

# University of Chittagong

Department of Computer Science & Engineering
Database Systems Lab

Name of the assignment:

# Chapters 5-7 Practice Problems

CSE 414

Assignment 02

Submitted By:

Debashish Chakraborty

ID: 23701034

Submitted To:

Dr. Rudra Pratap Deb

Nath

Associate Professor

# Contents

1	Chapter 5 Problems & Solutions	<b>2</b>
	1.1 Theoretical Questions	
2	Chapter 6 Problems & Solutions	6
3	Chapter 7 Problems & Solutions	11

# 1 Chapter 5 Problems & Solutions

# 1.1 Theoretical Questions

**Problem 1.** Group functions work across many rows to produce one result per group.

Solution: TRUE

**Problem 2.** Group functions include nulls in calculations.

Solution: FALSE

**Problem 3.** The WHERE clause restricts rows before inclusion in a group

calculation.

Solution: TRUE

## 1.2 Practical Problems

**Problem 4.** 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.



#### **Solution:**

```
SELECT

ROUND (MAX (SALARY)) "Maximum",

ROUND (MIN (SALARY)) "Minimum",

ROUND (SUM (SALARY)) "Sum",

ROUND (AVG (SALARY)) "Average"

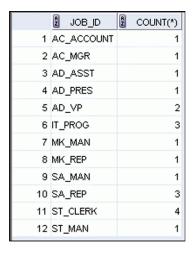
FROM EMPLOYEES;
```

**Problem 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	2 Maximum	Minimum	2 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

```
JOB_ID,
ROUND(MAX(SALARY)) "Maximum",
ROUND(MIN(SALARY)) "Minimum",
ROUND(SUM(SALARY)) "Sum",
ROUND(AVG(SALARY)) "Average"
FROM EMPLOYEES
GROUP BY JOB_ID;
```

**Problem 6.** Write a query to display the number of people with the same job.



#### **Solution:**

```
SELECT

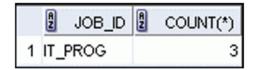
JOB_ID,

COUNT(*)

FROM EMPLOYEES

GROUP BY JOB_ID;
```

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.



```
SELECT

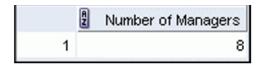
JOB_ID,

COUNT(*)

FROM EMPLOYEES

WHERE JOB_ID = UPPER('&job_id')
GROUP BY JOB_ID;
```

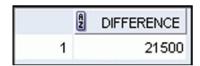
**Problem 7.** Determine the number of managers without listing them. Label the column as Number of Managers. Hint: Use the MANAGER\_ID column to determine the number of managers.



#### Solution:

```
SELECT COUNT(DISTINCT MANAGER_ID) "Number of Managers"
FROM EMPLOYEES;
```

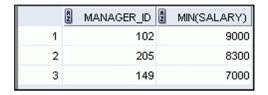
**Problem 8.** Find the difference between the highest and lowest salaries. Label the column DIFFERENCE.



#### Solution:

```
SELECT MAX(SALARY) - MIN(SALARY) DIFFERENCE
FROM EMPLOYEES;
```

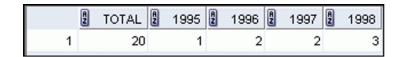
**Problem 9.** Create a report to display the manager number and the salary of the lowest-paid employee for that manager. Exclude anyone whose manager is not known. Exclude any groups where the minimum salary is \$6,000 or less. Sort the output in descending order of salary.



```
MANAGER_ID,
MIN(SALARY)

FROM EMPLOYEES
WHERE MANAGER_ID IS NOT NULL
GROUP BY MANAGER_ID
HAVING MIN(SALARY) > 6000
ORDER BY MIN(SALARY) DESC;
```

**Problem 10.** 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.



### **Solution:**

```
SELECT COUNT(*) TOTAL,

COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1995, '1')) "1995",

COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1996, '1')) "1996",

COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1997, '1')) "1997",

COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1998, '1')) "1998"

FROM EMPLOYEES;
```

**Problem 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.

	g Job	2 Dept 20	2 Dept 50	2 Dept 80	2 Dept 90	2 Total
1	IT_PROG	(null)	(null)	(null)	(null)	19200
2	AC_MGR	(null)	(null)	(null)	(null)	12000
3	AC_ACCOUNT	(null)	(null)	(null)	(null)	8300
4	ST_MAN	(null)	5800	(null)	(null)	5800
5	AD_ASST	(null)	(null)	(null)	(null)	4400
6	AD_VP	(null)	(null)	(null)	34000	34000
7	SA_MAN	(null)	(null)	10500	(null)	10500
8	MK_MAN	13000	(null)	(null)	(null)	13000
9	AD_PRES	(null)	(null)	(null)	24000	24000
10	SA_REP	(null)	(null)	19600	(null)	26600
11	MK_REP	6000	(null)	(null)	(null)	6000
12	ST_CLERK	(null)	11700	(null)	(null)	11700

```
SELECT DISTINCT JOB_ID "Job",

SUM(DECODE(DEPARTMENT_ID, 20, SALARY)) "Dept 20",

SUM(DECODE(DEPARTMENT_ID, 50, SALARY)) "Dept 50",

SUM(DECODE(DEPARTMENT_ID, 80, SALARY)) "Dept 80",

SUM(DECODE(DEPARTMENT_ID, 90, SALARY)) "Dept 90",

SUM(SALARY) TOTAL

FROM EMPLOYEES

GROUP BY JOB_ID;
```

# 2 Chapter 6 Problems & Solutions

**Problem 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 in the output. Use a NATURAL JOIN to produce the results.

	A	LOCATION_ID	A	STREET_ADDRESS	A	CITY	A	STATE_PROVINCE	A	COUNTRY_NAME
1		1400	201	4 Jabberwocky Rd	Sou	ıthlake	Tex	as	Unit	ted States of America
2		1500	2011	1 Interiors Blvd	Sou	ıth San Francisco	Cali	fornia	Unit	ted States of America
3		1700	2004	4 Charade Rd	Sea	attle	Wa	shington	Unit	ted States of America
4		1800	460	Bloor St. W.	Tor	onto	Ont	ario	Car	nada 🗦
5		2500	Mag	dalen Centre, The	Oxt	ford	Oxf	ord	Unit	ted Kingdom

#### **Solution:**

```
SELECT

LOCATION_ID,

STREET_ADDRESS,

CITY,

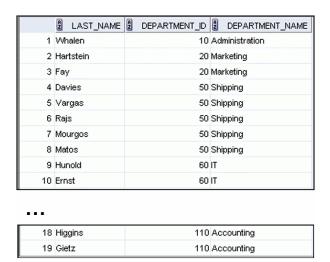
STATE_PROVINCE,

COUNTRY_NAME

FROM LOCATIONS

NATURAL JOIN COUNTRIES;
```

**Problem 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 the employees.



```
SELECT

LAST_NAME,

DEPARTMENT_ID,

DEPARTMENT_NAME

FROM EMPLOYEES

JOIN DEPARTMENTS USING (DEPARTMENT_ID);
```

**Problem 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.

	A	LAST_NAME	A	JOB_ID	A	DEPARTMENT_ID	A	DEPARTMENT_NAME
1	Hari	tstein	MK.	_MAN		20	Mar	keting
2	Fay		MK,	_REP		20	Mar	keting

#### **Solution:**

```
LAST_NAME,

JOB_ID,

DEPARTMENT_ID,

DEPARTMENT_NAME

FROM EMPLOYEES

JOIN DEPARTMENTS USING (DEPARTMENT_ID)

JOIN LOCATIONS USING (LOCATION_ID)

WHERE CITY = 'Toronto';
```

**Problem 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. **Solution:** 

	Employee	2 EMP#	Manager	2 Mgr#
1	Kochhar	101	King	100
2	De Haan	102	King	100
3	Hunold	103	De Haan	102
4	Ernst	104	Hunold	103
5	Lorentz	107	Hunold	103
6	Mourgos	124	King	100
7	Rajs	141	Mourgos	124
8	Davies	142	Mourgos	124
9	Matos	143	Mourgos	124
10	Vargas	144	Mourgos	124
•••				
15	Whalen	200	Kochhar	101
16	Hartstein	201	King	100
17	Fay	202	Hartstein	201
18	Higgins	205	Kochhar	101
19	Gietz	206	Higgins	205

```
1 SELECT
2     E.LAST_NAME "Employee",
3     E.EMPLOYEE_ID "Emp#",
4     M.LAST_NAME "Manager",
5     M.EMPLOYEE_ID "Mgr#"
6 FROM EMPLOYEES E
7 JOIN EMPLOYEES M ON (E.MANAGER_ID = M.EMPLOYEE_ID)
8 ORDER BY E.EMPLOYEE_ID;
```

205

**Problem 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	2 EMP#	Manager	2 Mgr#		
1	King	100	(null)	(null)		
2	Kochhar	101	King	100		
3	De Haan	102	King	100		
4	Hunold	103	De Haan	102		
5	Ernst	104	Hunold	103		
6	Lorentz	107	Hunold	103		
7	Mourgos	124	King	100		
8	Rajs	141	Mourgos	124		
9	Davies	142	Mourgos	124		
10	Matos	143	Mourgos	124		
•	•••					
18	Fay	202	Hartstein	201		
19	Higgins	205	Kochhar	101		

### **Solution:**

```
SELECT

E.LAST_NAME "Employee",

E.EMPLOYEE_ID "Emp#",

M.LAST_NAME "Manager",

M.EMPLOYEE_ID "Mgr#"

FROM EMPLOYEES E

LEFT JOIN EMPLOYEES M ON (E.MANAGER_ID = M.EMPLOYEE_ID)

ORDER BY E.EMPLOYEE_ID;
```

206 Higgins

20 Gietz

**Problem 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.

P	DEPARTMENT	2 EMPLOYEE	COLLEAGUE
1	20	Fay	Hartstein
2	20	Hartstein	Fay
3	50	Davies	Matos
4	50	Davies	Mourgos
5	50	Davies	Rajs
6	50	Davies	Vargas
7	50	Matos	Davies
8	50	Matos	Mourgos
9	50	Matos	Rajs
10	50	Matos	Vargas
42	110	Higgins	Gietz

```
SELECT

E.DEPARTMENT_ID DEPARTMENT,

E.LAST_NAME EMPLOYEE,

C.LAST_NAME COLLEAGUE

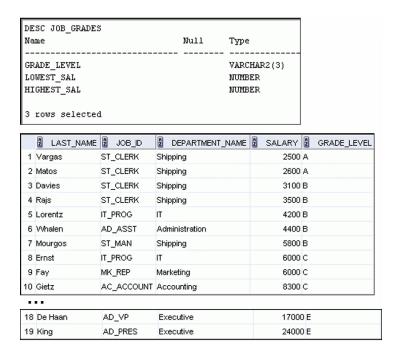
FROM EMPLOYEES E

JOIN EMPLOYEES C ON (E.DEPARTMENT_ID = C.DEPARTMENT_ID)

WHERE E.EMPLOYEE_ID <> C.EMPLOYEE_ID

ORDER BY DEPARTMENT, EMPLOYEE, COLLEAGUE;
```

**Problem 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.



```
SELECT

LAST_NAME,

JOB_ID,

DEPARTMENT_NAME,

SALARY,

GRADE_LEVEL

FROM EMPLOYEES

JOIN DEPARTMENTS USING (DEPARTMENT_ID)

JOIN JOB_GRADES ON (SALARY BETWEEN LOWEST_SAL AND HIGHEST_SAL)

ORDER BY SALARY;
```

**Problem 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

#### **Solution:**

```
SELECT

E.LAST_NAME,

TO_CHAR(E.HIRE_DATE, 'DD-MON-YY') HIRE_DATE

FROM EMPLOYEES E

JOIN EMPLOYEES DAVIES ON (DAVIES.LAST_NAME = 'Davies')

WHERE DAVIES.HIRE_DATE < E.HIRE_DATE;
```

**Problem 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

```
SELECT

E.LAST_NAME,

TO_CHAR(E.HIRE_DATE, 'DD-MON-YY') HIRE_DATE,

M.LAST_NAME,

TO_CHAR(M.HIRE_DATE, 'DD-MON-YY') HIRE_DATE1

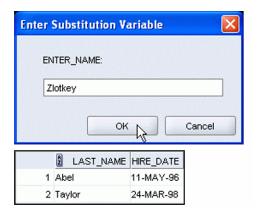
FROM EMPLOYEES E

JOIN EMPLOYEES M ON (E.MANAGER_ID = M.EMPLOYEE_ID)

WHERE E.HIRE_DATE < M.HIRE_DATE;
```

# 3 Chapter 7 Problems & Solutions

**Problem 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).



#### **Solution:**

```
SELECT
LAST_NAME,
TO_CHAR(HIRE_DATE, 'DD-MON-YY') HIRE_DATE
FROM EMPLOYEES
WHERE DEPARTMENT_ID = (
SELECT DEPARTMENT_ID
FROM EMPLOYEES
WHERE LAST_NAME = INITCAP('&&last_name')
)
AND LAST_NAME <> INITCAP('&&last_name');
```

**Problem 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	2 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

```
SELECT
EMPLOYEE_ID,
LAST_NAME,
SALARY
FROM EMPLOYEES
WHERE SALARY > (
SELECT AVG(SALARY)
FROM EMPLOYEES
)
ORDER BY SALARY;
```

**Problem 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

```
SELECT
EMPLOYEE_ID,
LAST_NAME
FROM EMPLOYEES
WHERE DEPARTMENT_ID IN (
SELECT DEPARTMENT_ID
FROM EMPLOYEES
WHERE LAST_NAME LIKE '%u%'
);
```

**Problem 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	A	DEPARTMENT_ID	A	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

#### Solution:

```
LAST_NAME,
DEPARTMENT_ID,
JOB_ID
FROM EMPLOYEES
WHERE DEPARTMENT_ID IN (
SELECT DEPARTMENT_ID
FROM DEPARTMENTS
WHERE LOCATION_ID = 1700

ORDER BY DEPARTMENT_ID;
```

Modify the query so that the user is prompted for a location ID. Save this to a file named lab\_07\_04.sql.

```
LAST_NAME,
DEPARTMENT_ID,
JOB_ID
FROM EMPLOYEES
WHERE DEPARTMENT_ID IN (
SELECT DEPARTMENT_ID
FROM DEPARTMENTS
WHERE LOCATION_ID = &LOCATION_ID

ORDER BY DEPARTMENT_ID;
```

**Problem 5.** Create a report for HR that displays the last name and salary of every employee who reports to King.

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

```
LAST_NAME,

SALARY

FROM EMPLOYEES

WHERE MANAGER_ID = (

SELECT EMPLOYEE_ID

FROM EMPLOYEES

WHERE LAST_NAME = 'King'

);
```

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

#### **Solution:**

```
DEPARTMENT_ID,
LAST_NAME,
JOB_ID
FROM EMPLOYEES
WHERE DEPARTMENT_ID = (
SELECT DEPARTMENT_ID
FROM DEPARTMENTS
WHERE DEPARTMENTS
WHERE DEPARTMENT_NAME = 'Executive'
);
```

**Problem 7.** 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.

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

```
1 SELECT
      EMPLOYEE_ID,
      LAST_NAME,
      SALARY
5 FROM EMPLOYEES
6 WHERE SALARY > (
      SELECT AVG (SALARY)
      FROM EMPLOYEES
9)
10 AND DEPARTMENT_ID IN (
      SELECT DEPARTMENT_ID
      FROM EMPLOYEES
12
      WHERE LAST_NAME LIKE '%u%'
13
14 );
```