

## Funcții SQL – single row

Funcțiile de tip single row sunt funcții utilizate în interogări și returnează un rezultat pentru fiecare linie din tabelul interogată. Pot fi utilizate în cadrul listei select, în clauzele WHERE, HAVING.

Apelate cu argument NULL funcțiile returnează NULL. Excepție fac CONCAT, NVL, REPLACE și REGEXP\_REPLACE.

**Funcții numerice** - primesc ca argument și returnează date de tip numeric

<b>CEIL (n)</b>	Rotunjește superior pe n	SELECT CEIL(21.3) AS "Ceiling" FROM DUAL;
<b>FLOOR(n)</b>	Rotunjește inferior pe n	SELECT FLOOR(21.3) AS "Flooring" FROM DUAL;
<b>ROUND(n [,m])</b>	Rotunjește pe n la m zecimale. Implicit m este 0, echivalent cu rotunjirea la cel mai apropiat întreg. m poate fi și negativ.	SELECT ROUND(21.365,2) AS "Rounding" FROM DUAL;  SELECT ROUND (21.665,-1) AS "Rounding " FROM DUAL;
<b>TRUNC(n [,m])</b>	Trunchează pe n la m zecimale; implicit m e 0; m poate fi și negativ.	SELECT TRUNC(21.365,2) AS "Truncating" FROM DUAL;  SELECT TRUNC(21.665,-1) AS "Truncating " FROM DUAL;
<b>MOD(m, n)</b>	Returnează restul împărțirii lui m la n	SELECT MOD (26,11) AS "Mod" FROM DUAL;

La acestea se mai adaugă funcțiile trigonometrice uzuale (**COS(n)**, **ACOS(n)**, **SIN(n)**, **ASIN(n)**, **TAN(n)**, **ATAN(n)**), funcțiile de calcul de logaritmi (**LN(n)**, **LOG(base, n)**), ridicarea la putere (**EXP(n)**, **POWER(m,n)**, rădăcina pătrată (**SQRT(n)**).

**Funcții ce iau ca argument date caracter și returnează valori numerice**

<b>LENGTH (char)</b>	Returnează lungimea șirului char.	SELECT LENGTH('word') AS "Length in characters" FROM DUAL;
<b>ASCII (char)</b>	Returnează codul ASCII al primului caracter din șir	SELECT ASCII('A') FROM DUAL;
<b>INSTR(char1, char2 [, n [, m ]])</b>	Caută a m-a apariție a șirului char2 în șirul char1 începând căutarea de la poziția n; returnează poziția în char1 a primului caracter din subșirul identificat, relativ la n. Implicit n și m sunt 1. Dacă nu se identifică nici o apariție rezultatul este 0.	SELECT INSTR('CORPORATE FLOOR','OR',3,2) AS "Instring" FROM DUAL;

### Funcții caracter – returnează date de tip caracter

<b>INITCAP(char)</b>	Prima literă a fiecărui cuvânt e transformată în majusculă.	SELECT INITCAP('the soap') "Capitals" FROM DUAL;
<b>UPPER(char)</b>	Returnează șirul char rescris cu majuscule	SELECT UPPER('Carol') FROM DUAL;
<b>LOWER(char)</b>	Returnează șirul char rescris cu litere mici	SELECT LOWER('LOWER') FROM DUAL;
<b>CONCAT(char1, char2)</b>	Concatenează șirurile argument echivalent cu operatorul de concatenare (  )	SELECT CONCAT( CONCAT(ename, ' is a '), job) "Job" FROM emp WHERE empno = 7900;
<b>SUBSTR(char, m [, n ])</b>	Extrage din șirul char n caractere începând de la poziția m; omisia lui n semnifică finalul șirului; o valoare negativă pentru m semnifică numărarea de la stânga la dreapta	SELECT SUBSTR('ABCDEFGF', 3, 4) "Subs" FROM DUAL;
<b>RPAD(char1, n [, char2 ])</b>	Returnează char1 cu spații adăugate la dreapta până la lungimea n, dacă se omite char2; altfel char2 este copiat de oricâte ori e nevoie pentru a completa char1 până la lungimea n	SELECT RPAD('ename', 12, 'ab') "RPAD example" FROM emp WHERE ename = 'TURNER';
<b>LPAD(char1, n [, char2 ])</b>	Similar RPAD cu diferența că alipirea se face la stânga	SELECT LPAD('Page1', 15, '*.') "LPAD example" FROM DUAL;
<b>CHR(n)</b>	Returnează caracterul corespunzător codului binar n, ca VARCHAR2	SELECT CHR(67)    CHR(65)    CHR(84) FROM DUAL;
<b>TRIM (char)</b>	Elimină spațiile goale de la începutul și sfârșitul textului	
<b>LTRIM(char)</b>	Elimină spațiile goale de la începutul textului	
<b>REPLACE(char, search_string [, replacement_string])</b>	Returnează șirul char în care subșirul search_string este înlocuit de replacement_string; absența ultimului reprezintă ștergerea tuturor aparițiilor subșirului căutat	SELECT REPLACE('JACK and JUE', 'J', 'BL') "Changes" FROM DUAL;
<b>TRANSLATE(char, from, to)</b>	Returnează char modificat astfel: fiecare apariție a unui caracter din șirul from este înlocuită de corespondentul (caracterul de pe aceeași poziție) din șirul to; dacă from este mai lung decât to,	SELECT TRANSLATE('2KRW229', '0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ', '9999999999XXXXXXXXXXXXXXXXXXXXXXXXXXXXX') "Licence" FROM DUAL;

	caracterele fără correspondent sunt șterse	
--	--	--

### Funcții pentru tipul DATE

<b>CURRENT_DATE,</b>	Returnează data curentă relativ la zona de timp (time zone) a sesiunii curente	SELECT CURRENT_DATE FROM DUAL;
<b>SYSDATE</b>	Returnează data curentă a sistemului	SELECT SYSDATE FROM DUAL;
<b>MONTHS_BETWEEN(d1, d2 )</b>	Numărul de luni dintre d1 și d2	SELECT MONTHS_BETWEEN( TO_DATE('02-02-1995','MM-DD-YYYY'), TO_DATE('01-01-1995','MM-DD-YYYY') ) AS "Months" FROM DUAL;
<b>ADD_MONTHS(d, n)</b>		SELECT TO_CHAR(ADD_MONTHS(hiredate,1)), 'DD-MM-YYYY' "Next month" FROM emp WHERE ename = 'SMITH'
<b>NEXT_DAY(d , char)</b>	Returnează data calendaristică corespunzătoare zilei lucrătoare specificate de char imediat următoare datei d	SELECT NEXT_DAY('15-MAR-92','TUESDAY') "NEXT DAY" FROM DUAL;
<b>LAST_DAY(d)</b>	Returnează data calendaristică corespunzătoare ultimei zile din luna specificată de d	SELECT SYSDATE, LAST_DAY(SYSDATE) AS "Last" FROM DUAL
<b>ROUND(d [, fmt ])</b>	Studiați formatul fmt la adresa <a href="http://download.oracle.com/docs/cd/B19306_01/server.102/b14200/functions230.htm">http://download.oracle.com/docs/cd/B19306_01/server.102/b14200/functions230.htm</a>	SELECT ROUND (TO_DATE ('27-OCT-00'), 'YEAR') "New Year" FROM DUAL;
<b>TRUNC(d [, fmt ])</b>	Studiați formatul fmt la adresa <a href="http://download.oracle.com/docs/cd/B19306_01/server.102/b14200/functions230.htm">http://download.oracle.com/docs/cd/B19306_01/server.102/b14200/functions230.htm</a>	SELECT TRUNC(TO_DATE('27-OCT-92','DD-MON-YY'), 'YEAR') "New Year" FROM DUAL;

### Funcții de conversie

Oracle face implicit următoarele conversii de tipuri:

VARCHAR2 -> NUMBER  
 CHAR -> NUMBER  
 VARCHAR2 -> DATE  
 CHAR -> DATE  
 NUMBER -> VARCHAR2

DATE -> VARCHAR2

Pentru alte conversii există funcțiile:

<b>TO_CHAR</b> (nchar   clob   nclob)	Returnează VARCHAR2	
<b>TO_CHAR</b> (n [, fmt])	Convertește n la VARCHAR2 conform formatului specificat fmt	SELECT TO_CHAR('01110' + 1) FROM dual;
<b>TO_CHAR</b> ({ datetime   interval } [, fmt ])	Convertește datetime la VARCHAR2 conform formatului specificat fmt	SELECT TO_CHAR (SYSDATE, 'Day, Month, DD, YYYY') "TO_CHAR example" FROM DUAL;
<b>TO_DATE</b> (char [, fmt ])		SELECT TO_DATE('January 26, 1996, 12:38 A.M.', 'Month dd YYYY HH:MI A.M.') FROM DUAL;
<b>TO_NUMBER</b> (char [, fmt ])		UPDATE EMP SET SAL = SAL + TO_NUMBER('100.52', '9,999.9 9') WHERE ENAME = 'BLAKE';
<b>CAST</b> (expr AS type_name)		SELECT CAST(CURRENT_TIMESTAMP AS VARCHAR(30)) FROM DUAL;
<b>CONVERT</b> (char, dest_char_set [, source_char_set ])	Convertește șirul char într-o nouă codificare. Codificările uzuale sunt: -US7ASCII: US 7-bit ASCII character set -WE8DEC: West European 8-bit character set -F7DEC: DEC French 7-bit character set -WE8EBCDIC500: IBM West European EBCDIC Code Page 500 -WE8ISO8859P1: ISO 8859-1 West European 8-bit character set -UTF8: Unicode 4.0 UTF-8 Universal character set, CESU-8 compliant -AL32UTF8: Unicode 4.0 UTF-8 Universal character set	SELECT CONVERT('Ä Ê Í Ó Ø Å B C D E ', 'US7ASCII', 'WE8ISO8859P1') FROM DUAL;

Pentru formatarea datelor calendaristice, in cadrul functiei to\_char:

[http://docs.oracle.com/cd/B19306\\_01/server.102/b14200/sql\\_elements004.htm#i34924](http://docs.oracle.com/cd/B19306_01/server.102/b14200/sql_elements004.htm#i34924)

## Alte funcții

<b>USER</b>	Returnează numele schemei curente	SELECT USER FROM DUAL;
<b>NVL(expr1, expr2)</b>	Dacă expr1 este NULL returnează expr2, altfel returnează expr1	SELECT ename, NVL(TO_CHAR(COMM), 'NOT APPLICABLE') "COMMISSION" FROM emp WHERE deptno = 30;
<b>NULLIF(expr1, expr2)</b>	Dacă expr1 este egală cu expr2 returnează NULL, altfel returnează expr1	SELECT e.last_name, NULLIF(e.job_id, j.job_id) "Old Job ID" FROM employees e, job_history j WHERE e.employee_id = j.employee_id ORDER BY last name;
<b>DECODE(expr, search1, result1 [, search2, result2]... [, default ] )</b>	Compară expr cu fiecare valoare search pe rând. Dacă obține egalitate returnează valoarea result corespunzătoare. Dacă nici o potrivire nu e găsită este returnată valoarea DEFAULT; dacă aceasta nu e precizată returnează NULL. Dacă expr e NULL, se consider egalitate doar dacă search e NULL.	SELECT DECODE (deptno, 10, 'ACCOUNTING', 20, 'RESEARCH', 30, 'SALES', 40, 'OPERATIONS', 'NONE') FROM DEPT;
<b>LEAST(expr1 [, expr2 ]... )</b>	Returnează cel mai mic element din listă	SELECT LEAST ('HARRY', 'HARRIOT', 'HAROLD') "LEAST" FROM DUAL;
<b>GREATEST (expr1 [, expr2 ]... )</b>	Returnează cel mai mare element din listă	SELECT GRETEST ('HARRY', 'HARRIOT', 'HAROLD') "GREATEST" FROM DUAL;