

**Lab Assignment 3**  
Database System Lab  
CSE414  
Thursday 10<sup>th</sup> November, 2022

Submitted to:  
**Dr. Rudra Pratap Deb Nath**  
Associate Professor  
Department of Computer Science and Engineering  
University of Chittagong

Submitted by:  
**Kazi Omar Sharif**  
20701015  
Department of Computer Science and Engineering  
University of Chittagong



## 1 Chapter 4

1. Write a query to display the last name, department number, and department name for all employees.

```
1 SELECT E.LAST_NAME, E.DEPARTMENT_ID, D.DEPARTMENT_NAME
2 FROM HR.EMPLOYEES E , HR.DEPARTMENTS D
3 WHERE E.DEPARTMENT_ID=D.DEPARTMENT_ID ;
```

2. Create a unique listing of all jobs that are in department 30. Include the location of department 90 in the output.

```
1 SELECT DISTINCT E.JOB_ID , D.LOCATION_ID
2 FROM HR.EMPLOYEES E JOIN HR.DEPARTMENTS D USING (DEPARTMENT_ID
3 )
4 WHERE DEPARTMENT_ID=30 ;
```

3. Write a query to display the employee last name, department name, location ID, and city of all employees who earn a commission

```
1 select e.last_name , d.department_name, d.location_id, l.city
2 from hr.employees e, hr.departments d , hr.locations l
3 where ( e.department_id=d.department_id ) and
4 (d.location_id=l.location_id )
```

4. Display the employee last name and department name for all employees who have an a (lowercase) in their last names.

```
1
2 SELECT E.LAST_NAME , D.DEPARTMENT_NAME
3 FROM HR.EMPLOYEES E JOIN HR.DEPARTMENTS D USING (DEPARTMENT_ID
4 )
5 WHERE LAST_NAME like ' %a%' ;
```

5. Write a query to display the last name, job, department number, and department name for all employees who work in Toronto.

```

1
2 SELECT E.LAST_NAME ,E.JOB_ID,E.DEPARTMENT_ID ,D.DEPARTMENT_NAME
3 FROM HR.EMPLOYEES E JOIN HR.DEPARTMENTS D
4 ON (E.DEPARTMENT_ID=D.DEPARTMENT_ID)
5 JOIN HR.LOCATIONS L
6 ON (D.LOCATION_ID=L.LOCATION_ID)
7 WHERE LOWER(L.CITY)='toronto' ;

```

6. Display the employee last name and employee number along with their manager's last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, respectively.

```

1
2 SELECT E.LAST_NAME "Employee" ,E.EMPLOYEE_ID "EMP#" , D.
   last_name "Manager" ,
3 D.employee_id "Mgr#"
4
5 FROM HR.EMPLOYEES E JOIN HR.EMPLOYEES D
6 ON (E.MANAGER_ID=D.EMPLOYEE_ID);

```

7. display all employees including King, who has no manager. Order the results by the employee number.

```

1
2 SELECT E.LAST_NAME "Employee" ,E.EMPLOYEE_ID "EMP#" , D.
   last_name "Manager" ,
3 D.employee_id "Mgr#"
4
5 FROM HR.EMPLOYEES E
6 LEFT OUTER JOIN HR.EMPLOYEES D
7 ON (E.MANAGER_ID=D.EMPLOYEE_ID);

```

8. Create a query that displays employee last names, department numbers, and all the employees who work in the same department as a given employee. Give each column an appropriate label.

```

1 SELECT E.DEPARTMENT_ID ID ,E.LAST_NAME EMPLOYEE ,D.LAST_NAME
   COLLEAGUE
2 FROM HR.EMPLOYEES E JOIN HR.EMPLOYEES D
3 ON (E.DEPARTMENT_ID=D.DEPARTMENT_ID)
4 WHERE E.EMPLOYEE_ID <> D.EMPLOYEE_ID

```

```
5 ORDER BY E.DEPARTMENT_ID ,E.LAST_NAME,D.LAST_NAME
```

9.Show the structure of the JOB GRADE table. Create a query that displays the name, job, department name, salary, and grade for all employees.

```
1  
2 DESC JOB_GRADES  
3  
4 SELECT e.last_name, e.job_id, d.department_name,  
5 e.salary, j.grade_level  
6 FROM HR.employees e JOIN HR.departments d  
7 ON (e.department_id = d.department_id)  
8 JOIN HR.job_grades j  
9 ON (e.salary BETWEEN j.lowest_sal AND j.highest_sal);
```

10.Create a query to display the name and hire date of any employee hired after employee Davies.

```
1  
2 SELECT e.last_name, e.hire_date  
3 FROM employees e JOIN employees davies  
4 ON (davies.last_name = 'Davies')  
5 WHERE davies.hire_date < e.hire_date;
```

11.Display the names and hire dates for all employees who were hired before their managers, along with their manager's names and hire dates. Label the columns Employee, Emp Hired, Manager, and Mgr Hired, respectively.

```
1 SELECT w.last_name, w.hire_date, m.last_name, m.hire_date  
2 FROM hr.employees w, hr.employees m  
3 WHERE w.manager_id = m.employee_id  
4 AND w.hire_date < m.hire_date;
```

## 2 chapter 8

3.Add the first row of data to the MY.EMPLOYEE table from the following sample data. Do not list the columns in the INSERT clause.

```
1  INSERT INTO my_employee
2  VALUES (1, 'Patel', 'Ralph', 'rpatel', 895);
```

4. Populate the MY\_EMPLOYEE table with the second row of sample data from the preceding list. This time, list the columns explicitly in the INSERT clause.

```
1  INSERT INTO my_employee (id, last_name, first_name,
2  userid, salary)
3  VALUES (2, 'Dancs', 'Betty', 'bdancs', 860);
```

5. Confirm your addition to the table

```
1
2  SELECT *
3  FROM my_employee;
```

6. Write an INSERT statement in a text file named loademp.sql to load rows into the MY\_EMPLOYEE table. Concatenate the first letter of the first name and the first seven characters of the last name to produce the user ID

```
1  SET ECHO OFF
2  SET VERIFY OFF
3  INSERT INTO my_employee
4  VALUES (&p_id, '&p_last_name', '&p_first_name',
5  lower(substr('&p_first_name', 1, 1) ||
6  substr('&p_last_name', 1, 7)), &p_salary);
7  SET VERIFY ON
8  SET ECHO ON
```

7. Populate the table with the next two rows of sample data by running the insert statement in the script that you created.

```
1  SET ECHO OFF
2  SET VERIFY OFF
3  INSERT INTO my_employee
4  VALUES (&p_id, '&p_last_name', '&p_first_name',
5  lower(substr('&p_first_name', 1, 1) ||
6  substr('&p_last_name', 1, 7)), &p_salary);
```

```
7 SET VERIFY ON
8 SET ECHO ON
```

7. Populate the table with the next two rows of sample data by running the insert statement in the script that you created.

```
1 SET ECHO OFF
2 SET VERIFY OFF
3 INSERT INTO my_employee
4 VALUES (&p_id, '&p_last_name', '&p_first_name',
5 lower(substr('&p_first_name', 1, 1) ||
6 substr('&p_last_name', 1, 7)), &p_salary);
7 SET VERIFY ON
8 SET ECHO ON
```

8. Confirm your additions to the table.

```
1 SELECT *
2 FROM my_employee;
```

9. Make the data additions permanent.

```
1
2 COMMIT ;
```

10. Change the last name of employee 3 to Drexler.

```
1
2 UPDATE my_employee
3 SET last_name = 'Drexler'
4 WHERE id = 3;
```

11. Change the salary to 1000 for all employees with a salary less than 900

```
1
2 UPDATE my_employee
3 SET salary = 1000
```

```
4 WHERE salary < 900;
```

12. Verify your changes to the table.

```
1  
2 SELECT last_name, salary  
3 FROM my_employee;
```

13.Delete Betty Dancs from the MY\_EMPLOYEE table

```
1  
2 DELETE  
3 FROM my_employee  
4 WHERE last_name = 'Dancs';
```

14.Confirm your changes to the table.

```
1  
2 SELECT *  
3 FROM my_employee;
```