**6G4Z0016 Databases**

**Introduction to MariaDB Databases and SQL**

**Part 6: Advanced Query Functions, Group Functions and GROUPBY**

**Aim**

The aim of this week’s labs is to:

* Write more advanced SQL queries using functions and groups

**Activities**

1. MySQL has several types of GROUP functions. For this unit you should know how to use: AVG, COUNT, MAX. MIN and SUM. Use the MariaDB knowledge base (<https://mariadb.com/kb/en/>) to help you answer the following questions:
2. Type in the following query and execute it.

SELECT AVG(salary), MAX(salary), MIN(salary), SUM(salary)

FROM employees

WHERE job\_id LIKE ‘%REP%’;

What does this query do? Record your answer in the box below.

1. Suppose we want to find out how many employees work in a specific department. Which of the group, a.k.a multiple row, functions would we use?

Write a query that counts the number of employees in department 90.

*TIP: check* [*https://mariadb.com/kb/en/count/*](https://mariadb.com/kb/en/count/)

1. Write a query to show the total number of different commission percentages that are currently earned by employees.

*TIP: to get only the different percentages you need to use a particular command. Check:* [*https://mariadb.com/kb/en/count-distinct/*](https://mariadb.com/kb/en/count-distinct/)*. Hint: Think about what you need to count and then specify that within the brackets that follow the count command.*

1. Group functions operate on sets of rows within a table(s) to give one result per group. Sometimes it is necessary to group the contents of a table into a number of smaller groups. This is achieved by using the GROUP BY clause. Type in and execute the following query which displays the average salary for employees in each department.

SELECT department\_id, AVG(salary)

FROM employees

GROUP BY department\_id;

Now remove, the GROUP BY statement and re-run the query. Observe what happens. Then remove “department\_id,” from the query and re-run the query. Observe what happens. In Oracle, if the GROUP BY is not there in this query it returns an error message. MySQL does not return an error but the output it returns doesn’t make sense - it returns the first department\_id it finds and places the average salary for all the departments next to it.

It is important that all columns in the SELECT list that are not in group functions must be in the GROUP BY otherwise you are likely to get nonsensical output.

If you would rather get an error than nonsensical output you can change MySQL to a mode more like that of Oracle by inputting the following command: SET sql\_mode = 'ONLY\_FULL\_GROUP\_BY’.

1. In SQL you cannot use the WHERE clause to restrict groups. You cannot use group functions in the WHERE clause. Instead, you can use the HAVING clause to achieve this result. Have a look at [*https://mariadb.com/kb/en/group-by/*](https://mariadb.com/kb/en/group-by/)(last example) and modify the query in (d) to display the average salary only for the employees with average salary greater than 8000. Write your answer in the box below:
2. Write a query to find the highest, lowest, sum and average salary of all employees for each job type. Label your columns as Maximum, Minimum, Sum and Average and ROUND them up. Hint – you will need to use the job\_id column from the employees table. Write your answer in the box below.
3. Write a query to display the number of people doing each job type. Make sure the job type is shown next to each count.
4. Write a query to find the difference between the highest and lowest salaries. Label the column “Difference”.