Title

stata.com

date( ) — Date and time manipulation

Syntax Description Conformability Diagnostics Also see

Syntax

```
    tc = clock(strdatetime, pattern [, year])
    tc = mdyhms(month, day, year, hour, minute, second)
    tc = dhms(td, hour, minute, second)
    tc = hms(hour, minute, second)
  hour = hh(tc)
minute = mm(tc)
second = ss(tc)
    td = dofc(tc)

    tC = Cofc(tc)
    tC = Clock(strdatetime, pattern [, year])
    tC = Cmdyhms(month, day, year, hour, minute, second)
    tC = Cdhms(td, hour, minute, second)
    tC = Chms(hour, minute, second)
  hour = hhC(tC)
minute = mmC(tC)
second = ssC(tC)
    td = dofC(tC)
    tc = cofC(tC)

    td = date(strdate, dpattern [, year])
    td = mdy(month, day, year)
    tw = yw(year, week)
    tm = ym(year, month)
    tq = yq(year, quarter)
    th = yh(year, half)
    tc = cofd(td)
    tC = Cofd(td)
```

td = dofb(*tb*, "*calendar*")
tb = bofd("*calendar*", *td*)

month = month(*td*)
day = day(*td*)
year = year(*td*)
dow = dow(*td*)
week = week(*td*)
quarter = quarter(*td*)
half = halfyear(*td*)
doy = doy(*td*)

ty = yearly(*strydate*, *ypattern* [, *year*])
ty = yofd(*td*)
td = dofy(*ty*)

th = halfyearly(*strhdate*, *hpattern* [, *year*])
th = hofd(*td*)
td = dofh(*th*)

tq = quarterly(*strqdate*, *qpattern* [, *year*])
tq = qofd(*td*)
td = dofq(*tq*)

tm = monthly(*strmdate*, *mpattern* [, *year*])
tm = mofd(*td*)
td = dofm(*tm*)

tw = weekly(*strwdate*, *wpattern* [, *year*])
tw = wofd(*td*)
td = dofw(*tw*)

hours = hours(*ms*)

minutes = minutes(*ms*)

seconds = seconds(*ms*)

ms = msofhours(*hours*)

ms = msofminutes(*minutes*)

ms = msofseconds(*seconds*)

where

tc: number of milliseconds from 01jan1960 00:00:00.000, unadjusted for leap seconds

tC: number of milliseconds from 01jan1960 00:00:00.000, adjusted for leap seconds

strdatetime: string-format date, time, or date/time, e.g., "15jan1992", "1/15/1992", "15-1-1992", "January 15, 1992", "9:15", "13:42", "1:42 p.m.", "1:42:15.002 pm", "15jan1992 9:15", "1/15/1992 13:42", "15-1-1992 1:42 p.m.", "January 15, 1992 1:42:15.002 pm"

pattern: order of month, day, year, hour, minute, and seconds in *strdatetime*, plus optional default century, e.g., "DMYhms" (meaning day, month, year, hour, minute, second), "DMYhm", "MDYhm", "hmMDY", "hms", "hm", "MDY", "MD19Y", "MDY20Yhm"

td: number of days from 01jan1960

tb: business date (days)

calendar: string scalar containing calendar name or %tb format

strdate: string-format date, e.g., "15jan1992", "1/15/1992", "15-1-1992", "January 15, 1992"

dpattern: order of month, day, and year in *strdate*, plus optional default century, e.g., "DMY" (meaning day, month, year), "MDY", "MD19Y"

hour: hour, 0–23

minute: minute, 0–59

second: second, 0.000–59.999 (maximum 60.999 in case of leap second)

month: month number, 1–12

day: day-of-month number, 1–31

year: year, e.g., 1942, 1995, 2008

week: week within year, 1–52

quarter: quarter within year, 1–4

half: half within year, 1–2

dow: day of week, 0–6, 0 = Sunday

doy: day within year, 1–366

ty:	calendar year
strydate:	string-format calendar year, e.g., `"1980"`, `"80"`
ypattern:	pattern of *strydate*, e.g., `"Y"`, `"19Y"`
th:	number of halves from `1960h1`
strhdate:	string-format *hdate*, e.g., `"1982-1"`, `"1982h2"`, `"2 1982"`
hpattern:	pattern of *strhdate*, e.g., `"YH"`, `"19YH"`, `"HY"`
tq:	number of quarters from `1960q1`
strqdate:	string-format *qdate*, e.g., `"1982-3"`, `"1982q2"`, `"3 1982"`
qpattern:	pattern of *strqdate*, e.g., `"YQ"`, `"19YQ"`, `"QY"`
tm:	number of months from `1960m1`
strmdate:	string-format *mdate*, e.g., `"1982-3"`, `"1982m2"`, `"3/1982"`
mpattern:	pattern of *strmdate*, e.g., `"YM"`, `"19YM"`, `"MR"`
tw:	number of weeks from `1960w1`
strwdate:	string-format *wdate*, e.g., `"1982-3"`, `"1982w2"`, `"1982-15"`
wpattern:	pattern of *strwdate*, e.g., `"YW"`, `"19YW"`, `"WY"`
hours:	interval of time in hours (positive or negative, real)
minutes:	interval of time in minutes (positive or negative, real)
seconds:	interval of time in seconds (positive or negative, real)
ms:	interval of time in milliseconds (positive or negative, integer)

Functions return an element-by-element result. Functions are usually used with scalars.

All variables are *real matrix* except the *str** and **pattern* variables, which are *string matrix*.

Description

These functions mirror Stata's date functions; see [D] **datetime**.

Conformability

`clock(`*strdatetime*`,` *pattern*`,` *year*`)`, `Clock(`*strdatetime*`,` *pattern*`,` *year*`)`:

strdatetime:	$r_1 \times c_1$
pattern:	$r_2 \times c_2$ (c-conformable with *strdatetime*)
year:	$r_3 \times c_3$ (optional, c-conformable)
result:	$\max(r_1,r_2,r_3) \times \max(c_1,c_2,c_3)$

`mdyhms(`*month*`,` *day*`,` *year*`,` *hour*`,` *minute*`,` *second*`)`,
`Cmdyhms(`*month*`,` *day*`,` *year*`,` *hour*`,` *minute*`,` *second*`)`:

month:	$r_1 \times c_1$
day:	$r_2 \times c_2$
year:	$r_3 \times c_3$
hour:	$r_4 \times c_4$
minute:	$r_5 \times c_5$
second:	$r_6 \times c_6$ (all variables c-conformable)
result:	$\max(r_1,r_2,r_3,r_4,r_5,r_6) \times \max(c_1,c_2,c_3c_4,c_5,c_6)$

`hms(`*hour*`,` *minute*`,` *second*`)`, `Chms(`*hour*`,` *minute*`,` *second*`)`:

hour:	$r_1 \times c_1$
minute:	$r_2 \times c_2$
second:	$r_3 \times c_3$
result:	$\max(r_1,r_2,r_3) \times \max(c_1,c_2,c_3)$

`dhms(`*td*`,` *hour*`,` *minute*`,` *second*`)`, `Cdhms(`*td*`,` *hour*`,` *minute*`,` *second*`)`:

td:	$r_1 \times c_1$
hour:	$r_2 \times c_2$
minute:	$r_3 \times c_3$
second:	$r_4 \times c_4$ (all variables c-conformable)
result:	$\max(r_1,r_2,r_3,r_4) \times \max(c_1,c_2,c_3c_4)$

`hh(`*x*`)`, `mm(`*x*`)`, `ss(`*x*`)`, `hhC(`*x*`)`, `mmC(`*x*`)`, `ssC(`*x*`)`,

x:	$r \times c$
result:	$r \times c$

`date(`*strdate*`,` *dpattern*`,` *year*`)`:

strdate:	$r_1 \times c_1$
dpattern:	$r_2 \times c_2$ (c-conformable with *strdate*)
year:	$r_3 \times c_3$ (optional, c-conformable)
result:	$\max(r_1,r_2,r_3) \times \max(c_1,c_2,c_3)$

`mdy(`*month*`,` *day*`,` *year*`)`:

month:	$r_1 \times c_1$
day:	$r_2 \times c_2$
year:	$r_3 \times c_3$ (all variables c-conformable)
result:	$\max(r_1,r_2,r_3) \times \max(c_1,c_2,c_3)$

`yw(`*year*`,` *detail*`)`, `ym(`*year*`,` *detail*`)`, `yq(`*year*`,` *detail*`)`, `yh(`*year*`,` *detail*`)`:

year:	$r_1 \times c_1$
detail:	$r_2 \times c_2$ (c-conformable with *year*)
result:	$\max(r_1,r_2) \times \max(c_1,c_2)$

`month(`*td*`)`, `day(`*td*`)`, `year(`*td*`)`, `dow(`*td*`)`, `week(`*td*`)`, `quarter(`*td*`)`, `halfyear(`*td*`)`, `doy(`*td*`)`:

td:	$r \times c$
result:	$r \times c$

`yearly(`*str*`,` *pat*`,` *year*`)`, `halfyearly(`*str*`,` *pat*`,` *year*`)`, `quarterly(`*str*`,` *pat*`,` *year*`)`,
`monthly(`*str*`,` *pat*`,` *year*`)`, `weekly(`*str*`,` *pat*`,` *year*`)`:

str:	$r_1 \times c_1$
pat:	$r_2 \times c_2$ (c-conformable with *str*)
year:	$r_3 \times c_3$ (optional, c-conformable)
result:	$\max(r_1,r_2,r_3) \times \max(c_1,c_2,c_3)$

`Cofc(`*x*`)`, `cofC(`*x*`)`, `dofc(`*x*`)`, `dofC(`*x*`)`, `cofd(`*x*`)`, `Cofd(`*x*`)`, `yofd(`*x*`)`, `dofy(`*x*`)`, `hofd(`*x*`)`, `dofh(`*x*`)`, `qofd(`*x*`)`, `dofq(`*x*`)`, `mofd(`*x*`)`, `dofm(`*x*`)`, `wofd(`*x*`)`, `dofw(`*x*`)`:

x:	$r \times c$
result:	$r \times c$

`dofb(`*tb*`, "`*calendar*`")`

tb:	$r \times c$
calendar:	1×1
result:	$r \times c$

`bofd("`*calendar*`",` *td*`)`

calendar:	1×1
td:	$r \times c$
result:	$r \times c$

`hours(`*x*`)`, `minutes(`*x*`)`, `seconds(`*x*`)`, `msofhours(`*x*`)`, `msofminutes(`*x*`)`: `msofseconds(`*x*`)`:

x:	$r \times c$
result:	$r \times c$

Diagnostics

None.

Also see

[M-4] **scalar** — Scalar mathematical functions