columns
-----------------
                3
```

8. **Write a query that displays the difference between highest and lowest salaries. Label the column "Difference"**

```
SQL> select max(sal)-min(sal)"Difference" from emp;

Difference
----------
      4200
```

9. **Display the manager number and salary of the lowest paid employee for the manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is less than $6,000. Sort the output in descending order of salary.**

```
SQL> select mgr, min(sal) from emp where mgr is not null and job='MANAGER' group by mgr having min(sal)>6000 order by min(sal) desc;

no rows selected
```

10. **Create a query that will display the total number of employees and, of that total, the number of employees hired in 1981, 1982 and 1983. Create appropriate column headings.**

```
SQL> select count(*) "hiredate" from emp where to_char (hiredate,'YYYY') in (1981,1982,2983);

  hiredate
----------
        11
```

11. Create a query that display the job, salary for that job based on department number, and total salary for that job, for department 10, 20, 30 and 40, giving each column an appropriate heading.

```
SQL> select job,
  2  sum(case deptno when 10 then sal END) "dept 10",
  3  sum(case deptno when 20 then sal END) "dept 20",
  4  sum(case deptno when 30 then sal END) "dept30",
  5  sum(case deptno when 40 then sal END) "dept 40",
  6  sum(sal)"TOTAL"
  7  from emp
  8  group by job;

JOB         dept 10    dept 20     dept30    dept 40      TOTAL
--------- ---------- ---------- ---------- ---------- ----------
CLERK           1300       1900        950                 4150
SALESMAN                               5600                5600
PRESIDENT       5000                                       5000
MANAGER         2450       2975       2850                8275
ANALYST                    6000                           6000
```