

DATEADD Function

Contents:

- *Basic Usage*
- *Syntax and Arguments*
 - *date*
 - *delta*
 - *date_units*
- *Examples*
 - *Example - DATEADD Function*

Add a specified number of units to a valid date. Units can be any supported Datetime unit (e.g. minute, month, year, etc.). Input must be a column reference containing dates.

NOTE: If this function computes values out of the supported range of dates, the values are written as mismatched values, and the column is likely to be typed as a Datetime column. For more information on supported date ranges, see *Datetime Data Type*.

Basic Usage

```
derive type:single value:DATEADD(myDate, 3, month)
```

Output: Generates a column of values calculating the values in the `myDate` column with three months added to them.

NOTE: Output dates are always formatted with dashes. For example, if the input values include 12/31/2016, a DATEADD function output might be 03-31-2016.

Syntax and Arguments

```
derive type:single value:DATEADD(date,delta,date_units)
```

Argument	Required?	Data Type	Description
date	Y	datetime	Starting date to compare
delta	Y	integer	Number of units to apply to the date value.
date_units	Y	string	String literal representing the date units to use in the comparison

For more information on syntax standards, see *Language Documentation Syntax Notes*.

date

Date values to which to add. It must be a column reference.

Usage Notes:

Required?	Data Type	Example Value
Yes	String (Date column reference)	LastContactDate

delta

Number of units to apply to the date values.

- Negative integer values are accepted.

Usage Notes:

Required?	Data Type	Example Value
Yes	Integer	-3

date_units

Unit of date measurement to which to apply the delta value.


Usage Notes:

Required?	Data Type	Example Value
Yes	String	year

Accepted Value for date units:

- year
- quarter
- month
- dayofyear
- day
- hour
- minute
- second
- millisecond

Examples

 **Tip:** For additional examples, see *Common Tasks*.

Example - DATEADD Function

Source:

Here are some example invoices and their dates. You want to calculate the 90-day and 180-day limits, at which point interest is charged.

InvNum	InvDate
--------	---------

inv0001	1/1/2016
inv0002	7/15/2016
inv0003	12/30/2016

Transform:

```
derive type:single value:DATEADD(InvDate,90,day) as:'plus90'
```

```
derive type:single value:DATEADD(InvDate,180,day) as:'plus180'
```

Results:

i **NOTE:** The output format is always formatted with dashes.

InvNum	InvDate	plus90	plus180
inv0001	1/1/2016	3-31-2016	6-29-2016
inv0002	7/15/2016	10-13-2016	1-11-2017
inv0003	12/30/2016	3-30-2017	6-28-2017