Determine SQL Server Date and Time Parts with DATEPART and DATENAME Functions

By: Tim Cullen (/sqlserverauthor/1/tim-cullen/) | Updated: 2019-05-24 | Comments (13) | Related: 1 (/sqlservertip/1145/date-and-time-conversions-using-sql-server/) | 2 (/sqlservertip/2655/format-sql-server-dates-with-format-function/) | 3 (/sqlservertip/2507/determine-sql-server-date-and-time-parts-with-datepart-and-datename-functions/) | 4 (/sqlservertip/1616/sql-server-2008-date-and-time-data-types/) | 5 (/sqlservertip/1712/sql-server-function-to-convert-integer-date-to-datetime-format/) | More (/sql-server-developer-resources/) > Dates (/sql-server-tip-category/121/dates/)

Problem

As with most applications and databases, our application retrieves data that has at least one date in the record. There are times when we need to return the name of day or retrieve the month or day from the data. What functions does SQL Server offer to assist in this area? Check out this tip to learn about the SQL Server **DatePart** and **DateName** functions.

Solution

SQL Server offers two functions that help you with retrieving parts of a date: **DATEPART** and **DATENAME**. Both functions require two parameters: the unit of time and date to be queried against.

- **DATEPART** functions returns an integer value
- **DATENAME** function returns a string value - with the DATENAME function, the only units of time that return values different than the DATEPART function are the WEEKDAY and MONTH.

Also, the case of the DatePart and DateName arguments are not case senstive, so you can use either upper case or lower case.

SQL Server DatePart and SQL Server DateName Examples

me examples using these functions which can be used in the WHERE (/sqlservertutorial/136/select-with-where/), HAVING (/) nd ORDER BY (/sqlservertutorial/135/select-with-order-by/) clauses. The examples use data type **datetime2**, but you can also use the **datetime** data type but not get as much precision for some of the date parts. Also, other date data types will work, but some of the datepart options will not work based on the date format.

```
SET NOCOUNT ON

DECLARE @Date datetime2
SET @Date = '2019-09-25  19:47:00.8631597'

SELECT DATEPART(ISO_WEEK,@Date)
SELECT DATEPART(TZoffset,@Date) -- not supported by datetime data type
SELECT DATEPART(NANOSECOND,@Date)
SELECT DATEPART(MICROSECOND,@Date)
SELECT DATEPART(MS,@Date)
SELECT DATEPART(SS,@Date)
SELECT DATEPART(MINUTE,@Date)
SELECT DATEPART(HH,@Date)
SELECT DATEPART(DW,@Date)
SELECT DATEPART(WEEK,@Date)
SELECT DATEPART(DAY,@Date)
SELECT DATEPART(DAYOFYEAR,@Date)
SELECT DATEPART(MM,@Date)
SELECT DATEPART(QUARTER,@Date)
SELECT DATEPART(YYYY,@Date)

SELECT DATENAME(ISO_WEEK,@Date)
SELECT DATENAME(TZoffset,@Date)
SELECT DATENAME(nanosecond,@Date)
SELECT DATENAME(microsecond,@Date)
SELECT DATENAME(millisecond,@Date)
SELECT DATENAME(ss,@Date)
SELECT DATENAME(minute,@Date)
SELECT DATENAME(HOUR,@Date)
SELECT DATENAME(weekday,@Date)
SELECT DATENAME(wk,@Date)
SELECT DATENAME(d,@Date)
SELECT DATENAME(dayofyear,@Date)
SELECT DATENAME(m,@Date)
SELECT DATENAME(quarter,@Date)
SELECT DATENAME(YYYY,@Date)

SET NOCOUNT OFF
```

Here is the output.

DATEPART (@Date value used is '2019-09-25 19:47:00.8631597')			
Unit of time	**DatePart Arguments**	**Query**	**Result**
ISO_WEEK	isowk, isoww, ISO_WEEK	SELECT DATEPART(ISO_WEEK,@Date)	39
TZoffset	tz, TZoffset	SELECT DATEPART(TZoffset,@Date)	0
NANOSECOND	ns, nanosecond	SELECT DATEPART(nanosecond,@Date)	863159700
MICROSECOND	mcs, microsecond	SELECT DATEPART(microsecond,@Date)	863159
MILLISECOND	ms, millisecond	SELECT DATEPART(millisecond,@Date)	863

SECOND	ss, s, second	SELECT DATEPART(ss,@Date)	0
MINUTE	mi, n, minute	SELECT DATEPART(minute,@Date)	47
HOUR	hh, hour	SELECT DATEPART(HOUR,@Date)	19
WEEKDAY	dw, weekday	SELECT DATEPART(weekday,@Date)	4
WEEK	wk, ww, week	SELECT DATEPART(wk,@Date)	39
DAY	dd, d, day	SELECT DATEPART(d,@Date)	25
DAYOFYEAR	dy, y, dayofyear	SELECT DATEPART(dayofyear,@Date)	268
MONTH	mm, m. month	SELECT DATEPART(m,@Date)	9
QUARTER	qq, q, quarter	SELECT DATEPART(quarter,@Date)	3
YEAR	yy, yyyy, year	SELECT DATEPART(YYYY,@Date)	2019
DATENAME (@Date value used is '2019-09-25 19:47:00.8631597')			
Unit of time	**DateName Arguments**	**Query**	**Result**
ISO_WEEK	isowk, isoww, ISO_WEEK	SELECT DATENAME(ISO_WEEK,@Date)	39
TZoffset	tz, TZoffset	SELECT DATENAME(TZoffset,@Date)	+00:00
NANOSECOND	ns, nanosecond	SELECT DATENAME(nanosecond,@Date)	863159700
MICROSECOND	mcs, microsecond	SELECT DATENAME(microsecond,@Date)	863159
MILLISECOND	ms, millisecond	SELECT DATENAME(millisecond,@Date)	863
SECOND	ss, s, second	SELECT DATENAME(ss,@Date)	0
MINUTE	mi, n, minute	SELECT DATENAME(minute,@Date)	47
HOUR	hh, hour	SELECT DATENAME(HOUR,@Date)	19
WEEKDAY	dw, weekday	SELECT DATENAME(weekday,@Date)	Wednesday
WEEK	wk, ww, week	SELECT DATENAME(wk,@Date)	39
DAY	dd, d, day	SELECT DATENAME(d,@Date)	25
DAYOFYEAR	dy, y, dayofyear	SELECT DATENAME(dayofyear,@Date)	268
MONTH	mm, m. month	SELECT DATENAME(m,@Date)	September
QUARTER	qq, q, quarter	SELECT DATENAME(quarter,@Date)	3
YEAR	yy, yyyy, year	SELECT DATENAME(YYYY,@Date)	2019

Build a Calendar Date Part Table

One use for the DATEPART function is if you need to "profile" a calendar year into the various date parts and names. The script below creates a table variable and inserts the various date parts into the table variable:

```
DECLARE @StartDate DATE = '01/01/2011', @EndDate DATE = '12/31/2011'

DECLARE @Dates TABLE (
    CalendarDate DATE PRIMARY KEY
  , MonthNumber TINYINT
  , DateNumber TINYINT
  , DateOfYear SMALLINT
  , WeekNumber TINYINT
  , DayOfWeekNumber TINYINT
  , NameOfMonth VARCHAR(15)
  , NameOfDay VARCHAR(15)
)

WHILE DATEDIFF(DAY,@StartDate,@EndDate) ≥ 0
BEGIN
   INSERT INTO @Dates (CalendarDate, MonthNumber, DateNumber, DateOfYear, WeekNumber, DayOfWeekNumber ,
   SELECT @StartDate
        , DATEPART(MONTH,@StartDate)
        , DATEPART(DAY,@StartDate)
        , DATEPART(DAYOFYEAR,@StartDate)
        , DATEPART(WEEK,@StartDate)
        , DATEPART(DW,@StartDate)
        , DATENAME(MONTH,@StartDate)
        , DATENAME(DW,@StartDate)

   SELECT @StartDate = DATEADD(DAY,1,@StartDate)

END

SELECT * FROM @Dates

SET NOCOUNT OFF
```

Next Steps

- Next time you have a need determine hour, day of week, month, quarter, year, etc. be sure to refer to the DatePart examples in this tip to get you started in the right direction.
- In addition, if you have a need determine week day, month, etc. as a character string be sure to refer back to DateName examples in this tip to get you started in the right direction.
- If you check out all of the options of the DatePart function, you can pass in the hour, day of week, month, quarter, year, etc. parameters and will be returned numeric results as opposed to a character string.
- Check out these related tips:
  - Date and Time Conversions Using SQL Server (/sqlservertip/1145/date-and-time-conversions-using-sql-server/)
  - SQL Server 2008 Date and Time Data Types (/sqlservertip/1616/sql-server-2008-date-and-time-data-types/)
  - SQL Server function to convert integer date to datetime format (/sqlservertip/1712/sql-server-function-to-convert-integer-date-to-datetime-format/)
  - Add and Subtract Dates using DATEADD in SQL Server (/sqlservertip/2509/add-and-subtract-dates-using-dateadd-in-sql-server/)
  - SQL Server DateDiff Example (/sqlservertip/2508/sql-server-datediff-example/)

About the author

(/sqlserverauthor/1/tim-cullen/) Tim Cullen has been working in the IT industry since 2003 and currently works as a SQL Server Reports Developer.

View all my tips (/sqlserverauthor/1/tim-cullen/)

Related Resources

- Date and Time Conversions Using SQL Server... (/sqlservertip/1145/date-and-time-conversions-using-sql-server/)
- Format SQL Server Dates with FORMAT Function... (/sqlservertip/2655/format-sql-server-dates-with-format-function/)
- Determine SQL Server Date and Time Parts with DATE... (/sqlservertip/2507/determine-sql-server-date-and-time-parts-with-datepart-and-datename-functions/)
- SQL Server 2008 Date and Time Data Types... (/sqlservertip/1616/sql-server-2008-date-and-time-data-types/)
- SQL Server function to convert integer date to dat... (/sqlservertip/1712/sql-server-function-to-convert-integer-date-to-datetime-format/)
- More Database Developer Tips... (/sql-server-developer-resources/)