

Department of Information Systems and Technologies
2025-2026 Fall Semester
CTIS259 Database Management Systems and Applications
Lab Guide 05

Instructor : Nimet Ceren SERİM	Week: 4
Assistant : Engin Zafer KIRAÇBEDEL, Hatice Zehra YILMAZ	Date: 09-10.10.2025
Aim of this lab session: 1. Practice 2-1: Retrieving Data Using the SQL SELECT Statement 2. Practice 4-1: Using Single-Row Functions to Customize Output	

ORACLE Server Configurations:

IP Address: 139.179.33.231

Port number: 1522

SID: orclctis

PLEASE USE ORAx accounts

Practices for Lesson 2

Lesson Overview

In this practice, you write simple `SELECT` queries. The queries cover most of the `SELECT` clauses and operations that you learned in this lesson.

Practice 2-1: Retrieving Data Using the SQL `SELECT` Statement

Part 1

Test your knowledge:

1. The following `SELECT` statement executes successfully:

```
SELECT last_name, job_id, salary AS Sal
FROM employees;
True/False
```

2. The following `SELECT` statement executes successfully:


```
SELECT *
FROM job_grades;
True/False
```

3. There are four coding errors in the following statement. Can you identify them?

```
SELECT employee_id, last_name
sal x 12 ANNUAL SALARY
FROM employees;
```

Part 2

Note the following points before you begin with the practices:

- Save all your lab files on your computer,
- SQL developer:
 - make sure that the required SQL worksheet is active and then from the **File Menu**, select **Save As** to save your SQL statement as a lab_<lessonno>_<stepno>.sql script. When you are modifying an existing script, make sure that you use **Save As** to save it with different file name.
 - To run the query, click the **Run Statement**  icon or **CTRL+Enter** or **F9**. For DML and DDL statements, use the **Run Script** icon or press **F5**.
 - After you have executed the query, make sure that you do not enter your next query in the same worksheet. Open a new worksheet.

You have been hired as a SQL programmer for Acme Corporation. Your first task is to create some reports based on data from the **Human Resources** tables.

4. Your first task is to determine the structure of the `DEPARTMENTS` table and its contents.

Name	Null	Type
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR2(30)
MANAGER_ID		NUMBER(6)
LOCATION_ID		NUMBER(4)

4 rows selected

	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
1	10	Administration	200	1700
2	20	Marketing	201	1800
3	50	Shipping	124	1500
4	60	IT	103	1400
5	80	Sales	149	2500
6	90	Executive	100	1700
7	110	Accounting	205	1700
8	190	Contracting	(null)	1700

5. Determine the structure of the `EMPLOYEES` table.

Name	Null	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

The HR department wants a query to display the last name, job ID, hiredate, and employee ID for each employee, with the employee ID appearing first. Provide an alias `STARTDATE` for the `HIRE_DATE` column. Save your SQL statement to a file named `lab_02_05.sql` so that you can dispatch this file to the HR department.

6. Test your query in the `lab_02_05.sql` file to ensure that it runs correctly.

Note: After you have executed the query, make sure that you do not enter your next query in the same worksheet. Open a new worksheet.

	EMPLOYEE_ID	LAST_NAME	JOB_ID	STARTDATE
1	100	King	AD_PRES	17-JUN-87
2	101	Kochhar	AD_VP	21-SEP-89
3	102	De Haan	AD_VP	13-JAN-93
4	103	Hunold	IT_PROG	03-JAN-90
5	104	Ernst	IT_PROG	21-MAY-91
...				
19	205	Higgins	AC_MGR	07-JUN-94
20	206	Gietz	AC_ACCOUNT	07-JUN-94

7. The HR department wants a query to display all unique job IDs from the `EMPLOYEES` table.

	JOB_ID
1	AC_ACCOUNT
2	AC_MGR
3	AD_ASST
4	AD PRES
5	AD_VP
6	IT_PROG
7	MK_MAN
8	MK_REP
9	SA_MAN
10	SA_REP
11	ST_CLERK
12	ST_MAN

Part 3

8. The HR department wants more descriptive column headings for its report on employees. Copy the statement from `lab_02_05.sql` to a new SQL Worksheet. Name the column headings **Emp #**, **Employee**, **Job**, and **Hire Date**, respectively. Then run the query again.

	Emp #	Employee	Job	Hire Date
1	100	King	AD PRES	17-JUN-87
2	101	Kochhar	AD_VP	21-SEP-89
3	102	De Haan	AD_VP	13-JAN-93
4	103	Hunold	IT_PROG	03-JAN-90
5	104	Ernst	IT_PROG	21-MAY-91

..

19	205	Higgins	AC_MGR	07-JUN-94
20	206	Gietz	AC_ACCOUNT	07-JUN-94

9. The HR department has requested a report of all employees and their job IDs. Display the last name concatenated with the job ID (separated by a comma and space) and name the column **Employee** and **Title**.

	Employee and Title
1	Abel, SA_REP
2	Davies, ST_CLERK
3	De Haan, AD_VP
4	Ernst, IT_PROG
5	Fay, MK_REP

...

19	Whalen, AD_ASST
20	Zlotkey, SA_MAN

Lesson Overview

This practice provides a variety of exercises using different functions that are available for character, number, and date data types.

Practice 4-1: Using Single-Row Functions to Customize Output

1. Write a query to display the system date. Label the column `Date`.

Note: If your database is remotely located in a different time zone, the output will be the date for the operating system on which the database resides.

SYSDATE
09-OCT-25

2. The HR department needs a report to display the employee number, last name, salary, and salary increased by 15.5% (expressed as a whole number) for each employee. Label the column `New Salary`. Save your SQL statement in a file named `Lab_04_02.sql`.

3. Run your query in the `Lab_04_02.sql` file.

	EMPLOYEE_ID	LAST_NAME	SALARY	NewSalary
1	200	Whalen	4400	5082
2	201	Hartstein	13000	15015
3	202	Fay	6000	6930
4	205	Higgins	12000	13860
5	206	Gietz	8300	9587

...

19	176	Taylor	8600	9933
20	178	Grant	7000	8085

4. Modify your query `Lab_04_02.sql` to add a column that subtracts the old salary from the new salary. Label the column `Increase`. Run the revised query.

	EMPLOYEE_ID	LAST_NAME	SALARY	NewSalary	Increase
1	200	Whalen	4400	5082	682
2	201	Hartstein	13000	15015	2015
3	202	Fay	6000	6930	930
4	205	Higgins	12000	13860	1860
5	206	Gietz	8300	9587	1287

...

19	176	Taylor	8600	9933	1333
20	178	Grant	7000	8085	1085

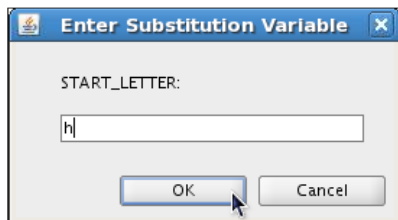
5. Write a query that displays the last name (with the first letter in uppercase and all the other letters in lowercase) and the length of the last name for all employees whose name starts with the letters "J," "A," or "M." Give each column an appropriate label. Sort the results by the employees' last names.

	Name	Length
1	Abel	4
2	Matos	5
3	Mourgos	7

Rewrite the query so that the user is prompted to enter a letter that the last name starts with. For example, if the user enters "H" (capitalized) when prompted for a letter, then the output should show all employees whose last name starts with the letter "H."

	Name	Length
1	Hartstein	9
2	Higgins	7
3	Hunold	6

Modify the query such that the case of the entered letter does not affect the output. The entered letter must be capitalized before being processed by the `SELECT` query.



	Name	Length
1	Hartstein	9
2	Higgins	7
3	Hunold	6

6. The HR department wants to find the duration of employment for each employee. For each employee, display the last name and calculate the number of months between today and the date on which the employee was hired. Label the column as `MONTHS_WORKED`. Order your results by the number of months employed. The number of months must be rounded to the closest whole number.

Note: Because this query depends on the date when it was executed, the values in the `MONTHS_WORKED` column will differ for you.

	LAST_NAME	MONTHS_WORKED
1	Zlotkey	308
2	Mourgos	311
3	Grant	317
4	Lorentz	320
5	Vargas	327
6	Matos	331
7	Taylor	331
8	Fay	338
9	Davies	344
10	Abel	353
11	Hartstein	356
12	Rajs	360
13	Higgins	376
14	Gietz	376
15	De Haan	393
16	Ernst	413
17	Hunold	429
18	Kochhar	433
19	Whalen	457
20	King	460