

Operatori și funcții

Cuprins

Ordinea în care se execută operațiile din comanda SELECT	1
Operatori de comparație	2
Operatori logici	3
Sortarea / Ordonarea rândurilor - Clauza ORDER BY.....	4
Single-Row Functions	5
Funcții pentru prelucrarea șirurilor de caractere	5
Multiple-Row Functions	8

Ordinea în care se execută operațiile din comanda SELECT

- clauza FROM: localizează tabela care conține datele ce urmează a fi prelucrate/afișate
- clauza WHERE: restricționează rândurile ce vor fi afișate
- SELECT: selectează din setul de rânduri rezultat după aplicarea restricțiilor lui WHERE doar coloanele specificate de SELECT
- clauza ORDER BY: ordonează setul de rezultate obținut după aplicarea pașilor precedenți

Operatori de comparație

< <= > >= = <>		
BETWEEN ... AND	operator folosit pentru a selecta și afișa rânduri aflate într-un domeniu de valori precizat prin limita inferioară și limita superioară, inclusiv	<pre>SELECT last_name, salary FROM employees WHERE salary BETWEEN 9000 AND 11000;</pre>
IN	operator folosit pentru a testa dacă o valoare <u>apartine</u> unui set de valori	<pre>SELECT city, state_province, country_id FROM locations WHERE country_id IN('UK', 'CA');</pre>
LIKE	operator care permite selectarea rândurilor care conțin un anumit șablon pentru șiruri de caractere, date calendaristice sau numere % se folosește în șablon pentru a reprezenta o secvență de 0 sau mai multe caractere _ se folosește în șablon pentru a reprezenta un singur caracter	<pre>SELECT last_name FROM employees WHERE last_name LIKE '_o%';</pre> afișează angajații al căror nume are a doua literă o

Operatori logici

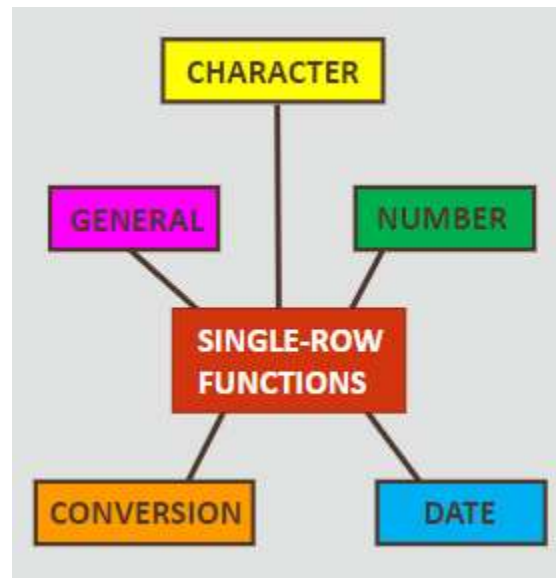
AND	<pre>SELECT last_name, hire_date, job_id FROM employees WHERE hire_date > '01-Jan-1998' AND job_id LIKE 'SA%';</pre>
OR	
NOT	<pre>SELECT department_name, location_id FROM departments WHERE location_id NOT IN (1700,1800);</pre>
	<pre>SELECT last_name ' ' salary*1.05 AS "Employee Raise", department_id, first_name FROM employees WHERE department_id IN(50,80) OR first_name LIKE 'C%' AND last_name LIKE '%s%';</pre> <p>ORACLE</p> <p>The order of operations is: first_name starts with a "C" AND last_name contains an "s". Both these conditions must be met to be returned. Any instance of employees in department 50 and 80 will be returned.</p>

Sortarea / Ordonarea rândurilor - Clauza ORDER BY

pentru ordonare descrescătoare se adaugă numelui coloanei cuvântul cheie DESC	<pre>SELECT last_name, hire_date FROM employees ORDER BY hire_date DESC;</pre>
ordonare după o coloană care nu apare în lista SELECT	<pre>SELECT employee_id, first_name FROM employees WHERE employee_id < 105 ORDER BY last_name;</pre>
ordonare folosind aliasul unei coloane	<pre>SELECT last_name, hire_date AS "Date Started" FROM employees ORDER BY "Date Started";</pre>
ordonare după mai multe coloane	<pre>SELECT department_id, last_name FROM employees WHERE department_id <= 50 ORDER BY department_id, last_name;</pre>
<ul style="list-style-type: none">Note: The ORDER BY clause must be the last clause of the SQL statement	

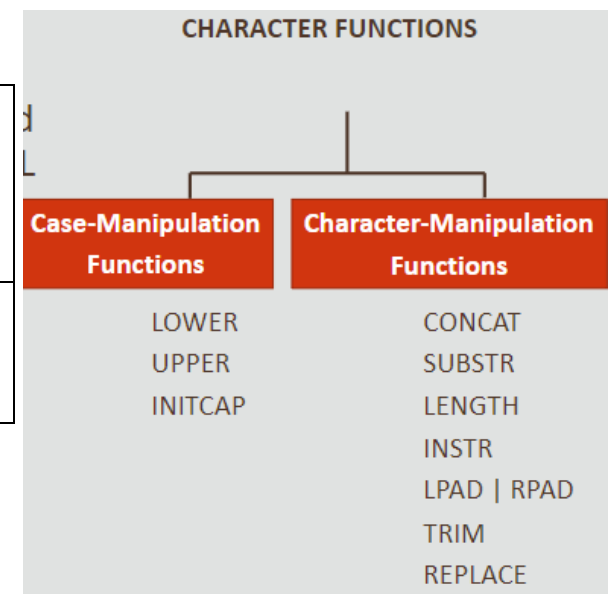
Single-Row Functions

- acționează pe un singur rând, acceptă unul sau mai multe argumente și returnează un singur rezultat pentru fiecare rând



Funcții pentru prelucrarea șirurilor de caractere

LOWER	se aplică unei coloane sau unei expresii și returnează șirul corespunzător în care literele sunt convertite la minusculă (literă mică)	<pre>SELECT last_name FROM employees WHERE LOWER(last_name) = 'abel';</pre>
UPPER	se aplică unei coloane sau unei expresii și returnează șirul corespunzător în care	<pre>SELECT last_name FROM empl oyees WHERE UPPER(last_name) = 'ABEL';</pre>



	literele sunt convertite la majusculă (literă mare)	
INITCAP	se aplică unei coloane sau unei expresii și returnează șirul corespunzător în care prima literă din fiecare cuvânt este majusculă	<pre>SELECT last_name FROM employees WHERE INITCAP(last_name) = 'Abel';</pre>
Notă: Șirul de caractere cu care se face comparația trebuie să respecte formatul rezultatului returnat de funcție.		

CONCAT	concatenează două șiruri de caractere	Examples:	Result
		SELECT CONCAT('Hello', 'World') FROM DUAL;	HelloWorld
		SELECT CONCAT(first_name, last_name) FROM employees;	EllenAbel CurtisDavies ...
SUBSTR	<ul style="list-style-type: none"> extrage un șir de caractere dintr-un șir dat are 3 argumente: <u>șirul</u> din care se face extragerea, <u>poziția de început</u> pentru extragere subșir, <u>număr de caractere</u> care se extrag (opțional). Dacă nu este precizat numărul de caractere pentru 	Examples:	Result
		SELECT SUBSTR('HelloWorld', 1, 5) FROM DUAL;	Hello
		SELECT SUBSTR('HelloWorld', 6) FROM DUAL;	World
		SELECT SUBSTR(last_name, 1, 3) FROM employees;	Abe Dav

	extragerea subșirului atunci se vor lua toate caracterele până la sfârșitul șirului.								
LENGTH	returnează lungimea șirului de caractere pe care îl are ca argument	<table><tr><th>Examples:</th><th>Result</th></tr><tr><td>SELECT LENGTH('HelloWorld') FROM DUAL;</td><td>10</td></tr><tr><td>SELECT LENGTH(last_name) FROM employees;</td><td>4 6 ...</td></tr></table>	Examples:	Result	SELECT LENGTH('HelloWorld') FROM DUAL;	10	SELECT LENGTH(last_name) FROM employees;	4 6 ...	
Examples:	Result								
SELECT LENGTH('HelloWorld') FROM DUAL;	10								
SELECT LENGTH(last_name) FROM employees;	4 6 ...								
INSTR	returnează poziția primei apariții a unui subșir într-un șir. Dacă subșirul nu este găsit, funcția returnează valoarea 0.	<table><tr><th>Examples:</th><th>Result</th></tr><tr><td>SELECT INSTR('HelloWorld', 'W') FROM DUAL;</td><td>6</td></tr><tr><td>SELECT last_name, INSTR(last_name, 'a') FROM employees;</td><td>Abel 0 Davies 2 ...</td></tr></table>	Examples:	Result	SELECT INSTR('HelloWorld', 'W') FROM DUAL;	6	SELECT last_name, INSTR(last_name, 'a') FROM employees;	Abel 0 Davies 2 ...	
Examples:	Result								
SELECT INSTR('HelloWorld', 'W') FROM DUAL;	6								
SELECT last_name, INSTR(last_name, 'a') FROM employees;	Abel 0 Davies 2 ...								
LPAD	<ul style="list-style-type: none">• umple partea stângă a unui șir de caractere cu un caracter dat, până la un șir cu un număr precizat de caractere• are 3 argumente: <u>șirul de caractere</u>, <u>numărul de caractere</u> pe care trebuie să le conțină șirul rezultat, <u>caracterul</u> cu care se	<table><tr><th>Examples:</th><th>Result</th></tr><tr><td>SELECT LPAD('HelloWorld', 15, '-') FROM DUAL;</td><td>-----HelloWorld</td></tr><tr><td>SELECT LPAD(last_name, 10, '*') FROM employees;</td><td>*****Abel ****Davies ...</td></tr></table>	Examples:	Result	SELECT LPAD('HelloWorld', 15, '-') FROM DUAL;	-----HelloWorld	SELECT LPAD(last_name, 10, '*') FROM employees;	*****Abel ****Davies ...	
Examples:	Result								
SELECT LPAD('HelloWorld', 15, '-') FROM DUAL;	-----HelloWorld								
SELECT LPAD(last_name, 10, '*') FROM employees;	*****Abel ****Davies ...								

	face umplerea										
RPAD	similar cu RPAD, completarea făcându-se la dreapta (R - right)	<table><tr><th>Examples:</th><th>Result</th></tr><tr><td>SELECT RPAD('HelloWorld', 15, '-') FROM DUAL;</td><td>HelloWorld-----</td></tr><tr><td>SELECT RPAD(last_name, 10, '*') FROM employees;</td><td>Abel***** Davies**** ...</td></tr></table>	Examples:	Result	SELECT RPAD('HelloWorld', 15, '-') FROM DUAL;	HelloWorld-----	SELECT RPAD(last_name, 10, '*') FROM employees;	Abel***** Davies**** ...			
Examples:	Result										
SELECT RPAD('HelloWorld', 15, '-') FROM DUAL;	HelloWorld-----										
SELECT RPAD(last_name, 10, '*') FROM employees;	Abel***** Davies**** ...										
TRIM	elimină toate caracterele specificate: <ul style="list-style-type: none">• de la începutul șirului LEADING• de la sfârșitul șirului TRAILING• de la ambele capete BOTH	<table><tr><th>Examples:</th><th>Result</th></tr><tr><td>SELECT TRIM(LEADING 'a' FROM 'abcba') FROM DUAL;</td><td>bcba</td></tr><tr><td>SELECT TRIM(TRAILING 'a' FROM 'abcba') FROM DUAL;</td><td>abcb</td></tr><tr><td>SELECT TRIM(BOTH 'a' FROM 'abcba') FROM DUAL;</td><td>bcb</td></tr></table>	Examples:	Result	SELECT TRIM(LEADING 'a' FROM 'abcba') FROM DUAL;	bcba	SELECT TRIM(TRAILING 'a' FROM 'abcba') FROM DUAL;	abcb	SELECT TRIM(BOTH 'a' FROM 'abcba') FROM DUAL;	bcb	
Examples:	Result										
SELECT TRIM(LEADING 'a' FROM 'abcba') FROM DUAL;	bcba										
SELECT TRIM(TRAILING 'a' FROM 'abcba') FROM DUAL;	abcb										
SELECT TRIM(BOTH 'a' FROM 'abcba') FROM DUAL;	bcb										
REPLACE											

Multiple-Row Functions

- poate prelucra grupuri de rânduri și returnează un singur rezultat pentru acel grup de rânduri