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). |
| 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'

## Oracle TO\_NUMBER Format Mask

The Oracle TO\_NUMBER format mask for this function must be a number format. The possible values for number formats you can use are:

|  |  |
| --- | --- |
| **Format** | **Explanation** |
| 9 | Single number character. Removes leading and trailing zeroes. Includes a leading hyphen (-) for negative numbers. |
| 0 | Single number character. Includes leading and trailing zeroes. |
| FM | Indicates that the format should supress all leading and trailing blanks |
| $ | Include a leading $ sign |
| B | Changes the ‘0’ format and replace leading 0’s with blank spaces. |
| S | Include poth positive (+) and negative (-) signs |
| PR | Indicate sign of number with angle brakcets (<>) |
| MI | Indicate the minus sign |
| RN or rn | Displays the number in Roman numerals (upper or lower case) |
| D | Indicates the position of decimal point (.) |
| G | Adds a group separator (often a comma) in a position |
| , | Adds a comma in a position |
| . | Adds a period in a position |
| C | Adds the ISO currency symbol |
| L | Adds the local currency symbol |
| EEEE | Displays the value using scientific notation |
| TM | Displays the Text Minimum in TM9(the default) or TME (Scientific notation) |
| U | Displays in the specified position the Euro (or other) dual currency symbol. |
| V | Displays the value multiplied by 10n , where n is the the number of 9’s after the letter V. The value may be rounded up if necessary |
| X | Returns the hexadecimal value of the specified number of digits. |