# **ORACLE SQL row Functions**

The following tables list the most commonly used Oracle built-in functions These functions can be used in either of the following:

- 1. SELECT clause of a select statement
- 2. WHERE clause of any statement (select, update, delete)
- 3. CONSTRAINT clause where creating or altering a table

For further details regarding these functions and for the rest of the Oracle built-in functions refer to  $\underline{\text{SQL}}$  reference

#### **Numeric Functions and Parameters**

Function	Meaning & Example
ABS(number)	Gives absolute value
	ABS(actual_amt) > 10000
GREATEST(value1,value2,)	Returns the largest of the values in the list
	Note: This function is used for multiple values in the same row. See the MAX
	function if you want the largest value from a group of rows.
	GREATEST(15, -25, 47) -> 47
LEAST(value1,value2,)	Returns the smallest of the values in the list
	Note: This function is used for multiple values in the same row. See the MIN
	function if you want the smallest value from a group of rows.
	LEAST(15, -25, 47) -> -25
ROUND(number, digits)	Rounds a value to the specified number of digits (decimal places)
	ROUND(123.456,2) -> 123.46
	ROUND(123.456,0) -> 123
	ROUND(234567.00,-3) -> 235000
TRUNC(number, digits)	Cuts off a value at the specified number of DIGITS (decimal places)
	TRUNC(123.456,2) -> 123.45
	TRUNC(123.456,0) -> 123
	TRUNC(234567.00,-3) -> 234000

# **String Functions and Parameters**

Function	Meaning & Example
String_a    string_b	Concatenates string_b to string_a
	Can be used with many arguments
	Ename    ' salary is '    sal    ' per month.'
INITCAP(string)	Converts a string to initial capital letters
	INITCAP('informatics') -> 'Informatics'
LENGTH(string)	Returns the number of characters in a string
	LENGTH('Computer Science') -> 16
LOWER(string)	Converts a string to all lowercase characters
	LOWER('Computer Science') -> 'computer science'
SUBSTR( string,	Extracts a portion of a string
starting value,	Note: If the starting value is 0, it is treated as 1. If the starting-value is
number of chars)	negative, Oracle counts backward from the end of the string. If the starting
	value is positive, Oracle counts forward from the beginning of the string.
	SUBSTR('ABCDEF',0,2) -> 'AB'
	SUBSTR('ABCDEF',2,3) -> 'BCD'
	SUBSTR('abcdef',-4,3) -> 'cde'
UPPER(string)	Converts a string to all uppercase characters
	UPPER('Computer Science') -> 'COMPUTER SCIENCE'
	WHERE UPPER(loc) LIKE '%BOSTON%

# **Date Functions and Parameters**

Function	Meaning & Example
ADD_MONTHS (date, months)	Adds the specified number of months to the date value (subtracts months if
	the number of months is negative)
	Note: Number of months should be an integer
	ADD_MONTHS('15-FEB-2003', 12) -> '15-FEB-2004'
LAST_DAY(date)	Returns the last day of the month that contains the date
	LAST_DAY('15-FEB-2000') -> '29-FEB-2000'
	LAST_DAY('15-FEB-2001') -> '28-FEB-2001'
MONTHS_BETWEEN( date1,	Returns the difference between two dates expressed as whole and fractional
date2)	months
	Note: If date1 is earlier than date2, the result is negative. The result also
	takes into account time differences between the two values.
	MONTHS_BETWEEN('02-FEB-2001','01-JAN-2001') -> 1.03225806
NEXT_DAY(date, day name)	Returns the date of the first day of the specified name that is later than the
	date supplied
	NEXT_DAY('17-OCT-2005','FRIDAY')-> '21-OCT-2005'
ROUND (datetime, format)	Returns the date-time rounded to the unit specified by the format, or to the
	nearest day if no format is supplied
	ROUND('27-OCT-1999', 'YEAR') -> '01-JAN-2000'
SYSDATE	SYSDATE Returns the current date-time from the server where the database
	is located
	SYSDATE + 1 -> Tomorrow same time
TRUNC(datetime)	Removes the time component from a date-time value
	TRUNC(SYSDATE) + 1 -> Tomorrow

# **Conversion Functions and Parameters**

Function	Meaning & Examble
TO_CHAR(date, format)	Converts a date to a string in the specified format
	TO_CHAR('28-MAR-1987', 'Month DD, YYYY') -> 'March 28, 1987'
TO_CHAR(number, format)	Converts a number to a string in the specified format
	TO_CHAR(15687,'\$9G999G999') -> \$15,687
	TO_CHAR(15687,'\$9G999G999D00') -> \$15,687.00
TO_DATE(string, dateformat)	Converts a string to a date using the specified format
	TO_DATE('01/02/1999', 'DD/MM/YYYY')
TO_NUMBER(string, format)	Converts a string to a number using the optional format if specified
	TO_NUMBER('100.00','9G999D99')
	TO_NUMBER(TO_CHAR(sysdate, 'YYYY'))

# **Other Miscellaneous Functions and Parameters**

Function	Meaning & Example
NVL(expr1, expr2)	Replaces null (returned as a blank) with a string in the results of a query. If <i>expr1</i> is null, then NVL returns <i>expr2</i> . If <i>expr1</i> is not null, then NVL returns
	expr1.
	sal + nvl(comm, 0)
DECODE( expr,	Compares <i>expr</i> to each <i>search</i> value one by one. If <i>expr</i> is equal to a
search1, result1,	search, then Oracle Database returns the corresponding result. If no match
search2, result2,	is found, then Oracle returns default. If default is omitted, then Oracle
	returns null.
default)	DECODE (office_id,
	1, 'Southlake',
	2, 'San Francisco',
	3, 'New Jersey',
	4, 'Seattle',
	'Overseas')