



Department of Computer Science and Engineering  
**Islamic University of Technology (IUT)**  
A subsidiary organ of OIC

## **Lab Report 4**

### **CSE 4508 :RDBMS**

Name:A N M Zahid Hossain Milkan

Student ID:200041202

Section:2B

Semester:Winter (5th)

Academic Year:2022-23

**Date of Submission:05.09.2023**

## Task A: Creating Table For the given Scenario and Inserting Value

### SQL:

```
CREATE TABLE Occupation(
```

```
    ID INT PRIMARY KEY,  
    General VARCHAR(30),  
    Specific VARCHAR(30),  
    Salary DECIMAL(10,2)  
);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (1, 'Teacher', 'School', 45000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (2, 'Engineer', 'CSE', 450000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (3, 'Teacher', 'University', 145000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (4, 'Teacher', 'College', 115000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (5, 'Engineer', 'Electrical', 45000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)  
VALUES (6, 'Engineer', 'CSE', 350000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)
VALUES (7, 'Student', 'School', 1000.00);
```

```
INSERT INTO Occupation (ID, General, Specific, Salary)
VALUES (8, 'Student', 'University', 6000.00);
```

## Result :

```
SQL> CREATE TABLE Occupation(
  2     ID INT PRIMARY KEY,
  3     General VARCHAR(30),
  4     Specific VARCHAR(30),
  5     Salary DECIMAL(10,2)
  6 );

Table created.

SQL>
SQL> INSERT INTO Occupation (ID, General, Specific, Salary)
  2 VALUES (1, 'Teacher', 'School', 45000.00);

1 row created.

SQL>
SQL> INSERT INTO Occupation (ID, General, Specific, Salary)
  2 VALUES (2, 'Engineer', 'CSE', 450000.00);

1 row created.

SQL>
SQL> INSERT INTO Occupation (ID, General, Specific, Salary)
  2 VALUES (3, 'Teacher', 'University', 145000.00);

1 row created.

SQL>
SQL> INSERT INTO Occupation (ID, General, Specific, Salary)
  2 VALUES (4, 'Teacher', 'College', 115000.00);

1 row created.

SQL>
SQL> INSERT INTO Occupation (ID, General, Specific, Salary)
  2 VALUES (5, 'Engineer', 'Electrical', 45000.00);

1 row created.
```

**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.**

**SQL:**

```
SELECT
General,Specific,COUNT(*) AS Count
FROM Occupation
GROUP BY General,Specific
ORDER BY Count DESC;
```

**Results:**

GENERAL	SPECIFIC	COUNT
Engineer	CSE	2
Teacher	University	1
Student	University	1
Engineer	Electrical	1
Student	School	1
Teacher	School	1
Teacher	College	1

7 rows selected.

**Task A-2:For each general group display the minimum, maximum and average salary.**

**SQL:**

```
SELECT
General,
MIN(Salary) AS Min,
MAX(Salary) AS Max,
AVG(Salary) AS Avg
FROM Occupation
```

```
GROUP BY General;
```

## Result :

```
SQL> SELECT
  2  General,
  3  MIN(Salary) AS Min,
  4  MAX(Salary) AS Max,
  5  AVG(Salary) AS Avg
  6  FROM Occupation
  7  GROUP BY General;
```

GENERAL	MIN	MAX	AVG
Teacher	45000	145000	101666.667
Engineer	45000	450000	281666.667
Student	1000	6000	3500

**Task A-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.**

**SQL:**

```
SELECT General
FROM Occupation
GROUP BY General
HAVING AVG(Salary) >= (SELECT AVG(Salary) FROM
Occupation);
```

## Result:

```
SQL> SELECT General
      2 FROM Occupation
      3 GROUP BY General
      4 HAVING AVG(Salary) >= (SELECT AVG(Salary) FROM Occupation);

GENERAL
-----
Engineer
```

**Task A-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 highest average salary.**

## SQL:

```
CREATE VIEW GeneralAvgSalary AS
SELECT
    General,
    AVG(Salary) AS AvgSalary
FROM Occupation
GROUP BY General;
SELECT General, AvgSalary
FROM GeneralAvgSalary
ORDER BY AvgSalary DESC
FETCH FIRST ROW ONLY;
```

## Result:

```
SQL> SELECT General, AvgSalary
2  FROM GeneralAvgSalary
3  ORDER BY AvgSalary DESC
4  FETCH FIRST ROW ONLY;
```

GENERAL	AVGSALARY
Engineer	281666.667

```
SQL> |
```

## Task-B:

### SQL:

```
CREATE TABLE Grades (
    ID INT PRIMARY KEY,
    Department VARCHAR(255),
    Programme VARCHAR(255),
    CourseCode VARCHAR(255),
    Grade VARCHAR(255)
);

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (1, 'CSE', 'BSc', 'CSE 4508', 'A+');

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (2, 'CSE', 'BSc', 'CSE 4551', 'A+');

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
```

```
VALUES (3, 'EEE', 'HD', 'EEE 2001', 'A-');

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (4, 'EEE', 'BSc', 'EEE 2005', 'B');

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (5, 'CSE', 'HD', 'CSE 4302', 'A');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (6, 'EEE', 'BSc', 'EEE 2003', 'A+');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (7, 'CSE', 'HD', 'CSE 4406', 'B-');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (8, 'MPE', 'MSc', 'MPE 1302', 'A-');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (9, 'BTM', 'MSc', 'BTM 6202', 'A-');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (10, 'MPE', 'HD', 'MPE 1302', 'A+');

    INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
```



```
VALUES (11, 'CEE', 'MSc', 'CEE 5302', 'A-');

INSERT INTO Grades (ID, Department, Programme,
CourseCode, Grade)
VALUES (12, 'BTM', 'BSc', 'BTM 6302', 'B-');
```

## Result:

```
SQL> CREATE TABLE Grades (
  2     ID INT PRIMARY KEY,
  3     Department VARCHAR(255),
  4     Programme VARCHAR(255),
  5     CourseCode VARCHAR(255),
  6     Grade VARCHAR(255)
  7 );

Table created.

SQL> INSERT INTO Grades (ID, Department, Programme, CourseCode, Grade)
  2 VALUES(1, 'CSE', 'BSc', 'CSE 4508', 'A+');

1 row created.

SQL>
SQL> INSERT INTO Grades (ID, Department, Programme, CourseCode, Grade)
  2 VALUES (2, 'CSE', 'BSc', 'CSE 4551', 'A+');

1 row created.

SQL>
SQL> INSERT INTO Grades (ID, Department, Programme, CourseCode, Grade)
  2 VALUES (3, 'EEE', 'HD', 'EEE 2001', 'A-');

1 row created.

SQL>
SQL> INSERT INTO Grades (ID, Department, Programme, CourseCode, Grade)
  2 VALUES (4, 'EEE', 'BSc', 'EEE 2005', 'B');

1 row created.

SQL>
SQL> INSERT INTO Grades (ID, Department, Programme, CourseCode, Grade)
  2 VALUES (5, 'CSE', 'HD', 'CSE 4302', 'A');

1 row created.
```

## TASK B-1:

Show the hierarchical count of the number of individuals, based on Department, then Programme, then Course Code and then Grades. Order them alphabetically, based on Department first, then Programme, and so on. (Rollup).

## SQL:

```
SELECT CONCAT('Anm',' ','Zahid') as myName FROM dual;
SELECT INSTR('Bismillahir Rahmanir Rahim','Rahman') as
Indx FROM dual;
SELECT LOWER('ANM ZAHID') as myName FROM dual;
SELECT UPPER('anm zahid') as myName FROM dual;
SELECT LENGTH('ANM ZAHID') as namelen FROM dual;
SELECT LPAD('ANM',3,'D') as leftpad FROM dual;
SELECT RPAD('ANM',3,'D') as rightpad FROM dual;
SELECT LTRIM('  ANM  ') as lefttrim FROM dual;
SELECT RTRIM('  ANM  ') as righttrim FROM dual;
SELECT SUBSTR('anm zahid',4,8) as myName FROM dual ;
SELECT COUNT(*) FROM Customer;
```

## Result:

DEPARTMENT	COURSECODE	GRADE	COUNT
BTM	BSc	B-	1
	BSc		1
	BSc		1
	MSc	A-	1
	MSc		1
	MSc		1
			2
CEE	MSc	A-	1
	MSc		1
	MSc		1
			1
CSE	BSc	A+	1
	BSc	A+	1
	BSc		1
	BSc		1
	BSc		2
	HD	A	1
	HD	B-	1
	HD		1
	HD		1
	HD		2
			4
EEE	BSc	A+	1
	BSc	B	1
	BSc		1
	BSc		1
	BSc		2
	HD	A-	1
	HD		1
	HD		1
			3
MPE	HD	A+	1
	HD		1

DEPARTMENT COURSECODE GRADE COUNT

BTM	BSc	B-	1
	BSc		1
	BSc		1
	MSc	A-	1
	MSc		1
	MSc		1

		2	
CEE	MSc	A-	1
	MSc	1	
	MSc	1	
		1	
CSE	BSc	A+	1
	BSc	A+	1
	BSc	1	
	BSc	1	
	BSc	2	
	HD	A	1
	HD	B-	1
	HD	1	
	HD	1	
	HD	2	
		4	
EEE	BSc	A+	1
	BSc	B	1
	BSc	1	
	BSc	1	
	BSc	2	
	HD	A-	1
	HD	1	
	HD	1	
		3	
MPE	HD	A+	1
	HD	1	
	HD	1	
	MSc	A-	1
	MSc	1	
	MSc	1	
		2	
		12	

39 rows selected.

**TASK B-2: Show the count across all possible combinations of these four dimensions (Cube).**

**SQL:**

```
SELECT Department, Programme  
CourseCode, Grade ,  
COUNT(*) AS Count  
FROM Grades  
GROUP BY CUBE (Department, Programme, CourseCode, Grade)  
ORDER BY Department, Programme, CourseCode, Grade;
```

**RESULT:**

DEPARTMENT	COURSECODE	GRADE	COUNT
BTM	BSc	B-	1
	BSc	B-	1
	BSc		1
	BSc		1
	MSc	A-	1
	MSc	A-	1
	MSc		1
	MSc		1
	A-		1

A-	1
B-	1
B-	1
	1
	1
	2

CEE	MSc	A-	1
	MSc	A-	1
	MSc		1
	MSc		1
		A-	1
		A-	1
			1
			1

CSE	BSc	A+	1
	BSc	A+	1
	BSc	A+	2
	BSc		1
	BSc		1
	BSc		2
	HD	A	1
	HD	A	1
	HD	B-	1
	HD	B-	1
	HD		1
	HD		1
	HD		2
		A	1
		A	1
		A+	1
		A+	1
		A+	2
		B-	1
		B-	1
			1
			1
			1
			1
			4

EEE	BSc	A+	1
	BSc	A+	1

BSc	B	1
BSc	B	1
BSc		1
BSc		1
BSc		2
HD	A-	1
HD	A-	1
HD		1
HD		1
	A+	1
	A+	1
	A-	1
	A-	1
	B	1
	B	1
		1
		1
		1
		3

MPE	HD	A+	1
	HD	A+	1
	HD		1
	HD		1
	MSc	A-	1
	MSc	A-	1
	MSc		1
	MSc		1
		A+	1
		A+	1
		A-	1
		A-	1
			2
			2

BSc	A+	1
BSc	A+	1
BSc	A+	1
BSc	A+	3
BSc	B	1
BSc	B	1
BSc	B-	1
BSc	B-	1
BSc		1

DEPARTMENT	COURSECODE	GRADE COUNT
------------	------------	-------------

-----		
BSc		1
BSc		1
BSc		1
BSc		1
BSc		5
HD	A	1
HD	A	1
HD	A+	1
HD	A+	1
HD	A-	1
HD	A-	1
HD	B-	1
HD	B-	1
HD		1
HD		1
HD		1
HD		1
HD		4
MSc	A-	1
MSc	A-	1
MSc	A-	1
MSc	A-	3
MSc		1
MSc		1
MSc		1
MSc		3
	A	1
	A	1
	A+	1
	A+	1
	A+	1
	A+	1
	A+	4
	A-	1
	A-	1
	A-	1
	A-	1
	A-	4
	B	1
	B	1
	B-	1



B-	1
B-	2
	1
	1
	1
	1
	1
	1
	1
	1
	1
	2
	12

147 rows selected.