

Practice 8: Restricting and Sorting Data

Jesús Manuel Juárez Pasillas

October 07 2021

1 Introduction

Queries in the database are very important, since they allow us to know everything we want to know about a database, and with this conclude many things, in addition, the queries allow us to select only the information we want with the conditions, In addition to being able to order the data to our liking.

2 Development

2.1 Activity 1:

Read all the choices carefully because there might be more than one correct answer. Choose all the correct answers for each question.

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

The where and order by clauses allow us to search only the elements that meet one or more conditions in the case of the where and allow us to order the results in the case of the 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';

The first answer is valid if you want the salary values to be consulted not to take the values of 1000 and 4000, and the second answer is valid if the values you want for the salary include the values 1000 and 4000.

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;

The values that are given for job_id are not in single quotes so it would give an error when choosing one since they are not a valid expression.

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'

To find out if a text string contains any character or set of characters, use the like clause together with the characters to search within single quotes and within%.

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)

To find out if an element is null, the is or is not clause has to be used to verify that it 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

The three sentences continue to fulfill a single range that is from 2000 to 5000 including them, the other sentences only allow some values or the sentence is misspelled.

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.

The order is not indicated in numerical form, but rather that the order is given by a column.

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;

To order the data from highest to lowest in the `commission_pct` column, it is done with `desc`, with this the nulls will appear at the end, and `asc` for ascending for the salary.

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;`

To refer to a variable it is necessary to place an `&` before the name of the defined variable.

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

Twice since on two occasions only a `&` is used, so that it only asks for the value once, you have to have a variable `&` and then refer to that variable with the double `&` and the name of the variable.

2.2 Activity 2:

Propose an answer to the following issues:

- a) The SELECT list of a query contains a single column. Is it possible to sort the results retrieved by this query by another column?

Yes, even if the column by which it is being sorted is not shown, the data displayed can be sorted by a column that will not appear in the query.

- b) Ampersand substitution variables support reusability of repetitively executed SQL statements. If a substituted value is to be used multiple times at different parts of the same statement, is it possible to be prompted to submit a substitution value just once and for that value to automatically be substituted during subsequent references to the same variable?

Yes, to request the value it is only necessary to place a variable with the sign & before the name of the variable, and to refer to that variable, && followed by the name of the variable to which it is referenced will be placed.

- c) You have been tasked to retrieve the LAST_NAME and DEPARTMENT_ID values for all rows in the EMPLOYEES table. The output must be sorted by the nullable DEPARTMENT_ID column, and all rows with NULL DEPARTMENT_ID values must be listed last. Is it possible to provide the results as requested?

The order by clause orders the data given a column, and always leaves the null values at the end when a specific order is not placed or when desc is used, it starts with null, so this request can be made.

- d) You have a complex query with multiple conditions. Is there a restriction on the number of conditions you can specify in the WHERE clause? Is there a limit to the number of comparison operators you can use in a single query?

There are no limits to the number of conditions and operators that can be used.

- e) You have been tasked to locate rows in the EMPLOYEES table where the SALARY values contain the numbers 8 and 0 adjacent to each other. The SALARY column has a NUMBER data type. Is it possible to use the LIKE comparison operator with numeric data?

If it can be used, since the salary will be compared with a chain, this is also made a chain for the comparison.

- f) By restricting the rows returned from the JOBS table to those which contain the value SA_REP in the JOB_ID column, is a projection, selection or join performed?

A selection since the records that meet the condition are being selected.

2.3 Activity 3:

Connect to the OE schema and complete the following tasks.

A customer requires a hard disk drive and a graphics card for her personal computer. She is willing to spend between \$500 and \$800 on the disk drive but is unsure about the cost of a graphics card. Her only requirement is that the resolution supported by the graphics card should be either 1024×768 or 1280×1024. As the sales representative, you have been tasked to write one query that searches the PRODUCT_INFORMATION table where the PRODUCT_NAME value begins with HD (hard disk) or GP (graphics processor) and their list prices. Remember the hard disk list prices must be between \$500 and \$800 and the graphics processors need to support either 1024×768 or 1280×1024. Sort the results in descending LIST_PRICE order.

- **SELECT PRODUCT_NAME, LIST_PRICE FROM PRODUCT_INFORMATION WHERE (PRODUCT_NAME LIKE 'HD%' AND LIST_PRICE BETWEEN 500 AND 800) OR (PRODUCT_NAME = 'GP 1024x768' OR PRODUCT_NAME = 'GP 1280x1024') ORDER BY LIST_PRICE DESC;**

	PRODUCT_NAME	LIST_PRICE
1	HD 18.2GB @10000 /E	800
2	HD 12GB @7200 /SE	775
3	HD 12GB /S	633
4	HD 12GB /R	612
5	HD 12GB /N	567
6	HD 12GB /I	543
7	GP 1280x1024	98
8	GP 1024x768	78

Figure 1: Consult the oe scheme.

2.4 Activity 4:

This exercise must be performed using HR schema.

- Retrieve a list of DEPARTMENT_NAME values that end with the three letters “ing” from the DEPARTMENTS table.
SELECT DEPARTMENT_NAME FROM DEPARTMENTS WHERE DEPARTMENT_NAME LIKE '%ing';

	DEPARTMENT_NAME
1	Marketing
2	Purchasing
3	Shipping
4	Accounting
5	Manufacturing
6	Contracting
7	Recruiting

Figure 2: End with the three letters “ing”.

- The JOBS table contains descriptions of different types of jobs an employee in the organization may occupy. It contains the JOB_ID, JOB_TITLE, MIN_SALARY, and MAX_SALARY columns. You are required to write a query that extracts the JOB_TITLE, MIN_SALARY, and MAX_SALARY columns, as well as an expression called VARIANCE, which is the difference between the MAX_SALARY and MIN_SALARY values, for each row. The results must include only JOB_TITLE values that contain either the word “President” or “Manager.” Sort the list in descending order based on the VARIANCE expression. If more than one row has the same VARIANCE value, then, in addition, sort these rows by JOB_TITLE in reverse alphabetic order.

```
SELECT JOB_TITLE, MIN_SALARY, MAX_SALARY,  
(MAX_SALARY-MIN_SALARY) VARIANCE FROM JOBS  
WHERE JOB_TITLE LIKE '%President%' OR JOB_TITLE LIKE  
'%Manager%' ORDER BY VARIANCE DESC, JOB_TITLE DESC;
```

	JOB_TITLE	MIN_SALARY	MAX_SALARY	VARIANCE
1	President	20080	40000	19920
2	Administration Vice President	15000	30000	15000
3	Sales Manager	10000	20080	10080
4	Finance Manager	8200	16000	7800
5	Accounting Manager	8200	16000	7800
6	Purchasing Manager	8000	15000	7000
7	Marketing Manager	9000	15000	6000
8	Stock Manager	5500	8500	3000

Figure 3: “President” or “Manager” variance.

- A common calculation performed by the Human Resources department relates to the calculation of taxes levied upon an employee. Although, this is done for all employees, there are always a few staff members who dispute the tax deducted from their income. The tax deducted per employee is calculated by obtaining the annual salary for the employee and multiplying this by the current tax rate, which may vary from year to year. You are required to write a reusable query using the current tax rate and the EMPLOYEE_ID number as inputs and return the EMPLOYEE_ID, FIRST_NAME, SALARY, ANNUAL SALARY (SALARY*12), TAX_RATE, and TAX (TAX_RATE * ANNUAL SALARY) information.
SELECT EMPLOYEE_ID, FIRST_NAME, SALARY, (SALARY*12)
ANNUAL_SALARY, &&TAX_RATE AS TAX_RATE,
(&TAX_RATE*(SALARY*12)) TAX FROM EMPLOYEES WHERE
EMPLOYEE_ID = &EMPLOYEE_ID;

Figure 4: Request data.

	EMPLOYEE_ID	FIRST_...		SALARY	ANNUAL_SALARY	TAX_RATE	TAX
1	102	Lex		17000	204000	0.14	28560

Figure 5: Results from tax rate.

2.5 Activity 5:

In this practice, you create more reports, including statements that use the WHERE clause and the ORDER BY clause. You can make SQL statements more reusable and generic by including ampersand substitution. Multiple declarations contain the substitution variables.

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_8.01.sql. Run your query.
 - The file is located inside the "Plugins.rar" file.

	LAST_NAME	SALARY
1	King	24000
2	Kochhar	17000
3	De Haan	17000
4	Greenberg	12008
5	Russell	14000
6	Partners	13500
7	Errazuriz	12000
8	Hartstein	13000
9	Higgins	12008

Figure 6: Employees who earn more than \$12,000.

2. Open a new SQL Worksheet. Create a report that displays the last name and department number for employee number 176. Run the query.
 - **SELECT LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES WHERE EMPLOYEE_ID = &EMPLOYEE_ID;**

	LAST_NAME	DEPARTMENT_ID
1	Taylor	80

Figure 7: Employee with the id:

3. The HR department needs to find high-salary and low-salary employees. Modify lab_8.01.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_8.03.sql.

- The file is located inside the "Plugins.rar" file.

	LAST_NAME	SALARY
1	King	24000
2	Kochhar	17000
3	De Haan	17000
4	Austin	4800
5	Pataballa	4800
6	Lorentz	4200
7	Greenberg	12008
8	Khoo	3100
9	Baida	2900
10	Tobias	2800

Figure 8: Salary is not in the range of \$5,000 to \$12,000.

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.

- **SELECT LAST_NAME, JOB_ID, HIRE_DATE FROM EMPLOYEES WHERE LAST_NAME LIKE 'Matos' OR LAST_NAME LIKE 'Taylor' ORDER BY HIRE_DATE ASC;**

	LAST_NAME	JOB_ID	HIRE_DATE
1	Taylor	SH_CLERK	24/01/06
2	Matos	ST_CLERK	15/03/06
3	Taylor	SA_REP	24/03/06

Figure 9: Last names of Matos and Taylor.

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

- **SELECT LAST_NAME, DEPARTMENT_ID FROM EMPLOYEES WHERE DEPARTMENT_ID IN (&DEPARTMENT_ID_1,&DEPARTMENT_ID_2) ORDER BY FIRST_NAME ASC;**

	LAST_NAME	DEPARTMENT_ID
1	Fripp	50
2	Walsh	50
3	Bull	50
4	Cabrio	50
5	Everett	50
6	Davies	50
7	OConnell	50
8	Grant	50
9	Geoni	50
10	Philtanker	50

Figure 10: Employees in departments 20 or 50.

6. Modify lab_08.03.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. Resave lab_08.03.sql as lab_08.06.sql.

- The file is located inside the "Plugins.rar" file.

	Employee	Monthly Salary
1	Weiss	8000
2	Fripp	8200
3	Kaufling	7900
4	Vollman	6500
5	Mourgos	5800
6	Fay	6000

Figure 11: Employees who earn between \$5,000 and \$12,000.

7. The HR department needs a report that displays the last name and hire date for all employees who were hired in 1994 (The year was changed to 2003, there were no elements in the year 1994, only from 2000 onwards.).
 - **SELECT LAST_NAME, HIRE_DATE FROM EMPLOYEES
WHERE EXTRACT(YEAR FROM HIRE_DATE) = &YEAR;**

	LAST_NAME	HIRE_DATE
1	King	17/06/03
2	Khoo	18/05/03
3	Kaufling	01/05/03
4	Ladwig	14/07/03
5	Rajs	17/10/03
6	Whalen	17/09/03

Figure 12: Employees who were hired in 2003.

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

- **SELECT LAST_NAME, JOB_ID FROM EMPLOYEES WHERE MANAGER_ID IS NULL;**

	LAST_NAME	JOB_ID
1	King	AD_PRES

Figure 13: Employees who do not have manager.

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.

- **SELECT LAST_NAME, SALARY, COMMISSION_PCT FROM EMPLOYEES WHERE COMMISSION_PCT IS NOT NULL ORDER BY SALARY DESC, COMMISSION_PCT DESC;**

	LAST_NAME	SALARY	COMMISSION_PCT
1	Russell	14000	0.4
2	Partners	13500	0.3
3	Errazuriz	12000	0.3
4	Ozer	11500	0.25
5	Cambrault	11000	0.3
6	Abel	11000	0.3
7	Vishney	10500	0.25
8	Zlotkey	10500	0.2
9	King	10000	0.35
10	Tucker	10000	0.3

Figure 14: Employees who earn commissions.

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_8_10.sql. If you enter 12000 when prompted, the report displays the following results:

- The file is located inside the "Plugins.rar" file.

	LAST_NAME	SALARY
1	King	24000
2	Kochhar	17000
3	De Haan	17000
4	Greenberg	12008
5	Russell	14000
6	Partners	13500
7	Hartstein	13000
8	Higgins	12008

Figure 15: lab_8_10.sql.

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:

- **SELECT EMPLOYEE_ID, LAST_NAME, SALARY, DEPARTMENT_ID FROM EMPLOYEES WHERE MANAGER_ID = &MANAGER_ID ORDER BY &COLUMN_NAME TO_ORDER;**

	EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
1	204	Baer	10000	70
2	108	Greenberg	12008	100
3	205	Higgins	12008	110
4	203	Mavris	6500	40
5	200	Whalen	4400	10

Figure 16: manager_id = 101, sorted by last_name.

	EMPLOYEE_ID	LAST_NAME	SALARY	DEPARTMENT_ID
1	200	Whalen	4400	10
2	203	Mavris	6500	40
3	204	Baer	10000	70
4	205	Higgins	12008	110
5	108	Greenberg	12008	100

Figure 17: manager_id = 101, sorted by salary.

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

- **SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE ‘_a%’;**

	LAST_NAME
1	Grant
2	Grant
3	Whalen

Figure 18: Third letter of the name is “a”.

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

- **SELECT LAST_NAME FROM EMPLOYEES WHERE LAST_NAME LIKE ‘%a%’ OR LAST_NAME LIKE ‘%e%’;**

	LAST_N...
1	Abel
2	Ande
3	Baer
4	Baida
5	Banda
6	Bates
7	Bell
8	Bernstein
9	Cabrio
10	Cambrault

Figure 19: Employees who have both an “a” and an “e” in their last name.

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.

- **SELECT LAST_NAME, JOB_ID, SALARY FROM EMPLOYEES WHERE JOB_ID IN (SELECT JOB_ID FROM JOBS WHERE JOB_TITLE IN ('Sales Representative','Stock Clerk')) AND SALARY NOT IN (2500,3500,7000);**

	LAST_N...	JOB_ID	SALARY
1	Tucker	SA_REP	10000
2	Bernstein	SA_REP	9500
3	Hall	SA_REP	9000
4	Olsen	SA_REP	8000
5	Cambrault	SA_REP	7500
6	King	SA_REP	10000
7	Sully	SA_REP	9500
8	McEwen	SA_REP	9000
9	Smith	SA_REP	8000
10	Doran	SA_REP	7500

Figure 20: Sales representative or of a stock clerk.

15. Modify lab_8.06.sql to display the last name, salary, and commission for all employees whose commission is 20%. Resave lab_8.06.sql as lab_8.15.sql. Rerun the statement in lab_8.15.sql.

- The file is located inside the "Plugins.rar" file.

	Emplo...	Monthly Salary	Commission
1	Zlotkey	10500	0.2
2	Olsen	8000	0.2
3	Cambrault	7500	0.2
4	Bloom	10000	0.2
5	Fox	9600	0.2
6	Taylor	8600	0.2
7	Livingston	8400	0.2

Figure 21: Employees whose commission is 20%.

3 Pre-Assessment:

- Practices pre-Assessment for Database Systems Laboratory II

Practice	Pre-Assessment
COMPLIES WITH THE REQUESTED FUNCTIONALITY	X
HAS THE CORRECT INDENTATION	X
HAS AN EASY WAY TO ACCESS THE PROVIDED FILES	X
HAS A REPORT WITH IDC FORMAT	X
REPORT INFORMATION IS FREE OF SPELLING ERRORS	X
DELIVERED IN TIME AND FORM	X
IS FULLY COMPLETED (SPECIFY THE PERCENTAGE COMPLETED)	100

Table 1: Pre-Assessment.

4 Conclusion:

The select statements in Oracle allow us a great variety of options with which we can give a more understandable format to the results. This allows us to create queries where the results have an order, only show data that meets specific conditions and only the data that is required is obtained.