

# Department of Computer Science and Engineering

|                               |              |
|-------------------------------|--------------|
| Course Code: CSE 370          | Credits: 3.0 |
| Course Name: Database Systems |              |

## Lab Homework 3

Proving yourself worthy of being able to handle bigger tasks, the tech lead has decided to give you a challenging job. However, this time, the data that you would be handling is very sensitive and no one wants this data to be leaked. Therefore, instead of getting the entire table, the tech lead has given you the list of attributes that the table contains and the table name. The information given is as follows:

|                                     |                |
|-------------------------------------|----------------|
| Table Name: <b><i>employees</i></b> |                |
| Attribute Name                      | Attribute type |
| <b><i>employee_id</i></b>           | char(10)       |
| <b><i>first_name</i></b>            | varchar(20)    |
| <b><i>last_name</i></b>             | varchar(20)    |
| <b><i>email</i></b>                 | varchar(60)    |
| <b><i>phone_number</i></b>          | char(14)       |
| <b><i>hire_date</i></b>             | date           |
| <b><i>job_id</i></b>                | char(10)       |
| <b><i>salary</i></b>                | int            |
| <b><i>commission_pct</i></b>        | decimal(5,3)   |
| <b><i>manager_id</i></b>            | char(10)       |
| <b><i>department_id</i></b>         | char(10)       |

You need to insert rows (around 10 or more to test all of the queries for the following tasks) as per the data types of the attributes/columns and problem statements of the following tasks.

Employee IDs can be 'EMP001', 'EMP002', etc. Job IDs can be 'JOB001', 'JOB002', etc. Manager IDs can be 'MNG001', 'MNG002', etc. **Department IDs should be DPT001, ..., DPT005, ... DPT007, etc.**

Write down the queries to retrieve the following information:

[7 X 2 =14]

1. Find the **first\_name, last\_name, email, phone\_number, hire\_date** and **department\_id** of all the employees with the latest **hire\_date**.
2. Find the **first\_name, last\_name, employee\_id, phone\_number, salary** and **department\_id** of all the employees with the lowest **salary** in each department.
3. Find the **first\_name, last\_name, employee\_id, commission\_pct** and **department\_id** of all the employees in the department 'DPT007' who have a lower **commission\_pct** than all of the employees of the department 'DPT005'.
4. Find the **department\_id** and total number of employees of each department which does not have a single employee under it with a **salary** more than 30,000.
5. For each of the departments, find the **department\_id, job\_id** and **commission\_pct** with **commission\_pct** less than at least one other **job\_id** in that department.
6. Find the **manager\_id** who does not have any employee under them with a **salary** less than 3500.
7. Find the **first\_name, last\_name, employee\_id, email, salary, department\_id** and **commission\_pct** of the employee who has the lowest **commission\_pct** under each manager.