UNIVERSITY OF CHITTAGONG

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SQL Exercises

Database Systems Exercises

Course Information

Course Code: 413

Course Title: Database Systems

Semester: 4th

Submitted By:

Md. Abdullah Al Mamun Emon
ID: 23701028
GitHub: emon4075

Submitted To:

Dr. Rudra Pratap Deb Nath
Associate Professor
Department of Computer Science & Engineering
University of Chittagong
Chittagong-4331

May 31, 2025

Table of Contents

Contents

1 Chapter 05 - Reporting Aggregated Data Using the Group Functions

Determine the validity of the following three statements. Circle either True or False.

Question 1

Group functions work across many rows to produce one result per group.
True/False

Answer 1

True

Question 2

Group functions include nulls in calculations.
True/False

Answer 2

False

Question 3

The WHERE clause restricts rows before inclusion in a group calculation.
True/False

Answer 3

True

Question 4

The HR department needs the following reports:
Find the highest, lowest, sum, and average salary of all employees. Label the columns as `Maximum`, `Minimum`, `Sum`, and `Average`, respectively. Round your results to the nearest whole number. Save your SQL statement as `lab_05_04.sql`. Run the query.

	Maximum	Minimum	Sum	Average
1	24000	2500	175500	8775

Answer 4

Command:

```
1 SELECT MAX(SALARY) AS "Maximum", MIN(SALARY) AS "Minimum", ROUND(SUM(SALARY)) AS "Sum",
      ROUND(AVG(SALARY)) AS "Average"
2 FROM HR.EMPLOYEES;
```

Question 5

Modify the query in lab_05_04.sql to display the minimum, maximum, sum, and average salary for each job type. Resave lab_05_04.sql as lab_05_05.sql. Run the statement in lab_05_05.sql.

	JOB_ID	Maximum	Minimum	Sum	Average
1	IT_PROG	9000	4200	19200	6400
2	AC_MGR	12000	12000	12000	12000
3	AC_ACCOUNT	8300	8300	8300	8300
4	ST_MAN	5800	5800	5800	5800
5	AD_ASST	4400	4400	4400	4400
6	AD_VP	17000	17000	34000	17000
7	SA_MAN	10500	10500	10500	10500
8	MK_MAN	13000	13000	13000	13000
9	AD_PRES	24000	24000	24000	24000
10	SA_REP	11000	7000	26600	8867
11	MK_REP	6000	6000	6000	6000
12	ST_CLERK	3500	2500	11700	2925

Answer 5

Command:

```
1 SELECT JOB_ID, MAX(SALARY) AS "Maximum", MIN(SALARY) AS "Minimum", ROUND(SUM(SALARY)) AS
      "Sum", ROUND(AVG(SALARY)) AS "Average"
2 FROM HR.EMPLOYEES
3 GROUP BY JOB_ID;
```

Question 6

Write a query to display the number of people with the same job.

	JOB_ID	COUNT(*)
1	AC_ACCOUNT	1
2	AC_MGR	1
3	AD_ASST	1
4	AD_PRES	1
5	AD_VP	2
6	IT_PROG	3
7	MK_MAN	1
8	MK_REP	1
9	SA_MAN	1
10	SA_REP	3
11	ST_CLERK	4
12	ST_MAN	1

Generalize the query so that the user in the HR department is prompted for a job title. Save the script to a file named lab_05_06.sql. Run the query. Enter IT_PROG when prompted.

	JOB_ID	COUNT(*)
1	IT_PROG	3

Answer 6

Command:

```sql
SELECT JOB_ID, COUNT(*)
FROM HR.EMPLOYEES
GROUP BY JOB_ID;

SELECT JOB_ID, COUNT(*)
FROM HR.EMPLOYEES
WHERE JOB_ID = '&USER_PROMPT'
GROUP BY JOB_ID;
```

Question 7

Determine the number of managers without listing them. Label the column as Number of Managers. *Hint: Use the MANAGER_ID column to determine the number of managers.*

	Number of Managers
1	8

Answer 7

Command:

```sql
SELECT COUNT(DISTINCT MANAGER_ID) AS "Number of Managers"
FROM HR.EMPLOYEES;
```

Question 8

Find the difference between the highest and lowest salaries. Label the column `DIFFERENCE`.

	DIFFERENCE
1	21500

Answer 8

Command:

```
SELECT MAX(SALARY)-MIN(SALARY) AS "DIFFERENCE"
FROM HR.EMPLOYEES;
```

Question 9

Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is \$6,000 or less. Sort the output in descending order of salary.

	MANAGER_ID	MIN(SALARY)
1	102	9000
2	205	8300
3	149	7000

Answer 9

Command:

```
SELECT MANAGER_ID, MIN(SALARY)
FROM HR.EMPLOYEES
WHERE MANAGER_ID IS NOT null
GROUP BY MANAGER_ID
HAVING MIN(SALARY) > 6000
ORDER BY MIN(SALARY) DESC;
```

Question 10

If you want an extra challenge, complete the following exercises:
Create a query to display the total number of employees and, of that total, the number of employees hired in 1995, 1996, 1997, and 1998. Create appropriate column headings.

	TOTAL	1995	1996	1997	1998
1	20	1	2	2	3

Answer 10

Command:

```
SELECT COUNT(*) AS "TOTAL",
    SUM(CASE WHEN TO_CHAR(HIRE_DATE, 'YYYY') = '1995' THEN 1 ELSE 0 END) AS "1995",
    SUM(CASE WHEN TO_CHAR(HIRE_DATE, 'YYYY') = '1996' THEN 1 ELSE 0 END) AS "1996",
    SUM(CASE WHEN TO_CHAR(HIRE_DATE, 'YYYY') = '1997' THEN 1 ELSE 0 END) AS "1997",
    SUM(CASE WHEN TO_CHAR(HIRE_DATE, 'YYYY') = '1998' THEN 1 ELSE 0 END) AS "1998"
FROM HR.EMPLOYEES;
```

Question 11

Create a matrix query to display the job, the salary for that job based on department number, and the total salary for that job, for departments 20, 50, 80, and 90, giving each column an appropriate heading.

Answer 11

Command:

```
SELECT
    JOB_ID AS "Job",
    SUM(CASE WHEN department_id = 20 THEN salary END) AS "Dept 20",
    SUM(CASE WHEN department_id = 50 THEN salary END) AS "Dept 50",
    SUM(CASE WHEN department_id = 80 THEN salary END) AS "Dept 80",
    SUM(CASE WHEN department_id = 90 THEN salary END) AS "Dept 90",
    SUM(salary) AS "Total"
FROM HR.EMPLOYEES
GROUP BY JOB_ID
ORDER BY JOB_ID;
```

2   Chapter 06 - Displaying Data From Multiple Tables

Practice 6

Question 1

Write a query for the HR department to produce the addresses of all the departments. Use the `LOCATIONS` and `COUNTRIES` tables. Show the location ID, street address, city, state or province, and country name in the output. Use a `NATURAL JOIN` to produce the results.

Answer 1

Command:

```
1 SELECT LOCATION_ID, STREET_ADDRESS, CITY, STATE_PROVINCE, COUNTRY_NAME
2 FROM HR.LOCATIONS
3 NATURAL JOIN HR.COUNTRIES;
```

Question 2

The HR department needs a report of all employees. Write a query to display the last name, department number, and department name for all employees.

Answer 2

Command:

```
1 SELECT LAST_NAME, DEPARTMENT_ID, DEPARTMENT_NAME
2 FROM HR.EMPLOYEES JOIN HR.DEPARTMENTS
3 USING (DEPARTMENT_ID);
```

Question 3

The HR department needs a report of employees in Toronto. Display the last name, job, department number, and the department name for all employees who work in Toronto.

	LAST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
1	Hartstein	MK_MAN	20	Marketing
2	Fay	MK_REP	20	Marketing

Answer 3

Command:

```
SELECT LAST_NAME, JOB_ID, DEPARTMENT_ID, DEPARTMENT_NAME
FROM HR.EMPLOYEES JOIN HR.DEPARTMENTS
USING (DEPARTMENT_ID)
JOIN HR.LOCATIONS
USING (LOCATION_ID)
WHERE CITY = 'Toronto';
```

Question 4

Create a report to display employees' last name and employee number along with their manager's last name and manager number. Label the columns Employee, Emp#, Manager, and Mgr#, respectively. Save your SQL statement as lab_06_04.sql. Run the query.

Answer 4

Command:

```
SELECT e.LAST_NAME AS Employee, e.EMPLOYEE_ID AS "Emp#", m.LAST_NAME AS Manager,
    m.EMPLOYEE_ID AS "Mgr#"
FROM HR.EMPLOYEES e JOIN HR.EMPLOYEES m
ON (e.MANAGER_ID = m.EMPLOYEE_ID);
```

Question 5

Modify lab_06_04.sql to display all employees including King, who has no manager. Order the results by the employee number. Save your SQL statement as lab_06_05.sql. Run the query in lab_06_05.sql.

	Employee	Emp#	Manager	Mgr#
1	King	100	(null)	(null)
2	Kochhar	101	King	100
3	De Haan	102	King	100
4	Harold	103	De Haan	102
5	Ernst	104	Harold	103
6	Lorentz	107	Harold	103
7	Mourgos	124	King	100
...	...	...	...	...
19	Fay	202	Hartstein	201
20	Gietz	206	Higgins	205

Answer 5

Command:

```
SELECT e.LAST_NAME AS Employee, e.EMPLOYEE_ID AS "Emp#", m.LAST_NAME AS Manager,
    m.EMPLOYEE_ID AS "Mgr#"
FROM HR.EMPLOYEES e LEFT JOIN HR.EMPLOYEES m
ON (e.MANAGER_ID = m.EMPLOYEE_ID)
ORDER BY e.EMPLOYEE_ID;
```

Question 6

Create a report for the HR department 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. Save the script to a file named lab_06_06.sql.

Answer 6

Command:

```
SELECT e.DEPARTMENT_ID AS DEPARTMENT, e.LAST_NAME AS EMPLOYEE, m.LAST_NAME AS COLLEAGUE
FROM HR.EMPLOYEES e JOIN HR.EMPLOYEES m
ON (e.DEPARTMENT_ID = m.DEPARTMENT_ID)
WHERE e.EMPLOYEE_ID <> m.EMPLOYEE_ID
ORDER BY e.DEPARTMENT_ID, e.LAST_NAME, m.LAST_NAME;
```

Question 7

The HR department needs a report on job grades and salaries. To familiarize yourself with the JOB_GRADES table, first show the structure of the JOB_GRADES table. Then create a query that displays the name, job, department name, salary, and grade for all employees.

Answer 7

Command:

```
SELECT e.LAST_NAME, e.JOB_ID, d.DEPARTMENT_NAME, e.SALARY, g.GRADE_LEVEL
FROM HR.EMPLOYEES e
JOIN HR.DEPARTMENTS d ON e.DEPARTMENT_ID = d.DEPARTMENT_ID
JOIN HR.JOB_GRADES g ON e.SALARY BETWEEN g.LOWEST_SAL AND g.HIGHEST_SAL
ORDER BY g.GRADE_LEVEL, e.SALARY;
```

Question 8

The HR department wants to determine the names of all the employees who were hired after Davies. Create a query to display the name and hire date of any employee hired after employee Davies.

	LAST_NAME	HIRE_DATE
1	Lorentz	07-FEB-99
2	Mourgos	16-NOV-99
3	Matos	15-MAR-98
4	Vargas	09-JUL-98
5	Zlotkey	29-JAN-00
6	Taylor	24-MAR-98
7	Grant	24-MAY-99
8	Fay	17-AUG-97

Answer 8

Command:

```sql
SELECT LAST_NAME, HIRE_DATE
FROM HR.EMPLOYEES
WHERE HIRE_DATE > (
    SELECT HIRE_DATE
    FROM HR.EMPLOYEES
    WHERE LAST_NAME = 'Davies'
);
```

Question 9

The HR department needs to find the names and hire dates of all the employees who were hired before their managers, along with their managers' names and hire dates. Save the script to a file named lab_06_09.sql.

	LAST_NAME	HIRE_DATE	LAST_NAME_1	HIRE_DATE_1
1	Whalen	17-SEP-87	Kochhar	21-SEP-89
2	Hunold	03-JAN-90	De Haan	13-JAN-93
3	Vargas	09-JUL-98	Mourgos	16-NOV-99
4	Matos	15-MAR-98	Mourgos	16-NOV-99
5	Davies	29-JAN-97	Mourgos	16-NOV-99
6	Rajs	17-OCT-95	Mourgos	16-NOV-99
7	Grant	24-MAY-99	Zlotkey	29-JAN-00
8	Taylor	24-MAR-98	Zlotkey	29-JAN-00
9	Abel	11-MAY-96	Zlotkey	29-JAN-00

Answer 9

Command:

```
1 SELECT e.LAST_NAME, e.HIRE_DATE, m.LAST_NAME AS LAST_NAME1, m.HIRE_DATE AS HIRE_DATE1
2 FROM HR.EMPLOYEES e JOIN HR.EMPLOYEES m
3 ON (e.MANAGER_ID = m.EMPLOYEE_ID)
4 WHERE e.HIRE_DATE < m.HIRE_DATE;
```

3   Chapter 07 - Using Subqueries To Solve Queries

Question 1

The HR department needs a query that prompts the user for an employee last name. The query then displays the last name and hire date of any employee in the same department as the employee whose name they supply (excluding that employee). For example, if the user enters Zlotkey, find all employees who work with Zlotkey (excluding Zlotkey).

	LAST_NAME	HIRE_DATE
1	Abel	11-MAY-96
2	Taylor	24-MAR-98

Answer 1

Command:

```sql
SELECT LAST_NAME, HIRE_DATE
FROM HR.EMPLOYEES
WHERE DEPARTMENT_ID = (
    SELECT DEPARTMENT_ID
    FROM HR.EMPLOYEES
    WHERE LAST_NAME = '&user_prompt'
)
AND LAST_NAME <> '&user_prompt';
```

Question 2

Create a report that displays the employee number, last name, and salary of all employees who earn more than the average salary. Sort the results in order of ascending salary.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	103	Hunold	9000
2	149	Zlotkey	10500
3	174	Abel	11000
4	205	Higgins	12000
5	201	Hartstein	13000
6	101	Kochhar	17000
7	102	De Haan	17000
8	100	King	24000

Answer 2

Command:

```
1  SELECT EMPLOYEE_ID, LAST_NAME, SALARY
2  FROM HR.EMPLOYEES
3  WHERE SALARY > (
4      SELECT AVG(SALARY)
5      FROM HR.EMPLOYEES
6  )
7  ORDER BY SALARY;
```

Question 3

Write a query that displays the employee number and last name of all employees who work in a department with any employee whose last name contains the letter "u". Save your SQL statement as lab_07_03.sql. Run your query.

	EMPLOYEE_ID	LAST_NAME
1	124	Mourgos
2	141	Rajs
3	142	Davies
4	143	Matos
5	144	Vargas
6	103	Hunold
7	104	Ernst
8	107	Lorentz

Answer 3

Command:

```
1  SELECT E.EMPLOYEE_ID, E.LAST_NAME
2  FROM HR.EMPLOYEES E
3  WHERE E.DEPARTMENT_ID IN (
4      SELECT DISTINCT E2.DEPARTMENT_ID
5      FROM HR.EMPLOYEES E2
6      WHERE UPPER(E2.LAST_NAME) LIKE '%U%'
7  );
```

Question 4

The HR department needs a report that displays the last name, department number, and job ID of all employees whose department location ID is 1700.

	LAST_NAME	**DEPARTMENT_ID**	**JOB_ID**
1	Whalen	10	AD_ASST
2	King	90	AD_PRES
3	Kochhar	90	AD_VP
4	De Haan	90	AD_VP
5	Higgins	110	AC_MGR
6	Gietz	110	AC_ACCOUNT

Modify the query so that the user is prompted for a location ID. Save this to a file named `lab_07_04.sql`.

Answer 4

Command:

```sql
SELECT LAST_NAME, DEPARTMENT_ID, JOB_ID
FROM HR.EMPLOYEES JOIN HR.DEPARTMENTS
USING (DEPARTMENT_ID)
WHERE DEPARTMENT_ID IN (
    SELECT DEPARTMENT_ID
    FROM HR.DEPARTMENTS
    WHERE LOCATION_ID = 1700
);

-- Extended:
SELECT LAST_NAME, DEPARTMENT_ID, JOB_ID
FROM HR.EMPLOYEES JOIN HR.DEPARTMENTS
USING (DEPARTMENT_ID)
WHERE DEPARTMENT_ID IN (
    SELECT DEPARTMENT_ID
    FROM HR.DEPARTMENTS
    WHERE LOCATION_ID = '&user_prompt'
);
```

Question 5

Create a report for HR that displays the last name and salary of every employee who reports to King.

	LAST_NAME	**SALARY**
1	Kochhar	17000
2	De Haan	17000
3	Mourgos	5800
4	Zlotkey	10500
5	Hartstein	13000

Answer 5

Command:

```sql
SELECT LAST_NAME, SALARY
FROM HR.EMPLOYEES
WHERE MANAGER_ID IN (
    SELECT EMPLOYEE_ID
    FROM HR.EMPLOYEES
    WHERE LAST_NAME = 'King'
);
```

Question 6

Create a report for HR that displays the department number, last name, and job ID for every employee in the Executive department.

	DEPARTMENT_ID	LAST_NAME	JOB_ID
1	90	King	AD_PRES
2	90	Kochhar	AD_VP
3	90	De Haan	AD_VP

Answer 6

Command:

```sql
SELECT DEPARTMENT_ID, LAST_NAME, JOB_ID
FROM HR.DEPARTMENTS JOIN HR.EMPLOYEES
USING (DEPARTMENT_ID)
WHERE DEPARTMENT_ID = (
    SELECT DEPARTMENT_ID
    FROM HR.DEPARTMENTS
    WHERE DEPARTMENT_NAME = 'Executive'
);
```

Question 7

If you have the time, complete the following exercise:
Modify the query in lab_07_03.sql to display the employee number, last name, and salary of all employees who earn more than the average salary, and who work in a department with any employee whose last name contains a "u". Resave lab_07_03.sql as lab_07_07.sql. Run the statement in lab_07_07.sql.

	EMPLOYEE_ID	LAST_NAME	SALARY
1	103	Hunold	9000

Answer 7

Command:

```
SELECT E.EMPLOYEE_ID, E.LAST_NAME, SALARY
FROM HR.EMPLOYEES E
WHERE E.DEPARTMENT_ID IN (
    SELECT DISTINCT E2.DEPARTMENT_ID
    FROM HR.EMPLOYEES E2
    WHERE UPPER(E2.LAST_NAME) LIKE '%U%'
) AND SALARY > (
    SELECT AVG(SALARY)
    FROM HR.EMPLOYEES
);
```