Lab: Database Programmability and Transactions

This document defines the **exercise assignments** for the MySQL course @ Software University.
Please submit your solutions (source code) to all the below-described problems in Judge.

You are provided with the **soft_uni** database. Use it in the following assignments.

1. Count Employees by Town

Write a function **`ufn_count_employees_by_town(town_name)`** that accepts **`town_name`** as parameter and returns the count of employees who live in that town. Submit your queries using the "**MySQL Run Skeleton, run queries and check DB**" strategy.

Example

The following example is given with employees living in **Sofia**.

count
3

2. Employees Promotion

Write a stored procedure **`usp_raise_salaries(department_name)`** to raise the **`salary`** of all employees in given department as parameter by 5%. Submit your queries using the "**MySQL Run Skeleton, run queries and check DB**" strategy.

Example

The following example is given with employees in the "**Finance**" department ordered by **`first_name`**, then by **`salary`**.

first_name	salary
Barbara	27 720.00
Bryan	19 950.00
Candy	19 950.00
...	...

3. Employees Promotion by ID

Write a stored procedure **`usp_raise_salary_by_id(id)`** that raises a given employee's **`salary`** (by **`id`** as parameter) by **5%**. Consider that you cannot promote an employee that **doesn't exist** – if that happens, no changes to the database should be made. Submit your queries using the "**MySQL Run Skeleton, run queries and check DB**" strategy.

Example

The following example is given with **`employee_id = 17`**.

salary

14175.0000

4. Triggered

Create a table **`deleted_employees(employee_id PK, first_name,last_name,middle_name,job_title,deparment_id,salary)`** that will hold information about fired(deleted) employees from the employees table. Add a trigger to employees table that inserts the corresponding information in **`deleted_employees`**. Submit your queries using the "**MySQL Run Skeleton, run queries and check DB**" strategy.