MySQL – Relational Databases: Workshop 3

Workshop contents:

- Running a script in MySQL Workbench to create the workshop database
- Writing SELECT queries containing the following elements:
    - o ORDER BY
    - o AS
    - o Functions and numerical operators
    - o Aggregate functions
    - o GROUP BY
    - o HAVING

Creating the workshop database tables:

1. Download the file "small_company_db.sql" from Canvas and save it your desktop. Open the file with Notepad++ and make sure you understand all the SQL queries it contains. Make sure you also understand the structure of the database tables being created.
2. Using MySQL Workbench, open a connection to your MySQL (Azure) database
3. In MySQL Workbench, manually delete all tables from your database (tables from previous workshop)
4. Import the new SQL file to create the new database tables and data used in this workshop

Running queries in MySQL Workbench: SELECT with: ORDER BY, AS, FUNCTIONS, AGGREGATE FUNCTIONS, GROUP BY, HAVING

For each of the items below, write a query that returns …

1. The project ID, Name, and Priority of all projects outside Helsinki listed in ascending order of priority
2. The project ID and person ID of all cases where the actual worked hours is greater than 90% of the planned hours (per person, per project)
3. The new salary of all persons that started in the company earlier than 2023 will be increased by 2,5%. Write a query that returns the person ID, first name, last name, and the new salary of all affected persons. Use "AS" to properly name the column where the new salary is listed
4. How many departments there are in the company?
5. What's the lowest tax rate among all persons?
6. How many projects occurred in Turku?
7. For how many days Mikko Nieminen has been working in the company? Hint: use a function called DATEDIFF() and CURRENT_DATE to help you finding the answer. "Google" "MySQL DATEDIFF" to figure out how to set its parameters.
8. The person id and the average amount of actual hours the person has worked in different projects. Round the average values to one decimal, and sort the result by descending order of average values

9. The average salary paid in different cities with the corresponding city name. Round the average values to integers (no decimals).
10. The IDs of persons that have worked in only two projects
11. The Department IDs and amount of persons that earn more than 3000 eur as a salary. Exclude persons from Helsinki in this query. (HINT: You will need to use a WHERE clause here)
12. The ID of departments having more than 2 persons working in it