

Database Management System – cs422 DE
Lab 1 – Wk 3 & 4

This Lab is based on lecture 3 & 4 (chapters 6 & 7).

- Submit your *own work* on time. No credit will be given if the lab is submitted after the due date.
 - Note that the completed lab should be submitted in .zip or .rar format only.
 - If you think that your answer needs explanation to get credit then please write it down.
-

Solve the questions from 6.32 to 6.40 in the Case Study 2 on page no. 173 (5th edition).

You are required to run & test all these queries in SQL Server. Note that you'll need to create and populate the tables first.

To get full credit for this lab, you need to submit the following:

- (1) Screenshots for at least 4 of the queries with output.
- (2) Answer SQL queries for all of the mentioned exercises.

For your quick reference, the schema and the questions are given below.

Employee (**empID**, fName, lName, address, DOB, sex, position, deptNo)

Department (**deptNo**, deptName, mgrEmpID)

Project (**projNo**, projName, deptNo)

WorksOn (**empID**, **projNo**, hoursWorked)

where

- *Employee* contains employee details and *empID* is the key.
- *Department* contains department details and *deptNo* is the key. *mgrEmpID* identifies the employee who is the manager of the department. There is only one manager for each department.
- *Project* contains details of the projects in each department and the key is *projNo* (no two departments can run the same project).
- *WorksOn* contains details of the hours worked by employees on each project, and *empID/projNo* form the key.

Exercises

1. List all employees in alphabetical order of surname and within surname, first name.

ANS:

```
SELECT * FROM Employee ORDER BY lName, fName;
```

The screenshot shows a database IDE with a query window containing the following SQL statement:

```
SELECT * FROM Employee ORDER BY lName, fName;
```

The results pane displays 10 rows of employee data:

empID	fName	lName	address	DOB	sex	position	deptNo
1	David	Brown	567 Maple St	1982-09-18	M	Manager	2
2	John	Doe	123 Main St	1980-05-15	M	Manager	1
3	Daniel	Garcia	234 Birch St	1975-12-30	M	Manager	3
4	Michael	Johnson	789 Oak St	1990-03-10	M	Analyst	2
5	Emma	Jones	890 Cedar St	1988-07-05	F	HR Specialist	2
6	Olivia	Martinez	876 Walnut St	1989-04-12	F	Marketing Coordinator	3
7	Sophia	Rodriguez	987 Pinecrest St	1987-02-14	F	Accountant	4
8	Jane	Smith	456 Elm St	1985-08-20	F	Developer	1
9	William	Taylor	543 Oakwood St	1991-06-28	M	Financial Analyst	4
10	Emily	Williams	321 Pine St	1992-11-25	F	Designer	3

2. List all the details of employees who are female.

ANS:

```
SELECT * FROM Employee WHERE sex = 'F';
```

The screenshot shows the same database IDE with the following SQL statement:

```
SELECT * FROM Employee WHERE sex = 'F';
```

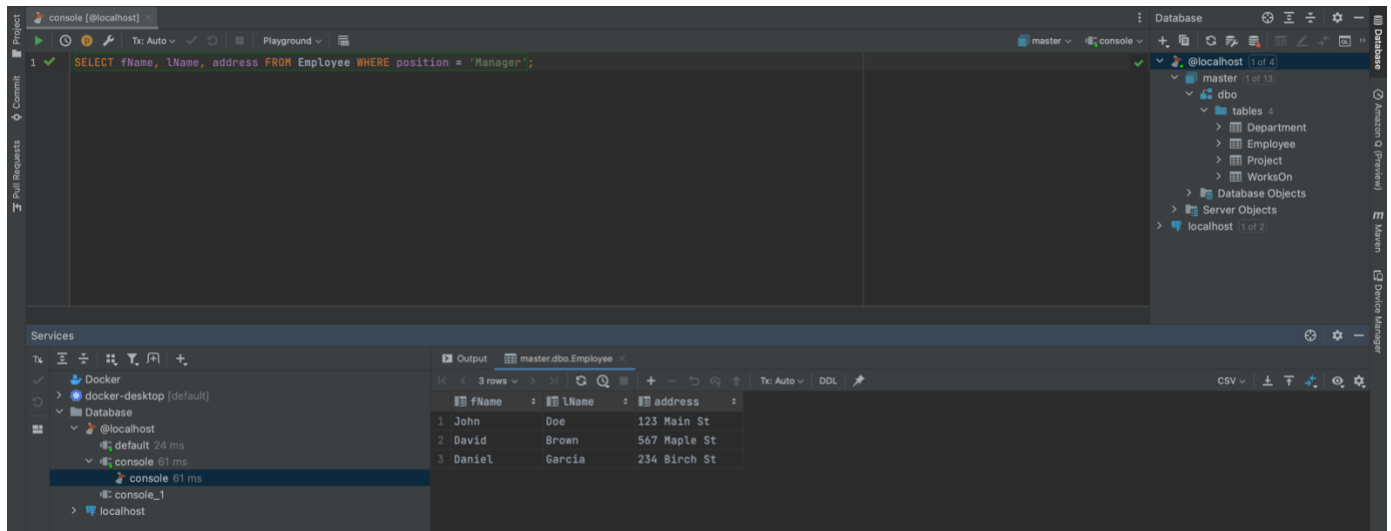
The results pane displays 5 rows of employee data, filtered by sex:

empID	fName	lName	address	DOB	sex	position	deptNo
2	Jane	Smith	456 Elm St	1985-08-20	F	Developer	1
4	Emily	Williams	321 Pine St	1992-11-25	F	Designer	3
6	Emma	Jones	890 Cedar St	1988-07-05	F	HR Specialist	2
4	Olivia	Martinez	876 Walnut St	1989-04-12	F	Marketing Coordinator	3
7	Sophia	Rodriguez	987 Pinecrest St	1987-02-14	F	Accountant	4

3. List the names and addresses of all employees who are Managers.

ANS:

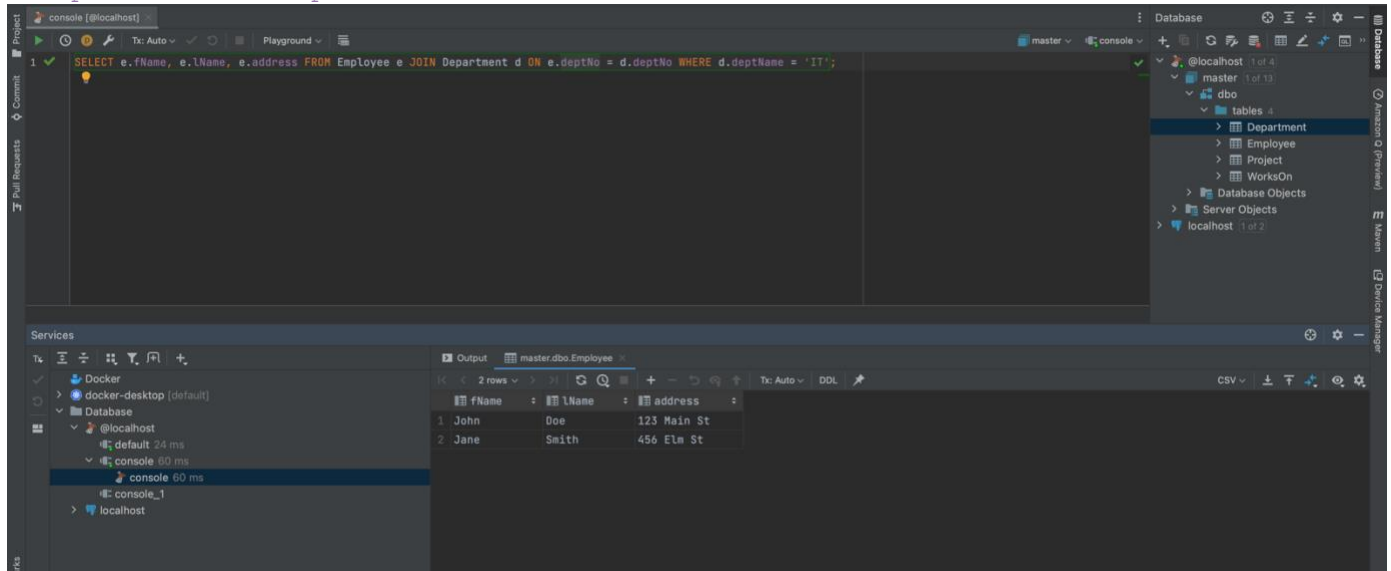
```
SELECT fName, lName, address FROM Employee WHERE position = 'Manager';
```



- Produce a list of the names and addresses of all employees who work for the IT department.

ANS:

```
SELECT e.fName, e.lName, e.address FROM Employee e JOIN Department d ON e.deptNo = d.deptNo WHERE d.deptName = 'IT';
```



- Produce a complete list of all managers who are due to retire this year, in alphabetical order of surname.

ANS:

```
SELECT e.fName, e.lName, e.address
FROM Employee e
      JOIN Department d ON e.deptNo = d.deptNo
WHERE e.position = 'Manager'
      AND YEAR(DATEADD(YEAR, 65, e.DOB)) = YEAR(GETDATE())
ORDER BY e.lName;
```

- Find out how many employees are managed by 'James Adams'.

ANS:

```
SELECT COUNT(*) -1
FROM Employee e INNER JOIN Department d ON e.deptNo = d.deptNo
```

```
WHERE d.mgrEmpID = (SELECT empID FROM Employee WHERE fName = 'James' AND lName = 'Adams');
```

7. Produce a report of the total hours worked by each employee, arranged in order of department number and within department, alphabetically by employee surname.

ANS:

```
SELECT e.fName, e.lName, d.deptNo, SUM(w.hoursWorked) AS TotalHoursWorked
FROM Employee e
      JOIN Department d ON e.deptNo = d.deptNo
      LEFT JOIN WorksOn w ON e.empID = w.empID
GROUP BY e.empID, e.fName, e.lName, d.deptNo
ORDER BY d.deptNo, e.lName;
```

The screenshot shows a database IDE with a SQL query editor on the left and a results pane on the right. The query is the same as the one provided in the previous block. The results pane shows 10 rows of data, sorted by department number and then by employee surname.

fName	lName	deptNo	TotalHoursWorked
John	Doe	1	40
Jane	Smith	1	35
David	Brown	2	40
Michael	Johnson	2	30
Emma	Jones	2	35
Daniel	Garcia	3	30
Olivia	Martinez	3	25
Emily	Williams	3	25
Sophia	Rodriguez	4	15
William	Taylor	4	20

8. For each project on which more than two employees worked, list the project number, project name and the number of employees who work on that project.

ANS:

```
SELECT p.projNo, p.projName, COUNT(*) AS NumberOfEmployees
FROM Project p
      JOIN WorksOn w ON p.projNo = w.projNo
GROUP BY p.projNo, p.projName
HAVING COUNT(*) > 2;
```

9. List the total number of employees in each department for those departments with more than 10 employees. Create an appropriate heading for the columns of the results table.

ANS:

```
SELECT d.deptNo 'Department Id', d.deptName 'Department Name', COUNT(*) 'Total
Employees'
FROM Employee e
      JOIN Department d ON e.deptNo = d.deptNo
GROUP BY d.deptNo, d.deptName
HAVING COUNT(*) > 10;
```