**UNIVERSIDAD AUTÓNOMA DE ZACATECAS**
**INGENIERÍA DE SOFTWARE**
**LABORATORIO DE SISTEMAS DE BASE DE DATOS II**
**FORMATO DE PRÁCTICAS**

| | |
|---|---|
| **PRÁCTICA:** | **5** |
| **TITULO:** | **Restricting and Sorting Data** |
| **OBJETIVO:** | **Realizar ejercicios sobre los temas del capítulo 2: Restricting and Sorting Data** |
| **DURACIÓN:** | **4 horas** |
| **FECHA:** | |
| **FECHA DE ENTREGA:** | |

## ACTIVIDADES A REALIZAR:

**Ejercicio 1:**

# Practices for Lesson 2

In this practice, you build more reports, including statements that use the WHERE clause and the ORDER BY clause. You make the SQL statements more reusable and generic by including the ampersand substitution.

## *Practice 2-1: Restricting and Sorting Data*

The HR department needs your assistance in creating some queries.

1) Because of budget issues, the HR department needs a report that displays the last name and salary of employees who earn more than $12,000. Save your SQL statement as a file named `lab_02_01.sql`. Run your query.

| | LAST_NAME | SALARY |
|---|---|---|
| 1 | Hartstein | 13000 |
| 2 | King | 24000 |
| 3 | Kochhar | 17000 |
| 4 | De Haan | 17000 |

2) Open a new SQL Worksheet. Create a report that displays the last name and department number for employee number 176. Run the query.

| | LAST_NAME | DEPARTMENT_ID |
|---|---|---|
| 1 | Taylor | 80 |

3) The HR department needs to find high-salary and low-salary employees. Modify `lab_02_01.sql` to display the last name and salary for any employee whose salary is not in the range of $5,000 to $12,000. Save your SQL statement as `lab_02_03.sql`.

| | LAST_NAME | SALARY |
|---|---|---|
| 1 | Whalen | 4400 |
| 2 | Hartstein | 13000 |
| 3 | King | 24000 |
| 4 | Kochhar | 17000 |
| 5 | De Haan | 17000 |
| 6 | Lorentz | 4200 |
| 7 | Rajs | 3500 |
| 8 | Davies | 3100 |
| 9 | Matos | 2600 |
| 10 | Vargas | 2500 |

4) Create a report to display the last name, job ID, and hire date for employees with the last names of Matos and Taylor. Order the query in ascending order by the hire date.

| | LAST_NAME | JOB_ID | HIRE_DATE |
|---|---|---|---|
| 1 | Matos | ST_CLERK | 15-MAR-98 |
| 2 | Taylor | SA_REP | 24-MAR-98 |

5) Display the last name and department ID of all employees in departments 20 or 50 in ascending alphabetical order by name.

| | LAST_NAME | DEPARTMENT_ID |
|---|---|---|
| 1 | Davies | 50 |
| 2 | Fay | 20 |
| 3 | Hartstein | 20 |
| 4 | Matos | 50 |
| 5 | Mourgos | 50 |
| 6 | Rajs | 50 |
| 7 | Vargas | 50 |

6) Modify `lab_02_03.sql` to display the last name and salary of employees who earn between $5,000 and $12,000, and are in department 20 or 50. Label the columns `Employee` and `Monthly Salary`, respectively. Save `lab_02_03.sql` as `lab_02_06.sql` again. Run the statement in `lab_02_06.sql`.

| | Employee | Monthly Salary |
|---|---|---|
| 1 | Fay | 6000 |
| 2 | Mourgos | 5800 |

7) The HR department needs a report that displays the last name and hire date for all employees who were hired in 1994.

| | LAST_NAME | HIRE_DATE |
|---|---|---|
| 1 | Higgins | 07-JUN-94 |
| 2 | Gietz | 07-JUN-94 |

8) Create a report to display the last name and job title of all employees who do not have a manager.

| | LAST_NAME | JOB_ID |
|---|---|---|
| 1 | King | AD_PRES |

9) Create a report to display the last name, salary, and commission of all employees who earn commissions. Sort data in descending order of salary and commissions.
Use the column's numeric position in the `ORDER BY` clause.

| | LAST_NAME | SALARY | COMMISSION_PCT |
|---|---|---|---|
| 1 | Abel | 11000 | 0.3 |
| 2 | Zlotkey | 10500 | 0.2 |
| 3 | Taylor | 8600 | 0.2 |
| 4 | Grant | 7000 | 0.15 |

10) Members of the HR department want to have more flexibility with the queries that you are writing. They would like a report that displays the last name and salary of employees who earn more than an amount that the user specifies after a prompt. Save this query to a file named `lab_02_10.sql`. If you enter 12000 when prompted, the report displays the following results:

| | LAST_NAME | SALARY |
|---|---|---|
| 1 | Hartstein | 13000 |
| 2 | King | 24000 |
| 3 | Kochhar | 17000 |
| 4 | De Haan | 17000 |

11) The HR department wants to run reports based on a manager. Create a query that prompts the user for a manager ID and generates the employee ID, last name, salary, and department for that manager's employees. The HR department wants the ability to sort the report on a selected column. You can test the data with the following values:

manager_id = 103, sorted by last_name:

| | EMPLOYEE_ID | LAST_NAME | SALARY | DEPARTMENT_ID |
|---|---|---|---|---|
| 1 | 104 | Ernst | 6000 | 60 |
| 2 | 107 | Lorentz | 4200 | 60 |

manager_id = 201, sorted by salary:

| | EMPLOYEE_ID | LAST_NAME | SALARY | DEPARTMENT_ID |
|---|---|---|---|---|
| 1 | 202 | Fay | 6000 | 20 |

manager_id = 124, sorted by employee_id:

| | EMPLOYEE_ID | LAST_NAME | SALARY | DEPARTMENT_ID |
|---|---|---|---|---|
| 1 | 141 | Rajs | 3500 | 50 |
| 2 | 142 | Davies | 3100 | 50 |
| 3 | 143 | Matos | 2600 | 50 |
| 4 | 144 | Vargas | 2500 | 50 |

If you have time, complete the following exercises:

12) Display all employee last names in which the third letter of the name is "a."

| | LAST_NAME |
|---|---|
| 1 | Grant |
| 2 | Whalen |

13) Display the last names of all employees who have both an "a" and an "e" in their last name.

| | LAST_NAME |
|---|---|
| 1 | Davies |
| 2 | De Haan |
| 3 | Hartstein |
| 4 | Whalen |

If you want an extra challenge, complete the following exercises:

14) Display the last name, job, and salary for all employees whose jobs are either those of a sales representative or of a stock clerk, and whose salaries are not equal to $2,500, $3,500, or $7,000.

| | LAST_NAME | JOB_ID | SALARY |
|---|---|---|---|
| 1 | Abel | SA_REP | 11000 |
| 2 | Taylor | SA_REP | 8600 |
| 3 | Davies | ST_CLERK | 3100 |
| 4 | Matos | ST_CLERK | 2600 |

15) Modify `lab_02_06.sql` to display the last name, salary, and commission for all employees whose commission is 20%. Save `lab_02_06.sql` as `lab_02_15.sql` again. Rerun the statement in `lab_02_15.sql`.

| | Employee | Monthly Salary | COMMISSION_PCT |
|---|---|---|---|
| 1 | Zlotkey | 10500 | 0.2 |
| 2 | Taylor | 8600 | 0.2 |

**Ejercicio 2:**

**1. Which two clauses of the SELECT statement facilitate selection and projection?**

A. SELECT, FROM

B. ORDER BY, WHERE

C. SELECT, WHERE

D. SELECT, ORDER BY

**2. Choose the query that extracts the LAST_NAME, JOB_ID, and SALARY values from the EMPLOYEES table for records having JOB_ID values of either SA_REP or MK_MAN and having SALARY values in the range of $1000 to $4000. The SELECT and FROM clauses are**

**SELECT LAST_NAME, JOB_ID, SALARY FROM EMPLOYEES:**

A. WHERE JOB_ID IN ('SA_REP','MK_MAN') AND SALARY > 1000 AND SALARY < 4000;

B. WHERE JOB_ID IN ('SA_REP','MK_MAN') AND SALARY BETWEEN 1000 AND 4000;

C. WHERE JOB_ID LIKE 'SA_REP%' AND 'MK_MAN%' AND SALARY > 1000 AND SALARY < 4000;

D. WHERE JOB_ID = 'SA_REP' AND SALARY BETWEEN 1000 AND 4000

OR JOB_ID='MK_MAN';

**3. Which of the following WHERE clauses contains an error? The SELECT and FROM clauses are SELECT * FROM EMPLOYEES:**

A. WHERE HIRE_DATE IN ('02-JUN-2004');

B. WHERE SALARY IN ('1000','4000','2000');

C. WHERE JOB_ID IN (SA_REP,MK_MAN);

D. WHERE COMMISSION_PCT BETWEEN 0.1 AND 0.5;

**4. Choose the WHERE clause that extracts the DEPARTMENT_NAME values containing the character literal "er" from the DEPARTMENTS table. The SELECT and FROM clauses are**

**SELECT DEPARTMENT_NAME FROM DEPARTMENTS:**

A. WHERE DEPARTMENT_NAME IN ('%e%r');

B. WHERE DEPARTMENT_NAME LIKE '%er%';

C. WHERE DEPARTMENT_NAME BETWEEN 'e' AND 'r';

D. WHERE DEPARTMENT_NAME CONTAINS 'e%r';

**5. Which two of the following conditions are equivalent to each other?**

A. WHERE COMMISSION_PCT IS NULL

B. WHERE COMMISSION_PCT = NULL

C. WHERE COMMISSION_PCT IN (NULL)

D. WHERE NOT(COMMISSION_PCT IS NOT NULL)

**6. Which three of the following conditions are equivalent to each other?**

A. WHERE SALARY <=5000 AND SALARY >=2000

B. WHERE SALARY IN (2000,3000,4000,5000)

C. WHERE SALARY BETWEEN 2000 AND 5000

D. WHERE SALARY > 1999 AND SALARY < 5001

E. WHERE SALARY >=2000 AND <=5000

**SORT THE ROWS RETRIEVED BY A QUERY**

**7. Choose one false statement about the ORDER BY clause.**

A. When using the ORDER BY clause, it always appears as the last clause in a SELECT statement.

B. The ORDER BY clause may appear in a SELECT statement that does not contain a WHERE clause.

C. The ORDER BY clause specifies one or more terms by which the retrieved rows are sorted. These terms can only be column names.

D. Positional sorting is accomplished by specifying the numeric position of a column as it appears in the SELECT list, in the ORDER BY clause.

**8. The following query retrieves the LAST_NAME, SALARY, and**

**COMMISSION_PCT values for employees whose LAST_NAME begins with the letter R. Based on the following query, choose the ORDER BY clause that first sorts the results by the COMMISSION_PCT column, listing highest commission earners first, and then sorts the results in ascending order by the SALARY column. Any records with NULL COMMISSION_PCT must appear last:**

**SELECT LAST_NAME, SALARY, COMMISSION_PCT**

**FROM EMPLOYEES**

**WHERE LAST_NAME LIKE 'R%'**

A. ORDER BY COMMISSION_PCT DESC, 2;

B. ORDER BY 3 DESC, 2 ASC NULLS LAST;

C. ORDER BY 3 DESC NULLS LAST, 2 ASC;

D. ORDER BY COMMISSION_PCT DESC, SALARY ASC;

**AMPERSAND SUBSTITUTION**

**9. The DEFINE command explicitly declares a session-persistent substitution variable with a specific value. How is this variable referenced in an SQL statement? Consider an expression that calculates tax on an employee's SALARY based on the current tax rate. For the following session-persistent substitution variable, which statement correctly references the TAX_RATE variable?**

**DEFINE TAX_RATE=0.14**

A. SELECT SALARY * :TAX_RATE TAX FROM EMPLOYEES;

B. SELECT SALARY * &TAX_RATE TAX FROM EMPLOYEES;

C. SELECT SALARY * :&&TAX TAX FROM EMPLOYEES;

D. SELECT SALARY * TAX_RATE TAX FROM EMPLOYEES;

**10. When using ampersand substitution variables in the following query, how many times will you be prompted to input a value for the variable called JOB the first time this query is executed?**

**SELECT FIRST_NAME, '&JOB'**

**FROM EMPLOYEES**

**WHERE JOB_ID LIKE '%'||&JOB||'%'**

**AND '&&JOB' BETWEEN 'A' AND 'Z';**

A. 0

B. 1

C. 2

D. 3,

**Ejercicio 3:**

**Sobre la base de datos "Hotel", elabore 3 sentencias para cada una de las siguientes opciones:**

a) = y <>
b) > y <
c) >= y <=
d) Between y not between
e) In y not in
f) Like y not like
g) Like y not like (ESCAPE)
h) Is null e is not null
i) And
j) Or
k) Uso de paréntesis
l) Precedencia de operadores
m) Order by
n) Variables de sustitución con &
o) Variables de sustitución con &&
p) Define