Dealing with Dates and Times in Excel and Access

(By Sarah Cohen, based on a FoxPro version of the document by Richard Mullins)

Our convention for dates are based on the names and numbers in our calendar. We take its quirks for granted, as if they're not complications. (February 29th, Thirty days hath September, etc.)

There's no real standard for how to represent, in character format, a date on the calendar.

Here's a date you might find on a drivers' license: **Feb. 14, 1967.**

In a computer file, it might be represented in many ways:

> "14-FEB-67"
> 19670214
> 2/14/67
> 021467

Each of these representations has its flaws.

Many aren't, in nerd-speak, "Year-2000 compliant." For retirement data or anything having to do with projections, that's pretty serious.

But that's not the only problem. All but one of these representations won't sort correctly. There's also a lot of work to be done to get summaries out of these dates.

UNDERSTANDING DATE AND TIME FIELDS

Access and Excel both overcome these problems in the same way. They both display a date or a time that looks like one you recognize. But they are storing something very different: The number of full days since Jan. 1, 1900. They store time as a fraction, or decimal representation, of a day. Here's how a spreadsheet sees different values:

What you see:	What Excel and Access see:
January 23, 1995	34722
12:05 PM	0.503472
23-Jan-95 12:05 PM	34722.503472

This is a powerful way to think of time for three reasons:

1. No matter how you look at your dates and times, they can be sorted in chronological order. Anyone who's tried to sort a text representation of a date, like "2/1/95," knows how powerful that can be.

2. The programs keep information about the date. For instance, they know whether the time is in the afternoon or the morning. They know a date is the 25th of December, no matter what year it is. And they know January 23, 1995 was a Monday.

3. You can do math on the results. This means that you can add a date to a time. Once you do that, you can find out the number of minutes, days, weeks or months that separate the values. So your spreadsheet will be able to calculate the number of

minutes it took for an ambulance to respond to a call, even when the call was made at 11:55 on Saturday night and the response took 20 minutes.

There is one *disadvantage* to using date and time values.

You may notice there's no room for, say, "sometime in March, '79." or "summer". This kind of value requires you know an exact date. In Access, you'll be forced to enter one -- March 1979 isn't good enough. Excel will accept Mar-79 as a date, but will actually use the 1st of the month. So reserve your use of date fields for times in which you either know a date or you don't -- not when you know about when something happened, as in your notes.

Excel limitations:

Excel only recognizes dates beginning in 1900. That's a real problem in dealing with retirement databases. You may want to switch to Access if you have that kind of data. It seems to recognize dates back to the Renaissance.

VIEWING (FORMATTING) DATES AND TIMES

You've probably used "number formats" in Excel before. When you calculated a percent change, for example, Excel displayed the results as a fraction. Then you hit the little "%" button to convert that number into a percentage that looks right. That button applied a built-in format to a number.

You can do the same thing with dates and times. Just like with your percents or with currency formats, none of the information about your date goes away. You're just looking at it in another way.

Excel date formatting

In Excel, you format a number in the Format, Cells, Number dialog box. Choose a date or time format, or type the special formatting codes in the box under Custom. Here are some ways that Excel can display the value 34722.503472:

Date format	What you see
yyyy	1995
mmmm d, yyyy	January 23, 1995
mm/dd/yy	01/23/95
m/d/yy	1/23/95
dddd, mmm d, yyyy hh:mm AM/PM	Monday, Jan 23, 1995 12:05 PM

Access date formatting

As always, Access is more bureaucratic than Excel. Each field can have one and only one format. You set it in the Table Design window for any Date/Time field. Access will suggest some standard formats, or you can use the Excel codes to design your own. Unless you tell it otherwise, you'll have to type dates in whichever way you view them.
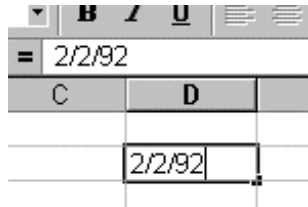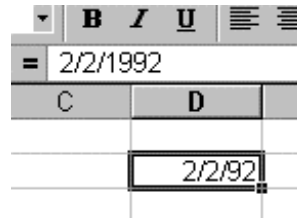
Excel

Just type your date and time values as you normally would. Excel understands most representations of a date or time.

One way to tell if you've typed a date or time it recognizes is to watch what happens after you hit the "Enter" key. If Excel understood a date, it will right-justify it (like a number) and will add digits for century if you didn't type them:

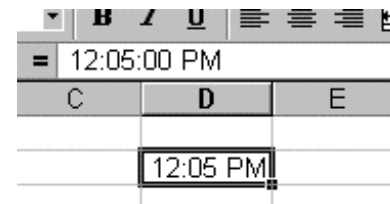By typing this: Excel does this:

The same visual clues work with times. Notice how Excel added in the extra :00 for seconds.

This time, though, you have to be a little more careful how you type them. You can't use any periods in the AM/PM indicators, and there must be a space between the time and, say, PM. Try sorting if you're not sure you did it right.

Access

In Tables:

For Access, you must format your field before you type, then type in whatever way you've told it to expect. Format the field in the Table Design view of your data, changing the default "Text" to "Date/Time", then choosing your format. You'll usually use separate fields for times and dates.

In Queries:

Surround something that looks like a date with the pound sign (#). Access sees the following enteries as the same thing:
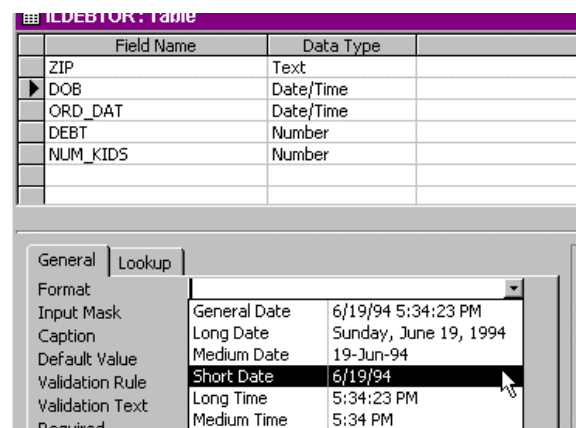
#January 1, 1974#
#1/1/74#
#1-Jan-74#

This is the equivalent of putting quotation marks around a string of text.

Use the same approach with time in a criteria: Between #6:00 AM# and #4:00 PM#

If a field is consistently a date or a time, you may be able to import it correctly. Be aware, though, that some data can be lost and you can get very odd results when you do.

The most serious problem with importing dates comes from the way Access and Excel treat dates that don't specify the century. Any year from 00 to 30 will become 2020 to 2029, not 1920 to 1929. That's a problem for us, since virtually none of the databases we use right now have the 2000s in them.

On the other hand, times we get from a mainframe computer are often missing the colon separating hour from minute.

These two problems lead many reporters to import their dates and times as text. They convert them to dates later on. (See "Converting dates and times" later in this handout.)

If you choose to try, here are some tips:

Excel

Remember that last window in the Text Import Wizard -- the one you always blow right by? Well take a minute look at it. You can choose "Date" as one of your types of data columns. The crucial element here is that you get the order of the month, day, and year correct in this screen. Use the drop-down menu to change it:
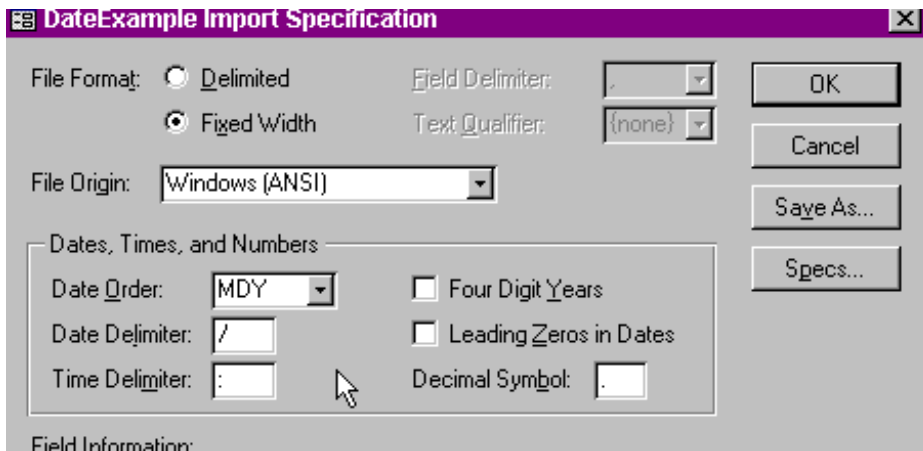
The key thing to tell Excel is how the date begins: Which comes first? The Year, the month or the day? Choose the drop-down menu to change it. Once we've successfully told it how to import, Excel treats them all the same.
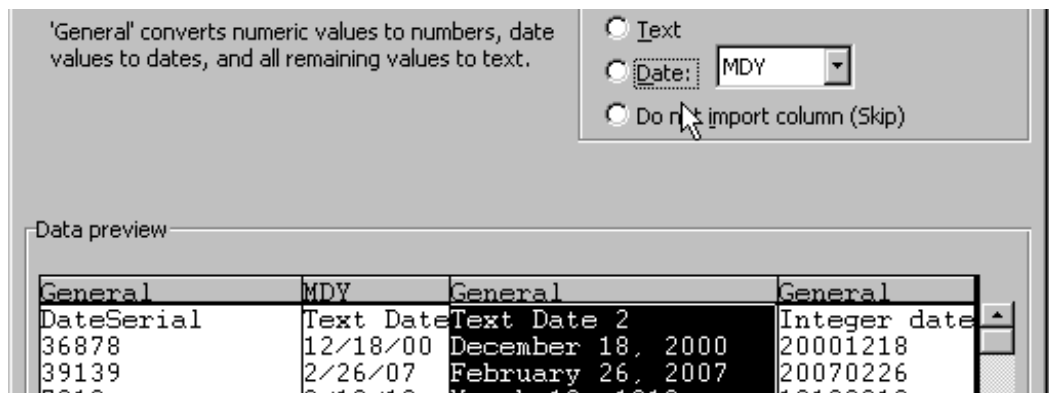
Access:

You get considerably more control over the way that Access reads dates. It asks you not only what comes first, but whether you have a "/", a "-" or something else separating the pieces of a date. You also tell it whether there are leading zeroes in your data, four-digit years, etc. It will also let you give another time delimiter.

You'll find this screen by choosing the "Advanced" options of the Import Wizard.

DATE FUNCTIONS

The key to the power of dates and times is in the functions you can use to manipulate them. The main ones are straightforward. They are the same in Excel and Access:

FUNCTIONS TO GET PIECES OF A DATE:

YEAR(Date) =   The year in the date.

MONTH(Date) =   The month of the date, as the number 1-12.

DAY(Date) =   The day of the month (1-31)

WEEKDAY(Date, Start Day) = The day of the week (1-7)

> The day of the week. If you leave Start Day blank, the week begins with the first day on Sunday. A 2 where you see that begins it on Monday, which is useful if you want to figure out when weekends are.
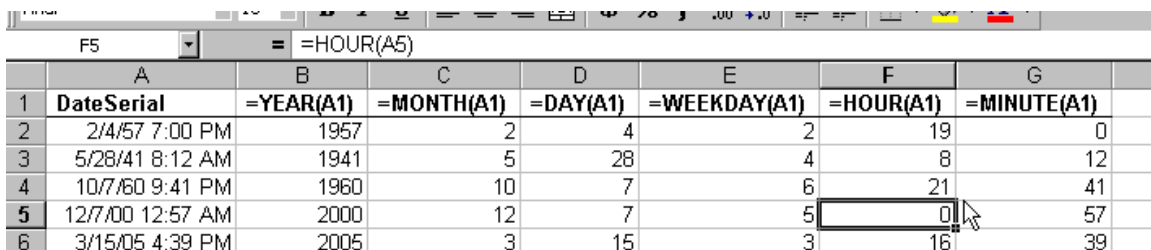
HOUR(Time) =   The hour of the day (0-23)

MINUTE(Time) =   The minute of the hour (0-59)

NOW() in Excel and Access gives you the date and time of your computer.

TODAY() in Excel gives you today's date (on your computer)

DATE() in Access gives you today's date.

There are two key differences between using these functions and formatting the dates the way you'd like to see them:

1. When you create a new field with the month, for example, all of the other information about the date and time goes away. It's simply a month. You sometimes want this. You might want to group values by month or year in a query, something that's very difficult without the function but very easy with it.

2. Formatting will work differently in Access: You can no longer see words instead of numbers about your date. (In Excel, you can.)

FUNCTIONS TO CALCULATE DATES FROM TEXT OR NUMBERS

There are two main ways to convert text values (or numbers) into a date.

1. DATEVALUE("2/4/57"): This takes something that looks like a date and treats it like a date. So if you have something that says, "January 1, 1993", you would put that inside the parentheses.

   You may have to use string functions to create the correct format for this function. It's most useful when you have a word or abbreviation for the month rather than the number. For instance, "2-Dec-57" can be translated into its corresponding date field without having to figure out that Dec is the 12th month.

2. DATE(Year, Month, Day) in numbers: If you have the year, month and day in numbers, then you'd use this instead. In Access, the function is DATESERIAL, not DATE. But it works the same way.

3. TIMEVALUE("12:00 PM"): Again, use it on something that looks like a time. It will accept military time.

… AND BACK TO TEXT AGAIN.

There are two other three other functions you may want to get to know even though they're not date and time functions:

1. TEXT(*number,"format")* in Excel;
   FORMAT(*number, "format"*) in Access:

   Why they use different words for the same thing is baffling. But these two functions do the same thing. You may want to turn a number into words at some point, essentially reversing your import. These do that:

   =TEXT(20855.79,"m/d/yy") gives you back a string rather than a date corresponding: "2/4/57" in Excel.
   Newdate:=FORMAT(20855.79,"yyyymmdd") gives you 19570204 in Access-- an ugly version of a date, but one that will always sort correctly.

2. INT(*number)*: Use this to return everything to the left of the decimal point, converting a date and time to just the date.

   =INT(20855.79) = 20,855.

3. MOD(*number, interval*). (Excel only) This gives you the remainder you may remember from 3rd grade arithmetic -- what's left over after you divide one number into another. The reason this is useful for dates and times is that the remainder of any number after dividing by 1 is the portion to the right of the decimal place:

   =MOD(20855.79, 1) = .79, or 7:00 PM, because that's what's left over when you divide by 1.

This only works in Excel. Access' MOD operator only recognizes integers, so you'll have to work with something a little more complex. Your formula would be: DATE-Int(DATE).

CALCULATING AGES :

Excel

One of the hidden treasures in Excel is the DATEDIF function. DATEDIF(Date1, Date2, type) gives you back a difference between two dates expressed in whatever unit of measure you tell it.

Here is how it works with TODAY() as the second date:

The "y" tells it to express it in years. Excel is smart enough not to round up to the next year -- you must be at or beyond a birthday to go to the next year old.

Replace the "y" with "m" to see how old someone is in months. This is useful in stories about retirement funds, in which you want to know the total number of months until someone's eligible.

Access

Unfortunately, Access has no such beauty in its DATEDIFF function (note the extra "F"). It works off of the year only, meaning that it won't properly calculate ages. You'll have to do some shenanigans to get it to work. One foolproof, but ugly, formula is this:

DateDiff("yyyy",[dob],Now())-IIf(Format(Now(),"mmdd")<Format([dob],"mmdd"),1,0)

This says, figure the difference in years between a field called DOB and today (or NOW()), then subtract a year if today's date is before the date of birth.

The reason for this ugliness is that Access' "DATEDIFF" function isn't as smart as Excel's. It just subtracts out the years as if all dates were Jan. 1. So it'll give you 1 year as an answer even if it's just one day, from Dec. 31 to Jan 1.

If you want something simpler, but not 100% corrected for leap years, try:

(now()-[dob])/365.25

OTHER DATE AND TIME MATH:

Most other time spans can just be calculated by subtraction.

Say you have a 911 call that begins at 11:55 pm and is answered at 12:05 am the next day. That means there's 10 minutes to the response time. Making sure that you have included the date with the time is crucial here. Otherwise you'll have to see which one's bigger. But if you have the date and time together, it's just the later time - the earlier time. Then you format it as [h]:mm (with the brackets to keep the hours from accumulating rather than flipping back over to midnight each day).