**How to Work with Dates Before 1900 in Excel**

If you work with dates prior to 1900, Excel offers little direct help. However, some Excel formulas and a free macro can provide much of the help you need.

By

[Charley Kyd](https://exceluser.com/author/kyd/)

[(Download the workbook.)](https://m2sn50lo.pages.infusionsoft.net)

If you work with dates prior to the year 1900, Excel’s standard date-handling system will be no help. However, there are several ways around this problem.

Excel stores recent dates as a date serial number, which allows us to sort those dates and perform date arithmetic. Unfortunately, Excel’s serial number begins on January 1, 1900; and negative serial numbers aren’t recognized.

(The default date for the Macintosh is 1904. To change the default date, choose Tools, Options, select the Calculation tab, and then select or clear the 1904 date system checkbox.)

To illustrate the 1900 date system, this formula…  
=DATEVALUE(“Jan 1, 1900”)  
…returns the value of 1, which indicates date serial number 1.

On the other hand, this formula for one day earlier…  
=DATEVALUE(“Dec 31, 1899”)  
…returns the #VALUE! error value.

This result is unfortunate, to say the least. Because when you work with dates prior to 1900 you generally will want to do the same two things you do with recent dates. Specifically, you want to sort them, and perform date arithmetic.

Luckily, Excel gives you the power to get around most common problems with early dates. However, you do need to keep the potential problems of early dates in mind.

**Gregorian vs Julian Calendars**

The world uses a different dating system today than we used several hundred years ago. Today, we use the Gregorian calendar; previously most of the Western world used regional variations of the Julian calendar.

One variation seems very strange today. In the British Empire and her colonies, until 1752, Lady Day (March 25) was the first day of the new year.

To make matters worse, different parts of the world converted from Julian to Gregorian at different times over several centuries. As a consequence, ships’ logs could show that they arrived at one port before they had departed their previous port.

To illustrate these regional differences in the United States…

* Texas, Florida, California, Nevada, Arizona, and New Mexico converted with Spain. Last Julian date: Oct 4, 1582. First Gregorian date: Oct 15, 1582.To be clear, Wednesday, Oct. 4, 1582 (Julian) was followed by Thursday, Oct. 15, 1582 (Gregorian).
* The Mississippi Valley converted with France. Last Julian date: Dec 9, 1582. First Gregorian date: Dec 20, 1582.
* Washington, Oregon, and the Eastern Seaboard converted with England. Last Julian date: Sep 2, 1752. First Gregorian date: Sep 14, 1752.

Notice in all cases that certain dates never existed for the three regions. On the Eastern Seaboard, for example, colonists never experienced the period September 3 through September 13, 1752. Also, their year of 1752 began March 25 and ended December 31.

From an Excel perspective, as I will demonstrate, you can generate a date of, say, Sep 10, 1752. But you will need to interpret that date properly within your data.

**Typical Excel Date Problems**

Before we can solve Excel’s date problems, we should list common problems we could have with them.

To illustrate, suppose we want to work with these dates in a spreadsheet:

|  |  |
| --- | --- |
| 21 Nov 2001 Mar 1, 1900 Feb 29, 1900 Feb 28, 1900 Jan 1, 1900 | 1 Sep 1841 Abt 1533 Circa 1492 Feb 29, 1200 Jan 1, 0100 |

Here, we have at least four problems:

1. One of these dates never existed.
2. Most dates are prior to 1900.
3. They use a variety of date formats.
4. Some dates are approximated.

To further complicate the problem, if you enter these dates into a spreadsheet, Excel converts only the recent dates to date serial numbers, giving you the results in column B:

In the range B3:B7, Excel converted the dates to date serial numbers. But the other dates remained as text.

What a mess!

**Fixing Excel’s Date Problems**

Column E shows one useful direction we can head with Excel. In that column, all the original dates have been converted to text values in year-month-day sequence. This allows you to sort them easily.

Column F shows a related solution. There, negative date serial numbers extend to the year 0100. Because each date serial number represents one day, the numbers allow you to perform time calculations easily.

Let’s look at the first row of data in this table:

A3:  ’21 Nov 2001

Here, I typed a single-quote (apostrophe) before I entered the date text. This forced Excel to treat this date as text, rather than as a number.

B3:  11/21/2001

When I typed the text into cell A3, Excel automatically converted the date to the number shown in the table.

C3:  =TEXT(B3,”dd mmm yyyy”)

This formula converts a number in cell B3 to text in the date format shown. If cell B3 already contains text, the formula merely returns that text.

D3:  =ISERROR(DATEVALUE(LEFT(C3,LEN(C3)-4)&”2000″))

This formula checks that Excel can recognize the month and day values in column C. It does so by replacing the original year with “2000”, and then it uses the [DATEVALUE function](https://support.office.com/en-us/article/datevalue-function-df8b07d4-7761-4a93-bc33-b7471bbff252) to try to generate a date serial number. If this effort fails, the formula returns TRUE; otherwise, it returns FALSE.

E3:  =RIGHT(C3,4)&”-“&IF(D3,”01-01″,  
TEXT(LEFT(C3,LEN(C3)-4)&”2000″,”mm-dd”))

E3:  =DateText(B3)

I offer two versions of this formula. The first version is split into two lines. In the first line, the RIGHT function returns the year. We have a dash. Then, if cell D3 says we have an error, the formula returns “01-01”. (By convention, I convert years beginning with “Abt”, “Circa”, and so on to January 1 of the specified year.)

If cell D3 returns FALSE, there is no error. In that case, the second line of the first formula replaces the original year with the year 2000; then it returns the month and day for that date, using the “mm-dd” format.

The second version of the formula in cell E3 calls a free macro that I introduce below. Both versions return the same results for column E, repeated here:

F3: =DateTextValue(B3)

It might be possible to create a formula in Excel that returns the values shown in column F above. However, the formula would be very complicated. Therefore, I used a simple macro.

To complete the table, copy the range C3:F3 down the columns as shown.

**An Intentional Error with Excel’s DATE Function**

In Excel, dates are numbers, called *date serial numbers*, which are formatted in special ways. In Excel spreadsheets, date serial number 1 represents January 1, 1900. But in VBA, date serial number 1 represents one day earlier: December 31, 1899. Lotus 1-2-3 can be blamed for this one-day discrepancy, because Lotus didn’t handle leap years correctly.

Our calendar adds a 29th day to February in all years evenly divisible by 4, except for those ending in 00. These years receive the extra day only if they are evenly divisible by 400. Therefore, 1600, 2000 and 2400 are leap years but 1700, 1800, 1900 and 2100 are not.

Unfortunately, the designers of Lotus 1-2-3 overlooked these facts and included the date February 29, 1900, in their version the [DATE function](https://support.office.com/en-us/article/date-function-e36c0c8c-4104-49da-ab83-82328b832349). So when Microsoft introduced Excel for Windows 1.0 in 1987, Excel’s designers had to faithfully mimic the incorrect performance of the market-leading spreadsheet, Lotus 1-2-3. Therefore, Excel’s DATE function also includes that non-existent date.

On the other hand, VBA had no such restriction when it was introduced in 1993. It could correctly ignore the non-existent leap day in 1900. VBA’s date serial number begins one day earlier so that it can sync up with Excel’s date serial number on March 1, 1900.

VBA also supports negative date serial numbers that can track dates back to January 1, 0100.

**Use a Macro to Overcome Excel Date Problems**

[Use this link to download a free VBA workbook](https://m2sn50lo.pages.infusionsoft.net) that calculates the results shown in columns E and F above.  Because I’ve heavily commented the macro, its operation should be clear if you’re familiar with VBA.

The workbook also includes a working version of the table shown above, and includes several other tables that illustrate how the two macros interpret a variety of common types of data.

To use the workbook, add worksheets to contain your data. Then use the formulas as shown in the tables. You can delete the sample worksheets if you want.

Alternatively, you can add a Module to an existing workbook then copy the VBA code to it. Then you can use the two VBA functions in that workbook.