

## STANDARD NUMERIC FORMATS

`FORMAT(1234.9, 'C')=$1,234.90`

The "C" (or currency). A currency value.

`FORMAT(-1234, 'D6')=-001234`

The "D" (or decimal). Integer digits with optional negative sign.

`FORMAT(12345.67, 'E')=1.234567E+004`

The exponential ("E"). Exponential notation.

`FORMAT(12.1, 'F')=12.10`

The fixed-point ("F"). Integral and decimal digits with optional negative sign.

`FORMAT(123.4, 'G')=123.4`

The general ("G"). The most compact of either fixed-point or scientific notation.

`FORMAT(12345.67, 'N')=12,345.67`

The numeric ("N"). Integral and decimal digits, group separators, and a decimal separator with optional negative sign.

`FORMAT(0.671, 'P')=67.10%`

The percent ("P"). Number multiplied by 100 and displayed with a percent symbol.

`FORMAT(PI(), 'R')=3.1415926535897931`

The round-trip ("R"). A string that can round-trip to an identical number.

`FORMAT(234, 'X')=EA`

The hexadecimal ("X"). A hexadecimal string.

## CUSTOM NUMERIC FORMATS

`FORMAT(12.78, '0')=13`

Replaces the zero with the corresponding digit if one is present; otherwise, zero appears in the result string.

`FORMAT(21.35, '#.0')=21.4`

Replaces the "#" symbol with the corresponding digit if one is present; otherwise, no digit appears in the result string.

`FORMAT(5, '0.00')=5.00`

Determines the location of the decimal separator in the result string.

`FORMAT(21.35, '0%')=2135%`

Multiplies a number by 100 and inserts a localized percentage symbol in the result string.

`FORMAT(2147483647, '#,##')=2,147,483,647`

Serves as both a group separator and a number scaling specifier. As a group separator, it inserts a localized group separator character between each group. As a number scaling specifier, it divides a number by 1000 for each comma specified.

`FORMAT(1000, '0.00E+00')=1.00E+03`

Scientific format. If the format expression contains at least one digit placeholder (0 or #) to the left of E-, E+, e-, or e+, the number is displayed in scientific format and E or e is inserted between the number and its exponent. The number of digit placeholders to the left determines the number of digits in the exponent.

`FORMAT(1000, '0.00E-0')=1.00E3`

+ - \$ () Literal characters. These characters are displayed exactly as typed in the format string. To display a character other than one of those listed, precede it with a backslash (\) or enclose it in double quotation marks ("").

`FORMAT(21.35, '.\0')=210`

Causes the next character to be interpreted as a literal rather than as a custom format specifier.

`FORMAT(100, "#degrees")=100degrees`

Indicates that the enclosed characters should be copied to the result string unchanged.

`FORMAT(14.365, '#0;0#')=14`

Defines sections with separate format strings for positive, negative, and zero numbers.

## STANDARD DATE AND TIME FORMATS

`DECLARE @NOW DATETIME = CONVERT(DATETIME, '2016-05-01 02:07:06.057');`

`FORMAT(@NOW, 'd')=5/1/2016`

Displays a short date format.

`FORMAT(@NOW, 'D')=Sunday, May 01, 2016`

Displays a long date format.

`FORMAT(@NOW, 'f')=Sunday, May 01, 2016 2:07 AM`

Displays a full date/time pattern (short time).

`FORMAT(@NOW, 'F')=Sunday, May 01, 2016 2:07:06 AM`

Displays a full date/time pattern (long time).

`FORMAT(@NOW, 'g')=5/1/2016 2:07 AM`

Displays a general date/time pattern (short time).

`FORMAT(@NOW, 'G')=5/1/2016 2:07:06 AM`

Displays a general date/time pattern (long time).

`FORMAT(@NOW, 'M')=May 01`

Displays a month/day pattern.

`FORMAT(@NOW, 'O')=2016-05-01T02:07:06.0570000`

Displays a round-trip date/time pattern.

`FORMAT(@NOW, 'R')=Sun, 01 May 2016 02:07:06 GMT`

Displays a RFC1123 pattern.

`FORMAT(@NOW, 'S')=2016-05-01T02:07:06`

Displays a sortable date/time pattern.

`FORMAT(@NOW, 't')=2:07 AM`

Displays a short time pattern.

`FORMAT(@NOW, 'T')=2:07:06 AM`

Displays a long time pattern.

`FORMAT(@NOW, 'u')=2016-05-01 02:07:06Z`

Displays a universal sortable date/time pattern.

`FORMAT(@NOW, 'U')=Saturday, April 30, 2016 11:07:06 PM`

Displays a universal full date/time pattern.

`FORMAT(@NOW, 'Y')=May, 2016`

Displays a year month pattern.

## CUSTOM DATE/TIME FORMATS

`DECLARE @NOW DATETIME = CONVERT(DATETIME, '2016-05-01 02:07:06.057');`

`FORMAT(@NOW, 'yyyy/MM/d')=2016/05/1`  
`FORMAT(@NOW, 'yyyy/MM/dd')=2016/05/01`

Displays the day of the month, from 01 through 31.

`FORMAT(@NOW, 'yyyy/MM/ddd')=2016/05/Sun`

Displays the abbreviated name of the day of the week.

`FORMAT(@NOW, 'yyyy/MM/dddd')=2016/05/Sunday`

Displays the full name of the day of the week.

`FORMAT(@NOW, 'hh:mm:ss.f')=02:07:06.0`

`FORMAT(@NOW, 'hh:mm:ss.ff')=02:07:06.05`

`FORMAT(@NOW, 'hh:mm:ss.fff')=02:07:06.057`

Displays the tenths, hundredths of a second and milliseconds in a date and time value.

`FORMAT(@NOW, 'hh:mm:ss.F')=02:07:06`

`FORMAT(@NOW, 'hh:mm:ss.FF')=02:07:06.05`

`FORMAT(@NOW, 'hh:mm:ss.FFF')=02:07:06.057`

Displays if non-zero, the tenths, hundredths of a second and milliseconds in a date and time value.

`FORMAT(@NOW, 'yyyy/MM/dd/g')=2016/05/01/A.D.`

Displays the period or era.

`FORMAT(@NOW, 'h:mm:ss')=2:07:06`

`FORMAT(@NOW, 'hh:mm:ss')=02:07:06`

Displays the hour, using a 12-hour clock from 1/01 to 12.

`FORMAT(@NOW, 'H:mm:ss')=2:07:06 or 14:07:06`

`FORMAT(@NOW, 'HH:mm:ss')=02:07:06 or 14:07:06`

Displays the hour, using a 24-hour clock from 0/00 to 23.

`FORMAT(@NOW, 'hh:m:ss')=02:7:06`

`FORMAT(@NOW, 'hh:mm:ss')=02:07:06`

Displays the minute, from 0/00 through 59.

`FORMAT(@NOW, 'yyyy/M/dd')=2016/5/01`

`FORMAT(@NOW, 'yyyy/MM/dd')=2016/05/01`

Displays the month, from 1/01 through 12.

`FORMAT(@NOW, 'yyyy/MMM/dd')=2016/May/01`

Displays the abbreviated name of the month.

`FORMAT(@NOW, 'yyyy/MMMM/dd')=2016/May/01`

Displays the full name of the month.

`FORMAT(@NOW, 'hh:mm:s')=02:07:6`

`FORMAT(@NOW, 'hh:mm:ss')=02:07:06`

Displays the second, from 0/00 through 59.

`FORMAT(@NOW, 'hh:mm:ss t')=02:07:06 A`

`FORMAT(@NOW, 'hh:mm:ss tt')=02:07:06 AM`

Displays the first character of the AM/PM designator or AM/PM designator.

`FORMAT(@NOW, 'y/MM/dd')=16/05/01`

`FORMAT(@NOW, 'yy/MM/dd')=16/05/01`

Displays the year, from 0/00 to 99.

`FORMAT(@NOW, 'yyy/MM/dd')=2016/05/01`

`FORMAT(@NOW, 'yyyy/MM/dd')=2016/05/01`

`FORMAT(@NOW, 'yyyyy/MM/dd')=02016/05/01`

Displays the year, with a minimum of three, four or five-digit number.

`FORMAT(@NOW, '02:07:06z')=02:07:06 +3`

`FORMAT(@NOW, '02:07:06zz')=02:07:06 +03`

`FORMAT(@NOW, '02:07:06zzz')=02:07:06 +03:00`

Displays hours offset from UTC, with no leading zeros, with a leading zero for a single-digit value and minutes offset from UTC.