TO_CHAR Function

Description

The Oracle/PLSQL TO_CHAR function converts a number or date to a string.

Syntax

The syntax for the TO_CHAR function in Oracle/PLSQL is:
TO_CHAR(value [, format_mask] [, nls_language])

Parameters or Arguments

value
A number or date that will be converted to a string.

format_mask
Optional. This is the format that will be used to convert *value* to a string.

nls_language
Optional. This is the nls language used to convert *value* to a string.

Returns

The TO_CHAR function returns a string value.

Applies To

The TO_CHAR function can be used in the following versions of Oracle/PLSQL:

- Oracle 12c, Oracle 11g, Oracle 10g, Oracle 9i, Oracle 8i

Example

Let us look at some Oracle TO_CHAR function examples and explore how to use the TO_CHAR function in Oracle/PLSQL.

Examples with Numbers

For example:
The following are number examples for the TO_CHAR function.

```
TO_CHAR(1210.73, '9999.9')
Result: ' 1210.7'

TO_CHAR(-1210.73, '9999.9')
Result: '-1210.7'

TO_CHAR(1210.73, '9,999.99')
Result: ' 1,210.73'

TO_CHAR(1210.73, '$9,999.00')
Result: ' $1,210.73'

TO_CHAR(21, '000099')
Result: ' 000021'
```

Examples with Dates

The following is a list of valid parameters when the TO_CHAR function is used to convert a date to a string. These parameters can be used in many combinations.

Parameter	Explanation
YEAR	Year, spelled out
YYYY	4-digit year
YYY YY Y	Last 3, 2, or 1 digit(s) of year.
IYY IY I	Last 3, 2, or 1 digit(s) of ISO year.
IYYY	4-digit year based on the ISO standard
Q	Quarter of year (1, 2, 3, 4; JAN-MAR = 1).
MM	Month (01-12; JAN = 01).
MON	Abbreviated name of month.
MONTH	Name of month, padded with blanks to length of 9 characters.
RM	Roman numeral month (I-XII; JAN = I).
WW	Week of year (1-53) where week 1 starts on the first day of the year and continues to the seventh day of the year.
W	Week of month (1-5) where week 1 starts on the first day of the month and ends on the seventh.
IW	Week of year (1-52 or 1-53) based on the ISO standard.
D	Day of week (1-7).
DAY	Name of day.
DD	Day of month (1-31).
DDD	Day of year (1-366).
DY	Abbreviated name of day.
J	Julian day; the number of days since January 1, 4712 BC.
HH	Hour of day (1-12).
HH12	Hour of day (1-12).
HH24	Hour of day (0-23).
MI	Minute (0-59).
SS	Second (0-59).
SSSSS	Seconds past midnight (0-86399).

Parameter	Explanation
FF	Fractional seconds.

The following are date examples for the TO_CHAR function.
TO_CHAR(sysdate, 'yyyy/mm/dd')
Result: '2003/07/09'

TO_CHAR(sysdate, 'Month DD, YYYY')
Result: 'July 09, 2003'

TO_CHAR(sysdate, 'FMMonth DD, YYYY')
Result: 'July 9, 2003'

TO_CHAR(sysdate, 'MON DDth, YYYY')
Result: 'JUL 09TH, 2003'

TO_CHAR(sysdate, 'FMMON DDth, YYYY')
Result: 'JUL 9TH, 2003'

TO_CHAR(sysdate, 'FMMon ddth, YYYY')
Result: 'Jul 9th, 2003'

You will notice that in some TO_CHAR function examples, the *format_mask* parameter begins with "FM". This means that zeros and blanks are suppressed. This can be seen in the examples below.

TO_CHAR(sysdate, 'FMMonth DD, YYYY')
Result: 'July 9, 2003'

TO_CHAR(sysdate, 'FMMON DDth, YYYY')
Result: 'JUL 9TH, 2003'

TO_CHAR(sysdate, 'FMMon ddth, YYYY')
Result: 'Jul 9th, 2003'

The zeros have been suppressed so that the day component shows as "9" as opposed to "09".

TO_DATE Function

TO_DATE() function in most SQL database management servers such as PostgreSQL and ORACLE is used to convert data values of character data types such as VARCHAR, NVARCHAR, CHAR etc. to standard DATE data type. The function takes two arguments, first the value of the character data type that has to be converted and second the datetime format in which the first argument has been written.

Syntax and Parameters

The basic syntax for using the above mentioned date conversion function is as follows:

to_date(text, datetime format);

The syntax for CONVERT() function in SQL server is as follows :

CONVERT(datetime, text);

The syntax for STR_TO_DATE() function in MYSQL is as follows :

STR_TO_DATE(text, datetime format);

Parameters:

- Text: Data value in character data types like char, text, varchar, nchar, varchar, etc. that has to be converted into date time format.
- Datetime format: The specific format based on date specifiers in which the mentioned text is written.

```
SELECT to_date('20200526','YYYYMMDD');
SELECT to_date('2020-JAN-15', 'YYYY-MON-DD');
SELECT TO_DATE('070920', 'MMDDYY');
SELECT TO_DATE('2020-05-26 13:27:18', 'YYYY-MM-DD HH24:MI:SS');
```

TO_NUMBER Function

This Oracle tutorial explains how to use the Oracle/PLSQL **TO_NUMBER function** with syntax and examples.

Description

The Oracle/PLSQL TO_NUMBER function converts a string to a number.

Syntax

The syntax for the TO_NUMBER function in Oracle/PLSQL is:
TO_NUMBER(string1 [, format_mask] [, nls_language])

Parameters or Arguments

string1

The string that will be converted to a number.

format_mask

Optional. This is the format that will be used to convert *string1* to a number.

nls_language

Optional. This is the nls language used to convert *string1* to a number.

Returns

The TO_NUMBER function returns a numeric value.

Applies To

The TO_NUMBER function can be used in the following versions of Oracle/PLSQL:

- Oracle 12c, Oracle 11g, Oracle 10g, Oracle 9i, Oracle 8i

Example

Let's look at some Oracle TO_NUMBER function examples and explore how to use the TO_NUMBER function in Oracle/PLSQL.
For example:
TO_NUMBER('1210.73', '9999.99')
Result: 1210.73

TO_NUMBER('546', '999')
Result: 546

TO_NUMBER('23', '99')
Result: 23

DATE FUNCTIONS ()

Function	Example	Result	Description
ADD_MONTHS	ADD_MONTHS(DATE '2016-02-29', 1)	31-MAR-16	Add a number of months (n) to a date and return the same day which is n of months away.
CURRENT_DATE	SELECT CURRENT_DATE FROM dual	06-AUG-2017 19:43:44	Return the current date and time in the session time zone
CURRENT_TIMESTAMP	SELECT CURRENT_TIMESTAMP FROM dual	06-AUG-17 08.26.52.742000000 PM **-07:00**	Return the current date and time with time zone in the session time zone
DBTIMEZONE	SELECT DBTIMEZONE FROM dual;	-07:00	Get the current database time zone
EXTRACT	EXTRACT(YEAR FROM SYSDATE)	2017	Extract a value of a date time field e.g., YEAR, MONTH, DAY, … from a date time value.
LAST_DAY	LAST_DAY(DATE '2016-02-01')	29-FEB-16	Gets the last day of the month of a specified date.
MONTHS_BETWEEN	MONTHS_BETWEEN(DATE '2017-07-01', DATE '2017-01-01')	6	Return the number of months between two dates.
NEXT_DAY	NEXT_DAY(DATE '2000-01-01', 'SUNDAY')	02-JAN-00	Get the first weekday that is later than a specified date.
ROUND	ROUND(DATE '2017-07-16', 'MM')	01-AUG-17	Return a date rounded to a specific unit of measure.
SYSDATE	SYSDATE	01-AUG-17	Return the current system date and time of the operating system where the Oracle Database resides.

Function	Example	Result	Description
TRUNC	TRUNC(DATE '2017-07-16', 'MM')	01-JUL-17	Return a date truncated to a specific unit of measure.