

Database Lab 7

GROUP BY + HAVING

Question 1

Find all the departments which have more than 4 employees. Use the label “Number of “Employees”.

The screenshot shows a database management tool interface. The top bar indicates the connection is 'tudublin (tudublin) - Warning - not supported'. The left sidebar shows the 'SCHEMAS' tree with 'tudublin' expanded, showing tables 'DEPT' and 'EMP', and views 'SALGRADE', 'Views', 'Stored Procedures', and 'Functions'. The main query editor displays the following SQL query:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", COUNT(e.empno) AS "Number of Employees"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno
4 GROUP BY d.deptno, d.dname
5 HAVING COUNT(e.empno) > 4;
6
```

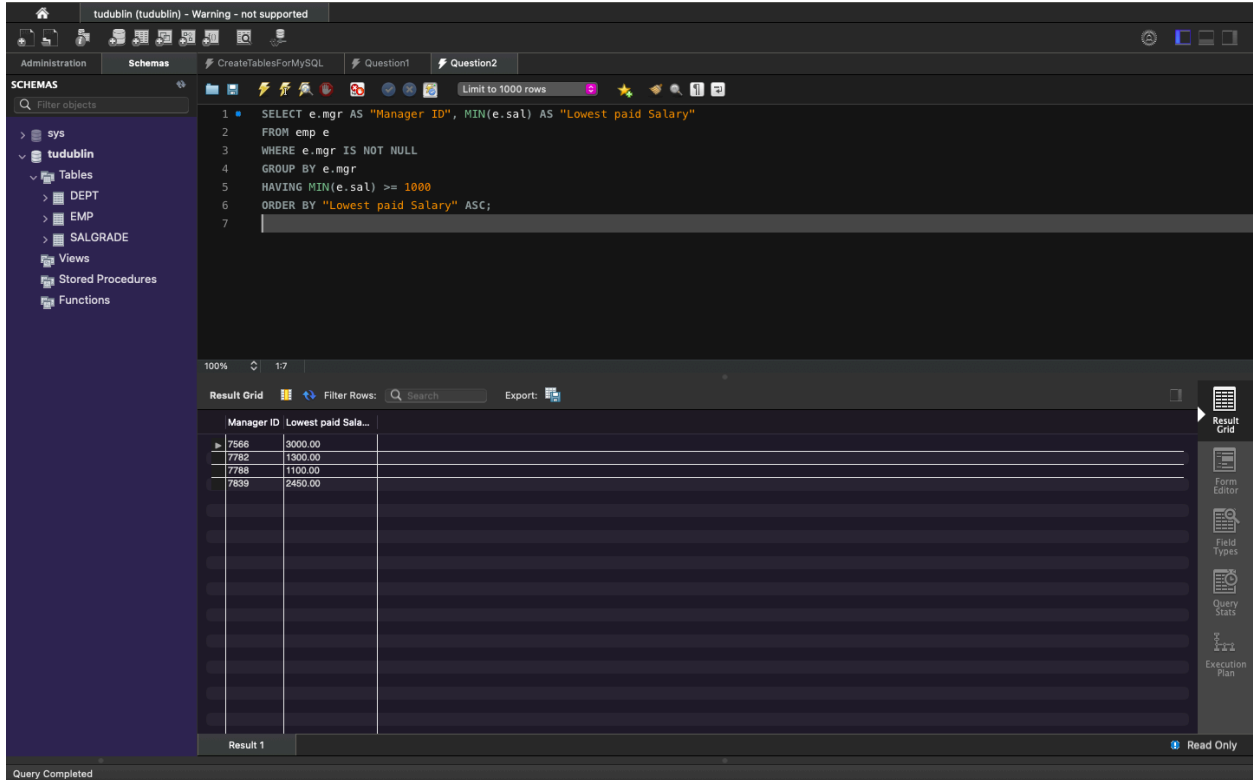
The 'Result Grid' at the bottom shows the results of the query. It has three columns: 'Department Num...', 'Department Na...', and 'Number of Emplo...'. The results are as follows:

Department Num...	Department Na...	Number of Emplo...
20	RESEARCH	5
30	SALES	6

The status bar at the bottom indicates 'Query Completed' and 'Read Only'.

Question 2

List the lowest salary for employees working for each manager. Exclude any groups where the salary is less than 1000. Sort the output by salary in ascending order. Use the heading “Lowest paid Salary”.



The screenshot shows a database management tool interface with a dark theme. The top bar indicates the database is 'tudublin (tudublin) - Warning - not supported'. The left sidebar shows the 'SCHEMAS' tree with 'tudublin' expanded, showing tables like DEPT, EMP, and SALGRADE. The main area displays a SQL query in a text editor:

```
1 SELECT e.mgr AS "Manager ID", MIN(e.sal) AS "Lowest paid Salary"
2 FROM emp e
3 WHERE e.mgr IS NOT NULL
4 GROUP BY e.mgr
5 HAVING MIN(e.sal) >= 1000
6 ORDER BY "Lowest paid Salary" ASC;
7
```

Below the query editor, the 'Result Grid' shows the query results. The grid has two columns: 'Manager ID' and 'Lowest paid Sala...'. The results are sorted by salary in ascending order. The first three rows are visible:

Manager ID	Lowest paid Sala...
7566	3000.00
7782	1300.00
7788	1100.00

The status bar at the bottom indicates 'Query Completed' and 'Read Only'.

INNER - JOINS

Question 3

Display each employee's number and name, and department number and name.

The screenshot shows a database management tool interface. The top menu bar includes 'Administration', 'Schemas', 'CreateTablesForMySQL', 'Question1', 'Question2', and 'Question3'. The 'Schemas' tab is active, showing a tree view on the left with 'sys' and 'tudublin' (expanded to show 'DEPT', 'EMP', 'SALGRADE', 'Views', 'Stored Procedures', and 'Functions'). The main area displays an SQL query:

```
1 SELECT e.empno AS "Employee Number", e.ename AS "Employee Name", d.deptno AS "Department Number", d.dname AS "Department Name"
2 FROM emp e
3 INNER JOIN dept d ON e.deptno = d.deptno;
4
```

Below the query, the 'Result Grid' shows the results of the query. The grid has four columns: 'Employee Number', 'Employee Name', 'Department Num...', and 'Department Na...'. The results are as follows:

Employee Number	Employee Name	Department Num...	Department Na...
7782	CLARK	10	ACCOUNTING
7839	KING	10	ACCOUNTING
7834	MILLER	10	ACCOUNTING
7369	SMITH	20	RESEARCH
7566	JONES	20	RESEARCH
7788	SCOTT	20	RESEARCH
7876	ADAMS	20	RESEARCH
7802	FORD	20	RESEARCH
7499	ALLEN	30	SALES
7621	WARD	30	SALES
7654	MARTIN	30	SALES
7668	BLAKE	30	SALES
7844	TURNER	30	SALES
7900	JAMES	30	SALES

The bottom status bar indicates 'SQL script saved to "/>

Question 4

Display the department number, name and location, and each employee's number and name.

The screenshot shows a database management interface with a SQL query editor and a results grid. The query is as follows:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", d.loc AS "Location", e.empno AS "Employee Number", e.ename AS "Employee Name"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno;
4
```

The results grid displays the following data:

Department Num...	Department Na...	Location	Employee Number	Employee Name
10	ACCOUNTING	NEW YORK	7782	CLARK
10	ACCOUNTING	NEW YORK	7839	KING
10	ACCOUNTING	NEW YORK	7934	MILLER
20	RESEARCH	DALLAS	7369	SMITH
20	RESEARCH	DALLAS	7566	JONES
20	RESEARCH	DALLAS	7788	SCOTT
20	RESEARCH	DALLAS	7676	ADAMS
20	RESEARCH	DALLAS	7902	FORD
30	SALES	CHICAGO	7499	ALLEN
30	SALES	CHICAGO	7521	WARD
30	SALES	CHICAGO	7654	MARTIN
30	SALES	CHICAGO	7698	BLAKE
30	SALES	CHICAGO	7844	TURNER
30	SALES	CHICAGO	7900	JAMES
40	OPERATIONS	BOSTON	HELL	HELL

The interface also includes a sidebar with a schema tree, a top menu bar, and a bottom status bar indicating the SQL script is saved to a file.

Question 5

Display each department number, and name and employee name.
Sort the employees and departments in ascending sequence.

The screenshot shows a database management interface with a SQL query editor and a results grid. The query is as follows:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", e.ename AS "Employee Name"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno
4 ORDER BY d.deptno ASC, e.ename ASC;
5
```

The results grid displays the following data:

Department Num...	Department Na...	Employee Name
10	ACCOUNTING	CLARK
10	ACCOUNTING	KING
10	ACCOUNTING	MILLER
20	RESEARCH	ADAMS
20	RESEARCH	FORD
20	RESEARCH	JONES
20	RESEARCH	SCOTT
20	RESEARCH	SMITH
30	SALES	ALLEN
30	SALES	BLAKE
30	SALES	JAMES
30	SALES	MARTIN
30	SALES	TURNER
30	SALES	WARD
40	OPERATIONS	NULL

The interface includes a sidebar with a tree view of schemas (sys, tudublin) and tables (DEPT, EMP, SALGRADE). The bottom status bar indicates "Query Completed".

Question 6

Display each department number and name and the total amount paid out in monthly salaries in each department. (SALARY + COMM).

The screenshot shows a database IDE interface with a SQL query editor and a results grid. The query is as follows:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", SUM(e.sal + IFNULL(e.comm, 0)) AS "Total Monthly Salaries"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno
4 GROUP BY d.deptno, d.dname;
```

The results grid displays the following data:

Department Num...	Department Na...	Total Monthly Salar...
10	ACCOUNTING	8750.00
20	RESEARCH	10875.00
30	SALES	11600.00
40	OPERATIONS	NULL

The interface also includes a sidebar with a schema tree, a top toolbar, and a bottom status bar indicating "Query Completed".

Question 7

Display each department's number and name and the number of employees employed in each department.

The screenshot shows a database IDE interface. The top toolbar includes icons for file operations and a 'Limit to 1000 rows' button. The 'SCHEMAS' panel on the left shows a tree view with 'sys' and 'tudublin' (containing tables, views, stored procedures, and functions). The central editor displays a SQL query:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", COUNT(e.empno) AS "Number of Employees"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno
4 GROUP BY d.deptno, d.dname;
5
```

Below the editor, the 'Result Grid' shows the query results. The grid has three columns: 'Department Num...', 'Department Na...', and 'Number of Employee...'. The data is as follows:

Department Num...	Department Na...	Number of Employee...
10	ACCOUNTING	3
20	RESEARCH	5
30	SALES	6
40	OPERATIONS	0

The bottom status bar indicates 'Query Completed' and 'Read Only'.

Question 8

Display each department's number and name and the number of employees employed in each department. Only display departments with more than 4 employees. (HAVING)

The screenshot shows a database management tool interface. The top bar indicates the connection is 'tudublin (tudublin) - Warning - not supported'. The main window is divided into several sections:

- SCHEMAS:** A sidebar on the left showing a tree view of the database structure. The 'tudublin' schema is expanded, showing tables like 'DEPT', 'EMP', and 'SALGRADE'.
- SQL Editor:** The central area contains a SQL query:

```
1 SELECT d.deptno AS "Department Number", d.dname AS "Department Name", COUNT(e.empno) AS "Number of Employees"
2 FROM dept d
3 LEFT JOIN emp e ON d.deptno = e.deptno
4 GROUP BY d.deptno, d.dname
5 HAVING COUNT(e.empno) > 4;
6
```
- Result Grid:** Below the SQL editor, a table displays the query results. The table has three columns: 'Department Num...', 'Department Na...', and 'Number of Employee...'. The results are as follows:

Department Num...	Department Na...	Number of Employee...
20	RESEARCH	5
30	SALES	6
- Tools:** On the right side, there are icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

At the bottom of the window, a status bar indicates 'Query Completed' and 'Read Only'.