**@SQL Introduction**

Bank of tomorrow

**Introduction:**

A local Bank contacted you with a request to create a database for managing its business. The management is providing you with information regarding the departments and business processes.

You are the database administrator/developer and business owner that needs to design and develop the necessary tables and procedures for data administration.

# 

# **Database design**

The company needs to manage data regarding the following entities

* Employees
* Clients
* Departments

You are not going to receive any specific information regarding the column identification. Try to extract as match as possible information from the business requirements.

**Create** a set of SQL query for creating the necessary tables for every entity. Try to normalise the information as much as possible. (**Read more about the data normalisation in RDBS**)

The tables must be properly built with all the necessary **Primary** and **Foreign** keys.

**Insert** as much information as possible. Not just a few entities. Try to extract as match information from the requirement as is necessary to properly test the concept.

**Update** specific database records in order to apply specific business query ( clients passwords for example )

**Delete** specific database records in order to apply specific business queries.

# 

# 

# **Business requirements**

The company provides you with a list of business procedures that must be implemented into the system as a SQL request.

**@General business queries**

* Find information about all departments names
* Find the salary of each employee
* Find the full name of each employee
* Create a query that produces an employ email address by using the employee first and last name. The user E-mail consists of first name and last name concatenated by a full stop. The domain of the E-mail is **fakecompany.com.** The result must be produced in a separate column named Full name
* Find all employ with job title Senior executive
* Find all employs with name starts with letter S
* Find all employees with a name that contains the letter l
* Find all employees that have a salary in the range 2000 – 3000
* Find all employees that have salaries 2500 / 3000 / 3500 / 5000
* Find all employees that do not have a manager
* Find all employees that have a senior position in the company and have a salary greater than 5000. Order them in decreasing order and by alphabetical order by there first name
* Find the top 5 best-paid employees

**@Advanced business queries**

* Find all employees and their addresses
* Find all employees and their manager
* Find all employees along with their manager and address
* Find all departments and all town names as a single list
* Find all of the employees and the manager for each of them along with the employees that do not have a manager
* Find all employees from “Sales” and all from “Finance” who are hired between 1995 and 2005
* Find the full name and salary of the employee that takes minimal salary in the company.
* Find the names and the salary of the employees that have a salary that is up to 10% higher than the minimum salary for the company
* Find the full name salary and the department of the employees that take the minimum salary in their department.
* Find the average salary in all department list the department name and the average salary
* Find the number of employee in all the departments. List the department and the number of employees in it
* Group all employee by the manager
* Find all employees whose names are exactly 5 characters

**@Extra requirements**

* Create a view that shows all clients that have been in the system today
* Change the passwords of all clients that are absent from the system since 10.03.2010
* Delete all clients without password
* Display the town with max employees

**@Transactions**

Create a transaction that deletes the information from the tables, drop all the tables and reroll the transaction at the end of the process

# **Development and code deployment**

Create all the necessary scripts for creating tables and business requirements and execute them in a proper manner based on the specifications that we provide you.

The produced code must execute without errors. Please provide as much data as possible in order to test the different scenarios.