



# 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

May 31, 2025

## Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Chapter 5 Problems &amp; Solutions</b> | <b>2</b>  |
| 1.1      | Theoretical Questions . . . . .           | 2         |
| 1.2      | Practical Problems . . . . .              | 2         |
| <b>2</b> | <b>Chapter 6 Problems &amp; Solutions</b> | <b>6</b>  |
| <b>3</b> | <b>Chapter 7 Problems &amp; Solutions</b> | <b>11</b> |

# 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.

|   | Maximum | Minimum | Sum    | Average |
|---|---------|---------|--------|---------|
| 1 | 24000   | 2500    | 175500 | 8775    |

**Solution:**

```

1 SELECT
2     ROUND(MAX(SALARY)) "Maximum",
3     ROUND(MIN(SALARY)) "Minimum",
4     ROUND(SUM(SALARY)) "Sum",
5     ROUND(AVG(SALARY)) "Average"
6 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       | 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    |

**Solution:**

```

1 SELECT
2     JOB_ID ,
3     ROUND(MAX(SALARY)) "Maximum",
4     ROUND(MIN(SALARY)) "Minimum",
5     ROUND(SUM(SALARY)) "Sum",
6     ROUND(AVG(SALARY)) "Average"
7 FROM EMPLOYEES
8 GROUP BY JOB_ID;

```

**Problem 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        |

**Solution:**

```

1 SELECT
2     JOB_ID ,
3     COUNT(*)
4 FROM EMPLOYEES
5 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.

|   | JOB_ID  | COUNT(*) |
|---|---------|----------|
| 1 | IT_PROG | 3        |

```

1 SELECT
2     JOB_ID ,
3     COUNT(*)
4 FROM EMPLOYEES
5 WHERE JOB_ID = UPPER('&job_id')
6 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.

|   | Number of Managers |
|---|--------------------|
| 1 | 8                  |

**Solution:**

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

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

|   | DIFFERENCE |
|---|------------|
| 1 | 21500      |

**Solution:**

```
1 SELECT MAX(SALARY) - MIN(SALARY) DIFFERENCE
2 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) |
|---|------------|-------------|
| 1 | 102        | 9000        |
| 2 | 205        | 8300        |
| 3 | 149        | 7000        |

**Solution:**

```
1 SELECT
2     MANAGER_ID ,
3     MIN(SALARY)
4 FROM EMPLOYEES
5 WHERE MANAGER_ID IS NOT NULL
6 GROUP BY MANAGER_ID
7 HAVING MIN(SALARY) > 6000
8 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.

|   | TOTAL | 1995 | 1996 | 1997 | 1998 |
|---|-------|------|------|------|------|
| 1 | 20    | 1    | 2    | 2    | 3    |

**Solution:**

```

1 SELECT COUNT(*) TOTAL,
2     COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1995, '1')) "1995",
3     COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1996, '1')) "1996",
4     COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1997, '1')) "1997",
5     COUNT(DECODE(TO_CHAR(HIRE_DATE, 'YYYY'), 1998, '1')) "1998"
6 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.

|    | Job        | Dept 20 | Dept 50 | Dept 80 | Dept 90 | 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 |

**Solution:**

```

1 SELECT DISTINCT JOB_ID "Job",
2     SUM(DECODE(DEPARTMENT_ID, 20, SALARY)) "Dept 20",
3     SUM(DECODE(DEPARTMENT_ID, 50, SALARY)) "Dept 50",
4     SUM(DECODE(DEPARTMENT_ID, 80, SALARY)) "Dept 80",
5     SUM(DECODE(DEPARTMENT_ID, 90, SALARY)) "Dept 90",
6     SUM(SALARY) TOTAL
7 FROM EMPLOYEES
8 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.

|   | LOCATION_ID | STREET_ADDRESS           | CITY                | STATE_PROVINCE | COUNTRY_NAME             |
|---|-------------|--------------------------|---------------------|----------------|--------------------------|
| 1 | 1400        | 2014 Jabberwocky Rd      | Southlake           | Texas          | United States of America |
| 2 | 1500        | 2011 Interiors Blvd      | South San Francisco | California     | United States of America |
| 3 | 1700        | 2004 Charade Rd          | Seattle             | Washington     | United States of America |
| 4 | 1800        | 460 Bloor St. W.         | Toronto             | Ontario        | Canada                   |
| 5 | 2500        | Magdalen Centre, The ... | Oxford              | Oxford         | United Kingdom           |

**Solution:**

```

1 SELECT
2     LOCATION_ID ,
3     STREET_ADDRESS ,
4     CITY ,
5     STATE_PROVINCE ,
6     COUNTRY_NAME
7 FROM LOCATIONS
8 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.

|     | LAST_NAME | DEPARTMENT_ID | DEPARTMENT_NAME |
|-----|-----------|---------------|-----------------|
| 1   | Whalen    | 10            | Administration  |
| 2   | Hartstein | 20            | Marketing       |
| 3   | Fay       | 20            | Marketing       |
| 4   | Davies    | 50            | Shipping        |
| 5   | Vargas    | 50            | Shipping        |
| 6   | Rajs      | 50            | Shipping        |
| 7   | Mourgos   | 50            | Shipping        |
| 8   | Matos     | 50            | Shipping        |
| 9   | Hunold    | 60            | IT              |
| 10  | Ernst     | 60            | IT              |
| ... |           |               |                 |
| 18  | Higgins   | 110           | Accounting      |
| 19  | Gietz     | 110           | Accounting      |

**Solution:**

```

1 SELECT
2     LAST_NAME ,
3     DEPARTMENT_ID ,
4     DEPARTMENT_NAME
5 FROM EMPLOYEES
6 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.

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

**Solution:**

```

1 SELECT
2     LAST_NAME ,
3     JOB_ID ,
4     DEPARTMENT_ID ,
5     DEPARTMENT_NAME
6 FROM EMPLOYEES
7 JOIN DEPARTMENTS USING (DEPARTMENT_ID)
8 JOIN LOCATIONS USING (LOCATION_ID)
9 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 | EMP# | Manager | 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;

```



**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.

| A   | Employee | A | EMP# | A         | Manager | A      | 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    |      |
| 20  | Gietz    |   | 206  | Higgins   |         | 205    |      |

**Solution:**

```

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 LEFT JOIN EMPLOYEES M ON (E.MANAGER_ID = M.EMPLOYEE_ID)
8 ORDER BY E.EMPLOYEE_ID;
```

**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.

| A   | DEPARTMENT | A         | EMPLOYEE  | A | 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     |   |           |

**Solution:**

```

1 SELECT
2     E.DEPARTMENT_ID DEPARTMENT,
3     E.LAST_NAME EMPLOYEE,
4     C.LAST_NAME COLLEAGUE
5 FROM EMPLOYEES E
6 JOIN EMPLOYEES C ON (E.DEPARTMENT_ID = C.DEPARTMENT_ID)
7 WHERE E.EMPLOYEE_ID <> C.EMPLOYEE_ID
8 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.

| DESC JOB_GRADES |      |             |
|-----------------|------|-------------|
| Name            | Null | Type        |
| -----           |      |             |
| GRADE_LEVEL     |      | VARCHAR2(3) |
| LOWEST_SAL      |      | NUMBER      |
| HIGHEST_SAL     |      | NUMBER      |
| 3 rows selected |      |             |

|     | LAST_NAME | JOB_ID     | DEPARTMENT_NAME | SALARY | GRADE_LEVEL |
|-----|-----------|------------|-----------------|--------|-------------|
| 1   | Vargas    | ST_CLERK   | Shipping        | 2500   | A           |
| 2   | Matos     | ST_CLERK   | Shipping        | 2600   | A           |
| 3   | Davies    | ST_CLERK   | Shipping        | 3100   | B           |
| 4   | Rajs      | ST_CLERK   | Shipping        | 3500   | B           |
| 5   | Lorentz   | IT_PROG    | IT              | 4200   | B           |
| 6   | Whalen    | AD_ASST    | Administration  | 4400   | B           |
| 7   | Mourgos   | ST_MAN     | Shipping        | 5800   | B           |
| 8   | Ernst     | IT_PROG    | IT              | 6000   | C           |
| 9   | Fay       | MK_REP     | Marketing       | 6000   | C           |
| 10  | Gietz     | AC_ACCOUNT | Accounting      | 8300   | C           |
| ... |           |            |                 |        |             |
| 18  | De Haan   | AD_VP      | Executive       | 17000  | E           |
| 19  | King      | AD_PRES    | Executive       | 24000  | E           |

**Solution:**

```

1 SELECT
2     LAST_NAME,
3     JOB_ID,
4     DEPARTMENT_NAME,
5     SALARY,
6     GRADE_LEVEL
7 FROM EMPLOYEES
8 JOIN DEPARTMENTS USING (DEPARTMENT_ID)
9 JOIN JOB_GRADES ON (SALARY BETWEEN LOWEST_SAL AND HIGHEST_SAL)
10 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:**

```

1 SELECT
2     E.LAST_NAME ,
3     TO_CHAR(E.HIRE_DATE , 'DD-MON-YY') HIRE_DATE
4 FROM EMPLOYEES E
5 JOIN EMPLOYEES DAVIES ON (DAVIES.LAST_NAME = 'Davies')
6 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   |

**Solution:**

```

1 SELECT
2     E.LAST_NAME ,
3     TO_CHAR(E.HIRE_DATE , 'DD-MON-YY') HIRE_DATE ,
4     M.LAST_NAME ,
5     TO_CHAR(M.HIRE_DATE , 'DD-MON-YY') HIRE_DATE1
6 FROM EMPLOYEES E
7 JOIN EMPLOYEES M ON (E.MANAGER_ID = M.EMPLOYEE_ID)
8 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).

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

**Solution:**

```

1 SELECT
2     LAST_NAME ,
3     TO_CHAR(HIRE_DATE , 'DD-MON-YY') HIRE_DATE
4 FROM EMPLOYEES
5 WHERE DEPARTMENT_ID = (
6     SELECT DEPARTMENT_ID
7     FROM EMPLOYEES
8     WHERE LAST_NAME = INITCAP('&&last_name')
9 )
10 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 | 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  |

**Solution:**

```
1 SELECT
2     EMPLOYEE_ID ,
3     LAST_NAME ,
4     SALARY
5 FROM EMPLOYEES
6 WHERE SALARY > (
7     SELECT AVG(SALARY)
8     FROM EMPLOYEES
9 )
10 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   |

**Solution:**

```
1 SELECT
2     EMPLOYEE_ID ,
3     LAST_NAME
4 FROM EMPLOYEES
5 WHERE DEPARTMENT_ID IN (
6     SELECT DEPARTMENT_ID
7     FROM EMPLOYEES
8     WHERE LAST_NAME LIKE '%u%'
9 );
```

**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.

|   | A<br>2 | LAST_NAME | A<br>2 | DEPARTMENT_ID | A<br>2 | 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:**

```

1 SELECT
2     LAST_NAME ,
3     DEPARTMENT_ID ,
4     JOB_ID
5 FROM EMPLOYEES
6 WHERE DEPARTMENT_ID IN (
7     SELECT DEPARTMENT_ID
8     FROM DEPARTMENTS
9     WHERE LOCATION_ID = 1700
10 )
11 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.

```

1 SELECT
2     LAST_NAME ,
3     DEPARTMENT_ID ,
4     JOB_ID
5 FROM EMPLOYEES
6 WHERE DEPARTMENT_ID IN (
7     SELECT DEPARTMENT_ID
8     FROM DEPARTMENTS
9     WHERE LOCATION_ID = &LOCATION_ID
10 )
11 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.

|   | A<br>2 | LAST_NAME | A<br>2 | SALARY |
|---|--------|-----------|--------|--------|
| 1 |        | Kochhar   |        | 17000  |
| 2 |        | De Haan   |        | 17000  |
| 3 |        | Mourgos   |        | 5800   |
| 4 |        | Zlotkey   |        | 10500  |
| 5 |        | Hartstein |        | 13000  |

**Solution:**

```

1 SELECT
2     LAST_NAME ,
3     SALARY
4 FROM EMPLOYEES
5 WHERE MANAGER_ID = (
6     SELECT EMPLOYEE_ID
7     FROM EMPLOYEES
8     WHERE LAST_NAME = 'King'
9 );

```

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

**Solution:**

```

1 SELECT
2     DEPARTMENT_ID ,
3     LAST_NAME ,
4     JOB_ID
5 FROM EMPLOYEES
6 WHERE DEPARTMENT_ID = (
7     SELECT DEPARTMENT_ID
8     FROM DEPARTMENTS
9     WHERE DEPARTMENT_NAME = 'Executive'
10 );

```

**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.

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

**Solution:**

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