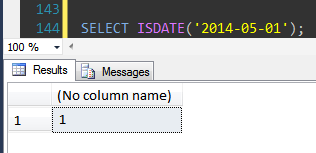
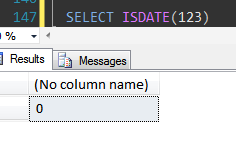
**IsDate Function**

* Returns 1 if the expressionis a valid date. Otherwise, it returns 0.
* The following table summarizes input expression formats that are not valid and that return 0 or an error.

|  |
| --- |
| NULL |
| Values of data types listed in [Data Types](https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql) in any data type category other than character strings, Unicode character strings, or date and time. |
| Values of text, ntext, or image data types. |
| Any value that has a second’s precision scale greater than 3, (.0000 through .0000000...n). ISDATE will return 0 if the *expression* is a datetime2 value, but will return 1 if the *expression* is a valid datetime value. |
| Any value that mixes a valid date with an invalid value, for example 1995-10-1a. |

ISDATE( expression )





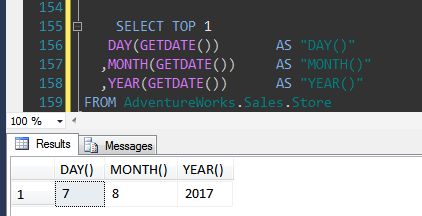
**Day, Month and Year**

* Day - Returns the day of the month (a number from 1 to 31) given a date value.
* Month - Returns the month (a number from 1 to 12) given a date value.
* Year - Returns a four-digit year (as a number) given a date value.

DAY (date value)

MONTH (date value)

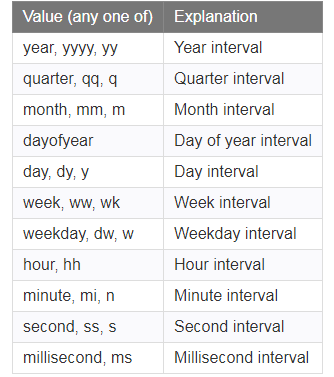
YEAR (date value)

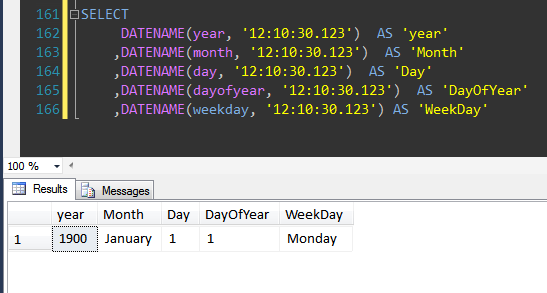


**DateName Function**

* Returns a specified part of a given date, as a string value.
* This function always returns result as nvarchar (character string).

**DATENAME (interval, date)**





**SELECT DATENAME (datepart,'2007-10-30 12:15:32.1234567 +05:10');**

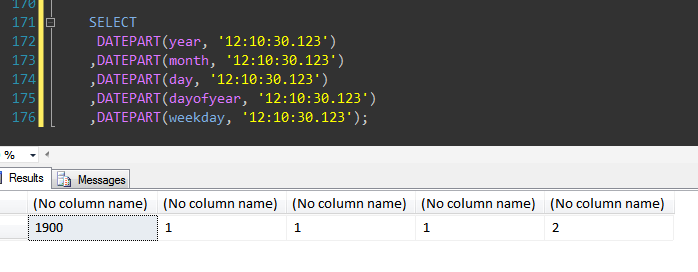
Here is the result set.

| *datepart* | Return value |
| --- | --- |
| **year, yyyy, yy** | 2007 |
| **quarter, qq, q** | 4 |
| **month, mm, m** | October |
| **dayofyear, dy, y** | 303 |
| **day, dd, d** | 30 |
| **week, wk, ww** | 44 |
| **weekday, dw** | Tuesday |
| **hour, hh** | 12 |
| **minute, n** | 15 |
| **second, ss, s** | 32 |
| **millisecond, ms** | 123 |
| **microsecond, mcs** | 123456 |
| **nanosecond, ns** | 123456700 |

**DatePart Function**

* Returns a specified part of a given date, as an integer value.
* Returns the result as an integer value.

**DATEPART (interval, date)**



**SELECT DATEPART(millisecond, '00:00:01.1234567'); -- Returns 123**

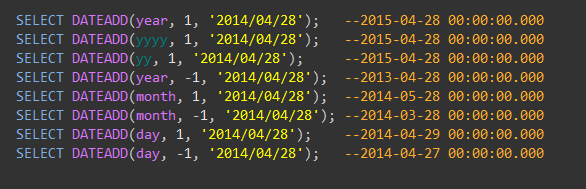
**SELECT DATEPART(microsecond, '00:00:01.1234567'); -- Returns 123456**

**SELECT DATEPART(nanosecond, '00:00:01.1234567'); -- Returns 123456700**

**DateAdd Function**

* Returns a date after which a certain time/date interval has been added.
* If you specify a positive value for the number parameter, the DATEADD function will add the interval to the date.
* If you specify a negative value for the number parameter, the DATEADD function will subtract the interval from the date.
* If you specify a decimal value for the number parameter, the DATEADD function will only use the integer portion of the number (and discard the decimal portion).

**DATEADD( interval, number, date )**



**DateDiff Function**

* Returns the difference between two date values, based on the interval specified.

**DATEDIFF (interval, date1, date2)**

