CSE 4508 – RDBMS Lab

Lab 4

Instructor: Shahriar Ivan

Materials: Prof. Dr. Abu Raihan Mostofa Kamal

Create a table called Occupation, with a field called ID. It should have another field called "general" with options such as "Teacher" and a "Specific" field with values such as "School" or "University" for the general profession of Teacher. Similarly, if the general profession is "Engineer", specific values could be "CSE" or "EEE". Store the salary in a field as well. Insert values.

Task A:

1. Group using "general" and then "specific". Here you should display the count of the number of people in each general-specific subgroup, and order the displayed list according to this count.
2. For each general group display the minimum, maximum, average salary, total salary as well as the number of individuals in each group.
3. Group according to general, and only display the general groups whose average salary is greater or equal to the overall average salary of the entire table
4. Group by general along with the average salary of each group, and save this grouped form in a view. Using this view, select the name and average salary of the group with the lowest average salary.

Task B:
Create a table called Grades with fields ID, Department (CSE, EEE, etc), Course Code (CSE 4508, CSE 4551, etc) and Grade(A, A+, A-, etc)

1. Show the hierarchical count of the number of individuals, based on Department, then Course Code and then Grades. Order them alphabetically, based on
   Department first, then Course, and so on. (Hint: Rollup)
2. Show the count across all possible combinations of these four dimensions (Hint: Cube)

Solution

TASK A

```
create table Occupation(
id int primary key,
general varchar2(10),
specific varchar2(10),
salary int,
constraint chk_1 check (
general in ('Eng') and specific in ('Mech','CSE')
),
constraint chk_2 check (
general in ('Teacher') and specific in ('School','Uni')
),
);
insert into Occupation values (1, 'Teacher', 'School', 20000);
insert into Occupation values (2, 'Eng', 'CSE', 30000);
insert into Occupation values (3, 'Teacher', 'Uni', 40000);
insert into Occupation values(4, 'Eng', 'Mech', 50000);
insert into Occupation values (5, 'Eng', 'CSE', 60000);
insert into Occupation values (6, 'Teacher', 'Uni', 70000);
```

1.

```
select general,specific,count(id)
from Occupation
group by general,specific
order by count(id);
```

```
SQL> select general,specific,count(id)
  2  from Occupation
  3  group by general,specific
  4  order by count(id);

GENERAL    SPECIFIC    COUNT(ID)
---------- ---------- ----------
Eng        Mech                1
Teacher    School              1
Eng        CSE                 2
Teacher    Uni                 2
```

2.

```
select general, min(salary), max(salary), avg(salary), count (id)
from Occupation
group by general;
```

```
SQL> select general, min(salary), max(salary), avg(salary), count (id)
  2  from Occupation
  3  group by general;

GENERAL    MIN(SALARY) MAX(SALARY) AVG(SALARY)  COUNT(ID)
---------- ----------- ----------- ----------- ----------
Eng              30000       60000  46666.6667          3
Teacher          20000       70000  43333.3333          3
```

3.

```
select general,avg(salary)
from Occupation
```

```
group by general
having avg(salary) >= (
select avg(salary) from Occupation
);
```

```
SQL> select general,avg(salary)
  2  from Occupation
  3  group by general
  4  having avg(salary) >= (
  5  select avg(salary) from occupation
  6  );

GENERAL     AVG(SALARY)
---------- -----------
Eng          46666.6667
```

4.

```
-- view creation
create view AVGSAL as
select general,avg(salary) as avg_salary
from Occupation
group be general;
--query
select general,avg_salary from AVGSAL
where avg_salary =(
select min(avg_salary) from AVGSAL);
```

```
SQL> create view AVGSAL as
  2  select general,avg(salary) as avg_salary
  3  from Occupation
  4  group by general;

View created.
```

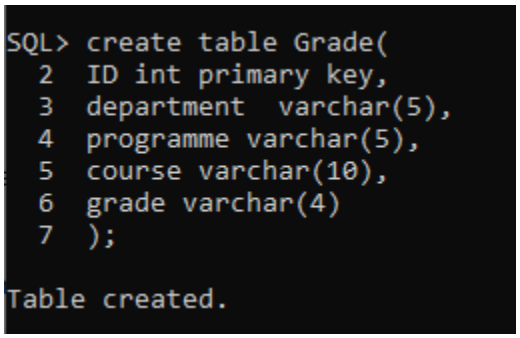
```
SQL> select general,avg_salary from AVGSAL
  2  where avg_salary =(
  3  select min(avg_salary) from AVGSAL);

GENERAL    AVG_SALARY
---------- ----------
Teacher    43333.3333
```

TASK B

```
create table Grade(
ID int primary key,
department varchar(5),
programme varchar(5),
course varchar(10),
grade varchar(4)
);

insert into Grade values(1, 'cse', 'Bsc', 'cs-101', 'A+');
insert into Grade values(2, 'eee', 'Bsc', 'ee-101', 'A-');
insert into Grade values(3, 'cse', HD', 'sw-101', 'A');
insert into Grade values(4, 'btm', HD', 'bba-101', 'B+');
insert into Grade values(5, 'mce', 'Bsc', 'ce-101', 'A');
insert into Grade values(6, 'mpe', 'Bsc', 'me-101', 'A+');
insert into Grade values(7, 'cse', 'HD', 'cs-101', 'A');
```



1.

select department, programme, course, grade, count(id)

from Grade

group by rollup(department,programme,course,grade)

order by department, programme, course, grade;

```
SQL> select department, programme, course, grade, count(id)
  2  from Grade
  3  group by rollup(department,programme,course,grade)
  4  order by department,programme,course,grade;

DEPAR PROGR COURSE     GRAD  COUNT(ID)
----- ----- ---------- ---- ----------
btm   HD    bba-101    B+            1
btm   HD    bba-101                  1
btm   HD                             1
btm                                  1
cse   Bsc   cs-101     A+            1
cse   Bsc   cs-101                   1
cse   Bsc                            1
cse   HD    cs-101     A             1
cse   HD    cs-101                   1
cse   HD    sw-101     A             1
cse   HD    sw-101                   1

DEPAR PROGR COURSE     GRAD  COUNT(ID)
----- ----- ---------- ---- ----------
cse   HD                             2
cse                                  3
eee   Bsc   ee-101     A-            1
eee   Bsc   ee-101                   1
eee   Bsc                            1
eee                                  1
mce   Bsc   ce-101     A             1
mce   Bsc   ce-101                   1
mce   Bsc                            1
mce                                  1
mpe   Bsc   me-101     A+            1

DEPAR PROGR COURSE     GRAD  COUNT(ID)
----- ----- ---------- ---- ----------
mpe   Bsc   me-101                   1
mpe   Bsc                            1
mpe                                  1
                                     7

26 rows selected.
```

2.

```
select department, programme, course, grade, count(*) as count
from Grade
group by cube(department,programme,course,grade)
order by department,programme,course,grade;
```

```
SQL> select department, programme, course, grade, count(*) as count
  2  from Grade
  3  group by cube(department,programme,course,grade)
  4  order by department,programme,course,grade;

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
btm   HD    bba-101    B+              1
btm   HD    bba-101                    1
btm   HD               B+              1
btm   HD                               1
btm         bba-101    B+              1
btm         bba-101                    1
btm                    B+              1
btm                                    1
cse   Bsc   cs-101     A+              1
cse   Bsc   cs-101                     1
cse   Bsc              A+              1

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
cse   Bsc                              1
cse   HD    cs-101     A               1
cse   HD    cs-101                     1
cse   HD    sw-101     A               1
cse   HD    sw-101                     1
cse   HD               A               2
cse   HD                               2
cse         cs-101     A               1
cse         cs-101     A+              1
cse         cs-101                     2
cse         sw-101     A               1

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
cse         sw-101                     1
      HD    cs-101     A               1
      HD    cs-101                     1

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
      HD    sw-101     A               1
      HD    sw-101                     1
      HD               A               2
      HD               B+              1
      HD                               3
            bba-101    B+              1
            bba-101                    1
            ce-101     A               1
            ce-101                     1
            cs-101     A               1
            cs-101     A+              1

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
            cs-101                     2
            ee-101     A-              1
            ee-101                     1
            me-101     A+              1
            me-101                     1
            sw-101     A               1
            sw-101                     1
                       A               3
                       A+              2
                       A-              1
                       B+              1

DEPAR PROGR COURSE      GRAD       COUNT
----- ----- ---------- ---- ----------
                                       7

89 rows selected.

SQL>
```