

Datetime values from other software — Date and time conversion from other software[Description](#)[Remarks and examples](#)[Reference](#)[Also see](#)

Description

Most software packages store dates and times numerically as durations from some base date in specified units, but they differ on the base date and the units. In this entry, we discuss how to convert date and time values that you have imported from other packages to Stata dates.

Remarks and examples

stata.com

Remarks are presented under the following headings:

[*Introduction*](#)[*Converting SAS dates*](#)[*Converting SPSS dates*](#)[*Converting R dates*](#)[*Converting Excel dates*](#)[*Example 1: Converting Excel dates to Stata dates*](#)[*Converting OpenOffice dates*](#)[*Converting Unix time*](#)

Introduction

Different software packages use different base dates for storing dates and times numerically. If you are using one of the specialized subcommands for importing data from another package, you do not need to convert your numeric dates after importing them into Stata. `import sas`, `import spss`, and `import excel` will properly convert those dates to Stata dates. However, if you store data from another package into a more general format, like a text file, you will need to do one of two things.

1. If you bring the date variable into Stata as a string, you will have to convert it to a numeric variable.
2. If you import the date variable as a numeric variable, with values representing the underlying numeric values that the other package used, you will have to convert that value to the numeric value for a Stata date.

Below, we discuss the date systems for different software packages and how to convert their date and time values to Stata dates.

Converting SAS dates

If you have data in a SAS-format file, you may want to use the `import sas` command. If the SAS file contains numerically encoded dates, `import sas` will read those dates and properly store them as Stata dates. You do not need to perform any conversion after importing your data with `import sas`.

On the other hand, if you import data originally from SAS that have been saved into another format, such as a text file, dates and datetimes may exist as the underlying numeric values that SAS used. The discussion below concerns converting those numeric values to Stata dates.

SAS provides dates measured as the number of days since 01jan1960 (positive or negative). This is the same coding as used by Stata:

```
. generate statadate = sasdate
. format statadate %td
```

SAS provides datetimes measured as the number of seconds since 01jan1960 00:00:00, assuming 86,400 seconds/day. SAS datetimes do not have leap seconds. To convert to a Stata datetime/c variable, type

```
. generate double statatime = (sastime*1000)
. format statatime %tc
```

It is important that variables containing SAS datetimes, such as `sastime` above, be imported into Stata as doubles.

Converting SPSS dates

If you have data in an SPSS-format file, you may want to use the `import spss` command. If the SPSS file contains numerically encoded dates, `import spss` will read those dates and properly store them as Stata dates. You do not need to perform any conversion after importing your data with `import spss`.

On the other hand, if you import data originally from SPSS that have been saved into another format, such as a text file, dates and datetimes may exist as the underlying numeric values that SPSS used. The discussion below concerns converting those numeric values to Stata dates.

SPSS provides dates and datetimes measured as the number of seconds since 14oct1582 00:00:00, assuming 86,400 seconds/day. SPSS datetimes do not have leap seconds. To convert to a Stata datetime/c variable, type

```
. generate double statatime = (spsstime*1000) + tc(14oct1582 00:00)
. format statatime %tc
```

To convert to a Stata date, type

```
. generate statadate = dofC((spsstime*1000) + tc(14oct1582 00:00))
. format statadate %td
```

Converting R dates

R stores dates as days since 01jan1970. To convert to a Stata date, type

```
. generate statadate = rdate - td(01jan1970)
. format statadate %td
```

R stores datetimes as the number of UTC-adjusted seconds (that is, with leap seconds) since 01jan1970 00:00:00. To convert to a Stata datetime/C variable, type

```
. generate double statatime = rtime - tC(01jan1970 00:00)
. format statatime %tC
```

To convert to a Stata datetime/c variable, type

```
. generate double statatime = cofC(rtime - tC(01jan1970 00:00))
. format statatime %tc
```

There are issues of which you need to be aware when working with datetime/C values; see [Why there are two datetime encodings](#) and [Advice on using datetime/c and datetime/C](#), both in [D] [Datetime conversion](#).

Converting Excel dates

If you have data in an Excel format file, you may want to use the `import excel` command. If the Excel file contains numerically encoded dates, `import excel` will read those dates and properly store them as Stata dates. You do not need to perform any conversion after importing your data with `import excel`.

On the other hand, if you are not using `import excel` and you need to manually convert Excel's numerically encoded dates to Stata dates, you can refer to the discussion below.

Excel has used different date systems across operating systems. Excel for Windows used the "1900 date system". Excel for Mac used the "1904 date system". More recently, Excel has been standardizing on the 1900 date system on all operating systems.

Regardless of operating system, Excel can use either encoding. See <https://support.microsoft.com/kb/214330> for instructions on converting workbooks between date systems.

Converted dates will be off by four years if you choose the wrong date system.

Converting Excel 1900 date-system dates:

Excel's 1900 date system stores dates as days since 31dec1899 (0jan1900), and it treats 1900 as a leap year, although it was not. Therefore, this date system contains the nonexistent day 29feb1900, which is not recognized by Stata. You can see <http://www.cpearson.com/excel/datetime.htm> for more information on how dates and times are handled in Excel.

Because of this behavior, we need to account for that additional day when converting these numerically encoded dates to Stata dates. In other words, to convert Excel dates on or after 01mar1900 to Stata dates, we instead use 30dec1899 as the base.

```
. generate statadate = exceldate + td(30dec1899)
. format statadate %td
```

To convert Excel dates on or before 28feb1900 to Stata dates, we use 31dec1899 as the base. For an example of working with these dates, see the [technical note](#) following [example 1](#).

Stata stores date and datetime values differently, with dates recorded as the number of days since 01jan1960 and datetimes recorded as the number of milliseconds from 01jan1960 00:00:00. However, Excel stores date and time values together in a single number. For datetimes on or after 01mar1900 00:00:00, Excel stores datetimes as days plus fraction of day since 30dec1899 00:00:00, such as dddddd.tttttt. The integer records the days, and the fractional part records the number of seconds from 00:00:00, the beginning of the day, divided by the number of seconds in 24 hours (24*60*60 = 86400).

To convert with a one-second resolution to a Stata datetime, type

```
. generate double statatime = round((exceltime+td(30dec1899))*86400)*1000
. format statatime %tc
```

Converting Excel 1904 date-system dates:

For dates on or after 01jan1904, Excel stores dates as days since 01jan1904. To convert to a Stata date, type

```
. generate statadate = exceldate + td(01jan1904)
. format statadate %td
```

For datetimes on or after 01jan1904 00:00:00, Excel stores datetimes as days plus the fraction of the day since 01jan1904 00:00:00. To convert with a one-second resolution to a Stata datetime, type

```
. generate double statatime = round((exceltime+td(01jan1904))*86400)*1000
. format statatime %tc
```

Example 1: Converting Excel dates to Stata dates

We have some Excel 1900 date-system dates saved in a tab-delimited file. The file contains patients' ID numbers and their dates of birth. The numeric variable `bdate` contains the numeric values that Excel used to store those dates.

```
. clear
. import delimited "exceldates.txt"
(encoding automatically selected: ISO-8859-1)
(2 vars, 3 obs)
. list
```

	patid	bdate
1.	1	33106
2.	2	31305
3.	3	37327

Stata dates measure the number of days since January 1, 1960. For dates on or after March 1, 1900, Excel's base date is December 30, 1899. To convert `bdate` to a Stata date, we need to add the number of days from January 1, 1960, to December 30, 1899 (which is a negative number of days).

```
. generate statadate = bdate + td(30dec1899)
. format statadate %td
. list
```

	patid	bdate	statadate
1.	1	33106	21aug1990
2.	2	31305	15sep1985
3.	3	37327	12mar2002

If you would like to confirm that the conversion has been done properly, you can copy those values of `bdate` into an Excel spreadsheet and format them as dates. You will see the same dates as those listed under `statadate`.

□ Technical note

Suppose we were working with data in Excel that contained dates between January 1, 1900, and February 28, 1900. If we saved these data to a `.txt` or `.csv` file and brought in those numerically encoded dates into Stata, we could not use the conversion function above. The reason these dates are treated differently is that Excel treats 1900 as a leap year, even though it was not; therefore, Excel behaves as if 29feb1900 was an actual date. If you are curious, the purpose of this behavior was to be compatible with a spreadsheet software that was dominant at the time. In short, what this means for us is that if we are working with these particular dates, we need to modify Excel's base date.

Below, we import a text file with dates between January 1, 1900, and February 28, 1900, to demonstrate.

```
. clear
. import delimited "exceldates2.txt"
(encoding automatically selected: ISO-8859-1)
(2 vars, 3 obs)
. list
```

	patid	bdate
1.	1	1
2.	2	15
3.	3	43

Instead of using December 30, 1899, as Excel's base date, as we did previously, we will now use December 31, 1899.

```
. generate statadate = bdate + td(31dec1899)
. format statadate %td
. list
```

	patid	bdate	statadate
1.	1	1	01jan1900
2.	2	15	15jan1900
3.	3	43	12feb1900

Now we have a Stata date recording dates between January 1, 1900, and February 28, 1900. □

Converting OpenOffice dates

OpenOffice uses the Excel 1900 date system described above.

Converting Unix time

Unix time is stored as the number of seconds since midnight, 01jan1970. To convert to a Stata datetime, type

```
. generate double statatime = unixtime * 1000 + mdyhms(1,1,1970,0,0,0)
```

To convert to a Stata date, type

```
. generate statadate = dofc(unixtime * 1000 + mdyhms(1,1,1970,0,0,0))
```

Reference

Gould, W. W. 2011. Using dates and times from other software. *The Stata Blog: Not Elsewhere Classified*. <http://blog.stata.com/2011/01/05/using-dates-and-times-from-other-software/>.

Also see

- [D] [Datetime](#) — Date and time values and variables
- [D] [Datetime business calendars](#) — Business calendars
- [D] [Datetime conversion](#) — Converting strings to Stata dates
- [D] [Datetime display formats](#) — Display formats for dates and times
- [D] [Datetime durations](#) — Obtaining and working with durations
- [D] [Datetime relative dates](#) — Obtaining dates and date information from other dates

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).