

DateDiff() Function (VB work-a-like)

The following is a DateDiff function for calculating the difference of intervals between two dates.

It is inspired by the [VB function of the same name](#) but is tweaked slightly from the original. The concept is; calculate the number of specified intervals between the two datetimes. Only whole intervals are counted and are specified with a string: yyyy - Year mm - Calendar Month w - Calendar week (7 days) dd - Day h - Hour m - Minute s - Second

If the first date is earlier than the second, the result will be positive otherwise negative.

There is little to no error checking so if you supply nonsense you will probably crash it or get rubbish back.

Note: The dates are subject to UnixTime constraints (no earlier than 01/01/1970). ISO8601 format is not yet supported.

Syntax: =DateDiff(interval,datestr1,datestr2)

Examples: Print DateDiff("mm","01-01-1971 00:00:00","01-06-1980 00:00:00") ' show the number of whole months between the dates InvoiceDays=DateDiff("d",InvDate\$,Now()) ' calculate the number of days since the invoice was raised Print DateDiff("yyyy","28-02-2010 17:00:00","01-01-1970 00:00:00") ' show the number of years between the two dates

Dependencies:

- [UnixTime](#)
- [HumanTime](#)

Code:

```
Function DateDiff(Interval As String,dt1 As String,dt2 As String) As Integer
    'return the number of whole intervals between two dates
    'Result is +ve when DT1<DT2
    'DateDiff ("dd","01/02/2001",Now()) returns the difference in whole
    days since 1st Feb 2001
    Local Integer n,s,t,u,v,x,y,z
    Select Case LCase$(Interval)
        Case "s" ' Seconds
            DateDiff=UnixTime(dt2)-UnixTime(dt1)
        Case "m" ' Minutes
            DateDiff=(UnixTime(dt2)-UnixTime(dt1))\60
        Case "h" ' Hours
            DateDiff=(UnixTime(dt2)-UnixTime(dt1))\3600
        Case "dd" ' Days
            DateDiff=(UnixTime(dt2)-UnixTime(dt1))\86400
        Case "w" ' Weeks
            DateDiff=(UnixTime(dt2)-UnixTime(dt1))\604800
        Case "mm" ' calendar Months
            t=UnixTime(dt1):u=UnixTime(dt2):s=Sgn(u-t)
            If s Then
                dt1=HumanTime(Min(t,u)):dt2=HumanTime(Max(t,u))
```

```
t=Val(Left$(dt1,2)):u=Val(Mid$(dt1,4,2)):v=Val(Mid$(dt1,7,4))
x=Val(Left$(dt2,2)):y=Val(Mid$(dt2,4,2)):z=Val(Mid$(dt2,7,4))
    for n=v+1 to z
        DateDiff=DateDiff+12
    next
    DateDiff=DateDiff+(y-u)
    If t>x then DateDiff=DateDiff-1
EndIf
DateDiff=DateDiff*s
Case "yyyy" ' Years
    t=UnixTime(dt1):u=UnixTime(dt2):s=Sgn(u-t)
    If s Then
        dt1=HumanTime(Min(t,u)):dt2=HumanTime(Max(t,u))
t=Val(Left$(dt1,2)):u=Val(Mid$(dt1,4,2)):v=Val(Mid$(dt1,7,4))
x=Val(Left$(dt2,2)):y=Val(Mid$(dt2,4,2)):z=Val(Mid$(dt2,7,4))
        DateDiff=z-v
        If u>y Then
            DateDiff=DateDiff-1
        ElseIf u=y Then
            If t>x Then DateDiff=DateDiff-1
        EndIf
    EndIf
    DateDiff=DateDiff*s
Case Else
    DateDiff=0
End Select
End Function
```

Bonus! Alternative Day of Week (DoW) function There are a number of Day-of-The-Week (DoW) functions here in this library, but just to flog the idea to death; here is a DoW function using DateDiff.

UT0 was a Thursday, so by calculating the number of days between then and a given date, adjusting for the position of Sunday (assuming Sunday=0) and then MODing by 7 (days in a week), we can derive the DoW for any given datetime - thus:

```
Print datediff("dd","01-01-1970 00:00:00","18-01-2018 00:00:00") mod 7 '
should be zero coz both dates are Thursdays
```

'...bump along for sunday adjustment

```
Print (4+datediff("dd","01-01-1970 00:00:00","01-01-1970 00:00:00")) mod 7 '
test forUT0 = 4 (thursday)
```

'try with today's date...

```
Print (4+datediff("dd","01-01-1970 00:00:00",Now())) mod 7 ' day today
```

If you are using DateDiff() in your code, this has to be about the simplest solution for DoW(). If not, you are better off looking at other examples here in the library as the overhead for DateDiff() is fairly

high and difficult to justify for DoW() alone.

```
Function DoW(d$) As Integer
    DoW=(4+DateDiff("dd","01-01-1970 00:00:00",d$)) mod 7
End Function
```

See Also: [DateAdd\(\)](#)

[Now\(\)](#)

[DatePart\(\)](#)

From:

<http://fruitoftheshed.com/wiki/> - **FotS**

Permanent link:

http://fruitoftheshed.com/wiki/doku.php?id=mmbasic:datediff_function_vb_work_a_like

Last update: **2024/01/19 09:30**

