

# What are the variables for Date-Time in SPSS?

Authored by  
**stats writer**

June 23, 2024

## RECOMMENDED CITATION

stats writer (2024). *What are the variables for Date-Time in SPSS?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=148982>

The variables for Date-Time in SPSS refer to the data input that records and represents specific dates and times. These variables can be in various formats, such as date, time, or date-time combinations, and can be used to analyze and manipulate temporal data in statistical analysis. They are essential for conducting time-based analyses and can be used to track changes over time, identify trends, and measure the impact of time-related factors on a particular outcome. In SPSS, these variables can be customized and formatted to suit the specific needs of the research, making it a valuable tool for handling temporal data in data analysis.

## Working with Dates and Times in SPSS

Datasets often include variables that denote dates or time. Thus, it is important to know how SPSS treats and works with such variables. In the following sections, we will discuss:

How date-time variables work in SPSS. Standard formats for dates and time. Defining date-time variables in the Variable View window and through the Date and Time Wizard. Setting the century range for two-digit years. Converting a string variable containing a date into an actual date variable. Computing the elapsed time between two date-time variables using the Date and Time Wizard.

## Date-Time Variables in SPSS

In SPSS, date-time variables are treated as a special type of numeric variable. All SPSS date-time variables, regardless of whether they're a date or a duration, are stored in SPSS as the number of seconds since October 14, 1582. This means that "under the hood", date-time variables are actually integers! This might not seem important, but it's what makes it possible to do "date arithmetic", such as computing the elapsed time between two dates, or adding and subtracting units of time from a date.

Fortunately, you as the user do not normally need to interact with the underlying integers, and you can type in data values for date and time variables using normal date-time conventions. However, dates and times can be written using a number of different conventions, so we need a way to tell SPSS how to read and parse our date strings. That's where the concept of *date formats* comes in.

## Standard Formats for Dates and Times

When reading data containing dates or using certain date-time functions, we need to tell SPSS which *date format* to use, so that it knows how to correctly parse the components of the input string. A *format* is a named, pre-defined pattern that tells SPSS how to interpret and/or display different types of variables. There are different formats for different variable types, and each format in SPSS has a unique name.

Date-time formats are used in several situations:

Initializing a new date or duration variable. Converting a variable from string or numeric to date. Changing the display format for an existing date variable (without changing the underlying data).

Your choice of format will depend on whether or not the input is a date or a duration, as well as the time units included in the data value, the order of the units (e.g. month-day-year versus year-month-day), and the presence or absence of delimiters .

## Date Formats

The actual date formats that you will use in your SPSS syntax are as follows.

Date-Time Unit	Format name (general form)	Format name (actual)	Example
dd-mmm-yy	DATEw	DATE9	31-JAN-13
dd-mmm-yyyy	DATEw	DATE11	31-JAN-2013
mm/dd/yy	ADATEw	ADATE8	01/31/13
mm/dd/yyyy	ADATEw	ADATE10	01/31/2013
dd.mm.yy	EDATEw	EDATE8	31.01.13
dd.mm/yyyy	EDATEw	EDATE10	31.01.2013
yyddd	JDATEw	JDATE5	13031
yyyyddd	JDATEw	JDATE7	2013031
yy/mm/dd	SDATEw	SDATE8	13/01/31
yyyy/mm/dd	SDATEw	SDATE10	2013/01/31
q Q yy	QYRw	QYR6	1 Q 13
q Q yyyy	QYRw	QYR8	1 Q 2013
mmm yy	MOYRw	MOYR6	JAN 13
mmm yyyy	MOYRw	MOYR8	JAN 2013
ww WK yy	WKYRw	WKYR8	5 WK 13
ww WK yyyy	WKYRw	WKYR10	5 WK 2013
dd-mmm-yyyy hh:mm	DATETIMEw	DATETIME17	31-JAN-2013 01:02
dd-mmm-yyyy hh:mm:ss	DATETIMEw	DATETIME20	31-JAN-2013 01:02:33

Date-Time Unit	Format name (general form)	Format name (actual)	Example
dd-mmm-yyyy hh:mm:ss.s	DATETIMEw.d	DATETIME23.2	31-JAN-2013 01:02:33.72
yyyy-mm-dd hh:mm	YMDHMSw	YMDHMS16	2013-01-31 1:02
yyyy-mm-dd hh:mm:ss	YMDHMSw	YMDHMS19	2013-01-31 1:02:33
yyyy-mm-dd hh:mm:ss.s	YMDHMSw	YMDHMS19.2	2013-01-31 1:02:33.72
(abbr. name of the day)	WKDAYw	WKDAY3	THU
(full name of the day)	WKDAYw	WKDAY9	THURSDAY
(abbr. name of month)	MONTHw	MONTH3	JAN
(full name of the month)	MONTHw	MONTH9	JANUARY

In the "Date-Time Unit" column, the date components are represented using the following codes:

"d" = day

"dd" = day of month as two-digit number (01, 02, ..., 31)"ddd" = day of year as three-digit number

(1, 2, ..., 365)"m" = month

"mm" = month as two-digit number (01, 02, ..., 12)"mmm" = month as three-character abbreviation

(JAN, FEB, ..., DEC)"y" = year

"yy" = two-digit year (century is omitted)"yyyy" = four-digit year"q" = quarter of year as one-digit

number (1, 2, 3, 4)"ww" = week of year as two-digit number (1, 2, ..., 53)"h" = hour (01, 02, ...

23)"m" = minute (01, 02, ..., 59)"ss" = seconds (01, 02, ..., 59)

In the "general form" column, the name of the format appears first, followed by the letter *w* (or *w.d*).

The letter *w* denotes the number of "columns" (typically the number of characters in the input string), and the letter *d* represents the number of decimal places, if present. You will replace these with the appropriate number to use for the width of the date.

You'll see an example of how date-time formats are used in the example of converting a string variable to a date variable.

## Durations

The actual duration formats that you will use in your SPSS syntax are as follows.

Duration Unit	Format code (general form)	Format code (actual)	Example
mm:ss	MTIMEw	MTIME5	1754:36

Duration Unit	Format code (general form)	Format code (actual)	Example
mm:ss.s	MTIMEw.d	MTIME8.2	1754:36.58
hh:mm	TIMEw	TIME5	29:14
hh:mm:ss	TIMEw	TIME8	29:14:36
hh:mm:ss.s	TIMEw.d	TIME11.2	29:14:36.58
ddd hh:mm	DTIMEw	DTIME9	1 05:14
ddd hh:mm:ss	DTIMEw	DTIME12	1 05:14:36
ddd hh:mm:ss.s	DTIMEw.d	DTIME15.2	1 05:14:36.58

In the "Duration Unit" column, the time components are represented using the following codes:

"ddd" = days as a number greater than or equal to 0  
 "hh" = hours  
 "mm" = minutes  
 "ss" or "ss.s" = seconds

Just as with date formats, the "general form" of the format name contains *w* (or *w.d*). The letter *w* denotes the number of "columns" (typically the number of characters in the input string), and the letter *d* represents the number of decimal places, if present. You will replace these with the appropriate number to use for the width of the date.

Notice how in the column of examples, SPSS took the same underlying data and automatically converted the time units based on the formats we chose. When we used the DTIME format, it knew that 29 hours should "roll over" to 1 day, 5 hours. When we used the MTIME format, it knew that 29 hours, 14 minutes is equal to  $(29 \times 60) + 14 = 1754$  minutes. This is one of the benefits of using date-time variables to represent dates and durations: they give us the option to change how the data is displayed without needing to do the conversion arithmetic ourselves.

Note: As of SPSS version 24, the above date formats will correctly recognize date strings without delimiters *as long as* the lengths of the other elements are correct (i.e., leading zeroes where necessary in the day, month, hour, minute, and second, so that those components are each two characters long). (Source) In previous versions, these date formats would not recognize dates that did not contain the appropriate delimiters.

## Defining Date-Time Variables in the Variable View Tab

It is important to specify which variables in your data are dates/ times so that SPSS can recognize and use these variables appropriately. However, the procedure for defining a variable as date/time depends on its currently defined type (e.g., string, numeric, date/time). The following sections outline how to define a variable as date/time based on the variable's current type.

## Changing a variable type from string or numeric to date/time

If your dataset includes a variable whose values represent dates or time, but the variable is currently defined as string or numeric, you should specify that the variable is actually a date/time. You can specify the variable type as date/time by clicking the **Variable View** tab, locating the variable, and clicking on the cell beneath the "Type" column. A blue "..." button will appear. Clicking the blue "..." button opens the Variable Type window. Select "Date" from the list of variable types. Then, on the right, select the format in which the date/time for that variable should appear (by selecting the date/time format in which the values already appear). Click **OK**. Now SPSS will recognize the variable as date/time.

Note: These steps work only if the variable values are already in a standard date/time format but are currently defined as string/numeric...and only if you define the variable as date/time by selecting a date/time format that already mirrors the existing format. For example, if the values appear as "Aug 1991" you should select a date/time format that mirrors the existing format. If you try to select a format that includes additional or different information, the change in format may fail and blank out the data.

Example: This scenario is likely if you import data from another file source, such as Excel, and SPSS does not immediately define the variable type as date/time, even though the values are in a standard date/time format.

Thus, the following criteria must apply in order to use the steps outlined above:

Your variable values already appear in a standard date/time format. Your variable is currently defined as "string" or "numeric" rather than date/time. You wish to re-define your variable type from string or numeric to date/time.

## Changing the variable type from string or numeric to a date/time format that is different from the date/time format in which the values currently appear

If the variable is already in a standard date/time format but is currently defined as string or numeric, and you wish to both A) define the variable as date/time, and B) choose a different date/time format than the one that matches the current format, you must proceed in two steps.

You must first define the variable as date/time and select the format in which your dates/times currently appear. After you have specified the current format of date/time values for that variable, you can then change the format of the date following the same steps you used to define the variable type and date format during the first step.

Note: If the dates for a selected variable appear as mm/dd/yyyy and are currently defined as

"String" in the "Type" of variable in Variable View, you cannot change the "Type" to "Date" and select the new format in which you want the date/time values to appear. You must first select the format in which the dates/times currently appear. Then, you can repeat this process to select the new format in which you want the dates to appear. If you do not first define a variable as a "Date" and select the current date/time format before selecting the format to which you want to change it, the values for that variable will be defined as missing.

Example: If a variable with date/time values is currently defined as string or numeric, but all the values follow the form mm/dd/yyyy (e.g., 01/31/2013), then you must select this format (mm/dd/yyyy) when you change the variable's type to date/time. Do not select a format that does not match the current format of the values.

Thus, the following criteria must apply in order to use the steps outlined above:

Your variable values already appear in a standard date/time format. Your variable is currently defined as "string" or "numeric" rather than date/time. You wish to re-define your variable type from string or numeric to a date/time format that is different from the date/time format in which the values currently appear.

### Changing variables defined as dates/time from one date format to another date format

If a variable type is already defined as date/time, then changing the format of the values to a different date/time format is simple. In Variable View, under the column "Type," select the cell that corresponds to the variable you want to change. A blue "..." button will appear, which opens the "Variable Type" dialog box. "Date" should already be selected from the list of variable types on the left. On the right, select the new date/time format in which you would like the variable values to appear. Click **OK**. Now click the **Data View** tab to view your data; your dates should now appear in the format you selected.

Note: If you select a new format that includes space for information that does not actually appear in your dataset, it will appear as 0s in the data. For example, if your data only includes information about the month, day, and year, and you select a format that also includes space for the hour, minute, and second, values will appear like this one: 31-JAN-2013 00:00:00.

Example: Perhaps your date is defined as date/time and appears as "01/31/2013," but you would like it to appear as "2013/01/31," instead.

### Setting the Century Range for Two-Digit Years

When writing dates, it's common to see individuals abbreviate the year to two digits, especially in

contexts where the century is "obvious" to the reader. This is fine when making notes to yourself, but when you're trying to compile data for analysis, this can be hugely problematic, especially when working with data that covers a large time range, or is very far in the past.

**In general, we recommend always using four-digit years when entering data for dates.** But sometimes you may not be in control of how the data was entered -- you may receive or request a dataset where the dates only used two-digit years. For these situations, it's important to know how to appropriately define the century range in SPSS.

In SPSS, the *century range* refers to the 100-year range that SPSS will assume when parsing date variables with two-digit years. For example: when you read the date 1/1/80, do you assume that I mean 1/1/1980 or 1/1/2080? If you didn't have any other context clues, you'd probably base your guess on the current year (2020). You might go with the century that makes the two-digit year closer to the current year, which would mean 1/1/1980. Or, you might assume that the century should match the current century, which would mean 1/1/2080.

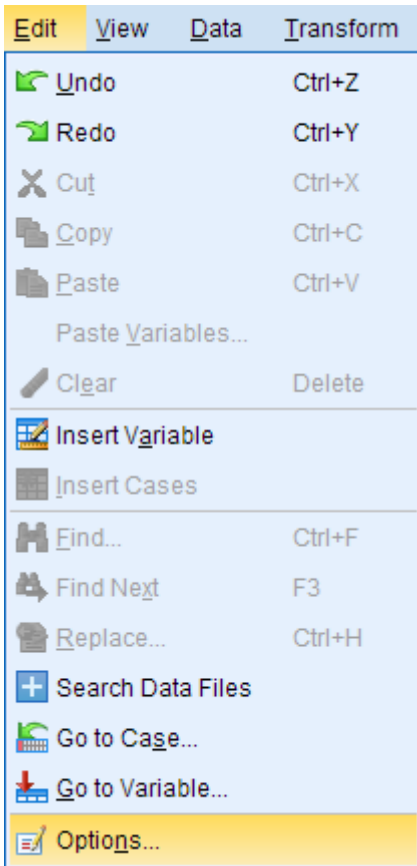
**The default century range in SPSS is based on the current year: it will start the range at 69 years prior to the current year and end the range at 30 years after the current year** ([source](#)). So if you are using SPSS in the year 2020, it will assume that the century range is 1951 to 2050; but if you open SPSS a year later, SPSS will assume that the century range is 1952 to 2051.

**Why does the century range matter?** If you are going to compute elapsed time, or want to use your date variables as a predictor in a model, you can imagine how problematic it would be if one of the dates was off by 100 years! For this reason, it's critical that you specify the appropriate century range when working with dates containing two-digit years.

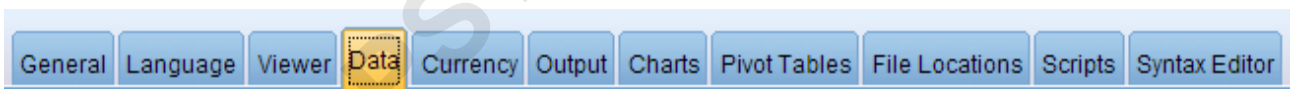
To change the century range for two-digit years, follow these steps:

## Using the Dialog Windows

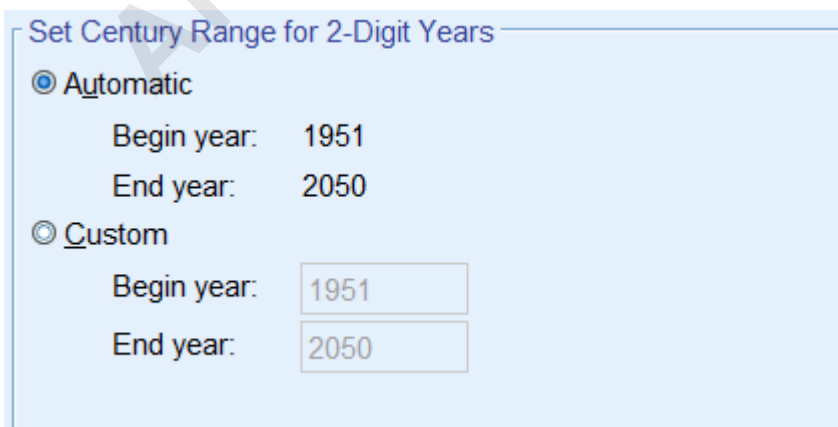
Click **Edit > Options**.



The Options window will appear. Click the **Data** tab at the top.



On the right-hand side you will see the **Set Century Range for 2-Digit Years** area.



By default, **Automatic** will be selected and two-digit years will be understood to fall in the range of the current year minus 69 to the current year plus 30. You can change the century range by clicking **Custom**, which will allow you to input a new beginning year (and the end year will be imputed for you). When you are finished, click **Apply**, and then click **OK**.

## Using Syntax

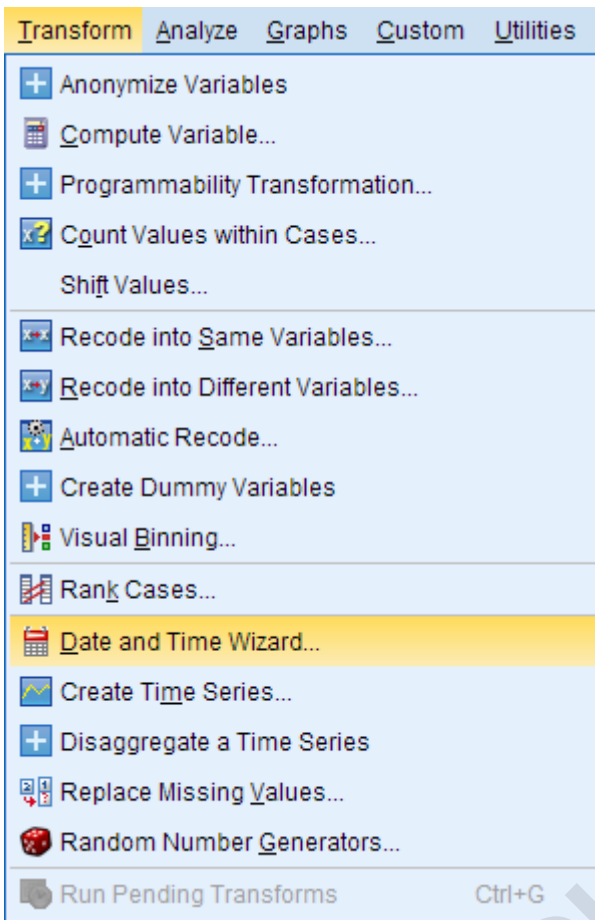
Alternatively, you can set the century range using the SET EPOCH command:

```
SET EPOCH=yyyy.
```

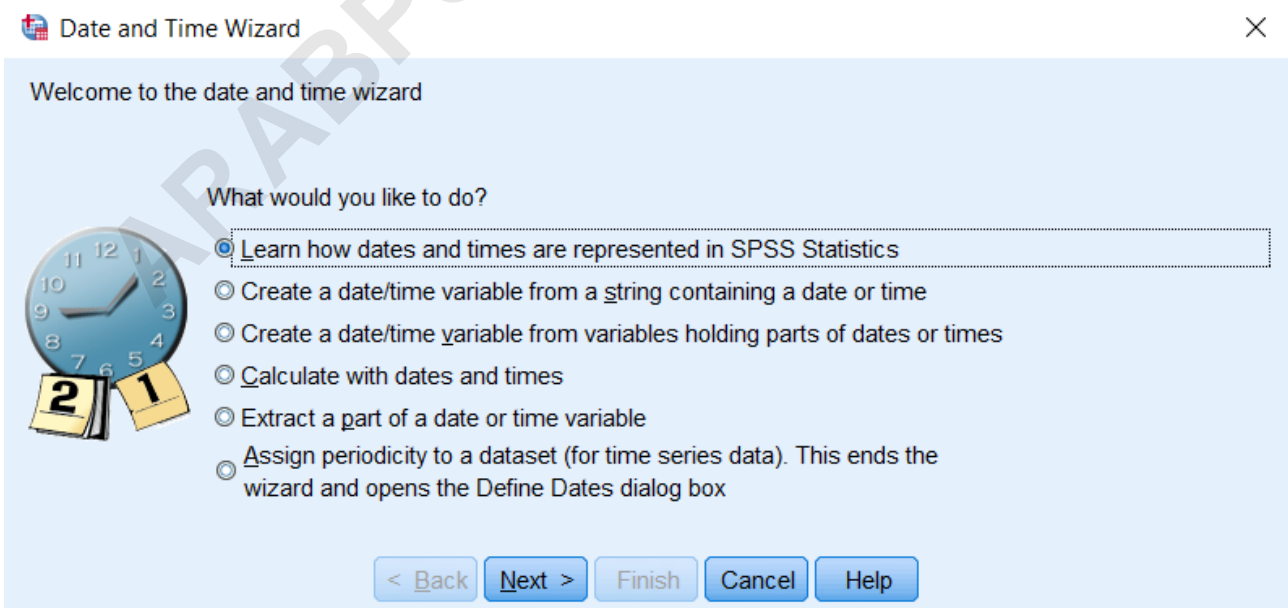
The yyyy to the right of the equals sign is the desired beginning year for the century range. For example, `SET EPOCH=1900` would set the century range to 1900 to 1999, while `SET EPOCH=1950` would set the century range to 1950 to 2049.

## Date and Time Wizard

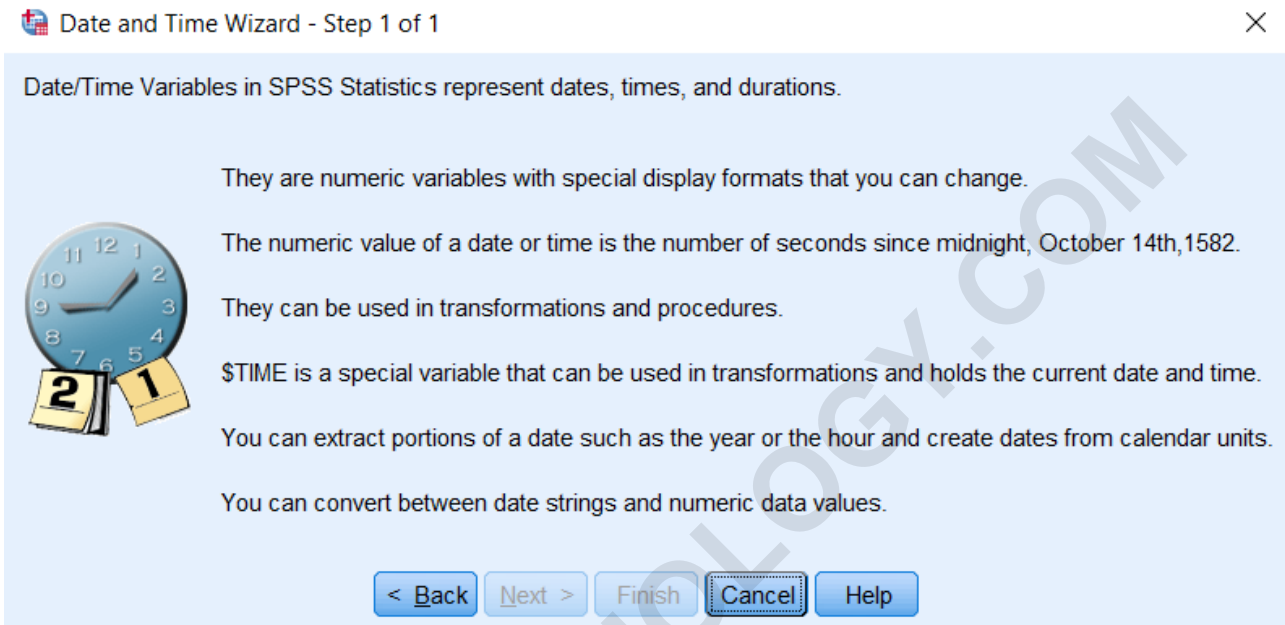
SPSS conveniently includes a Date and Time Wizard that can assist with transformations and calculations that involve date and time variables. To access the Date and Time Wizard, click **Transform > Date and Time Wizard**.



The Date and Time Wizard window will appear.



Although there are many options, it is useful to begin by first reading about how dates and times are represented in SPSS. We have selected this option (**Learn how dates and times are represented**) in the Date and Time Wizard window (depicted above). Now, click **Next**. You will see the following window.



When you are finished reading, click **Back** to return to the main Date and Time Wizard menu.

Note that the Date and Time Wizard can assist with many tasks related to dates and time, including:

Creating a date/time variable from a string containing a date or time  
 Creating a date/time variable from variables that contain parts of dates or times  
 Calculating with dates and time  
 Extracting parts of dates or time  
 Assigning periodicity to a dataset for time series data

We will not cover each of these options in this tutorial, but we will cover one of the most common uses for the Date and Time Wizard: calculations involving dates and times.

## Example: Converting a string variable to a date variable

### Problem Statement

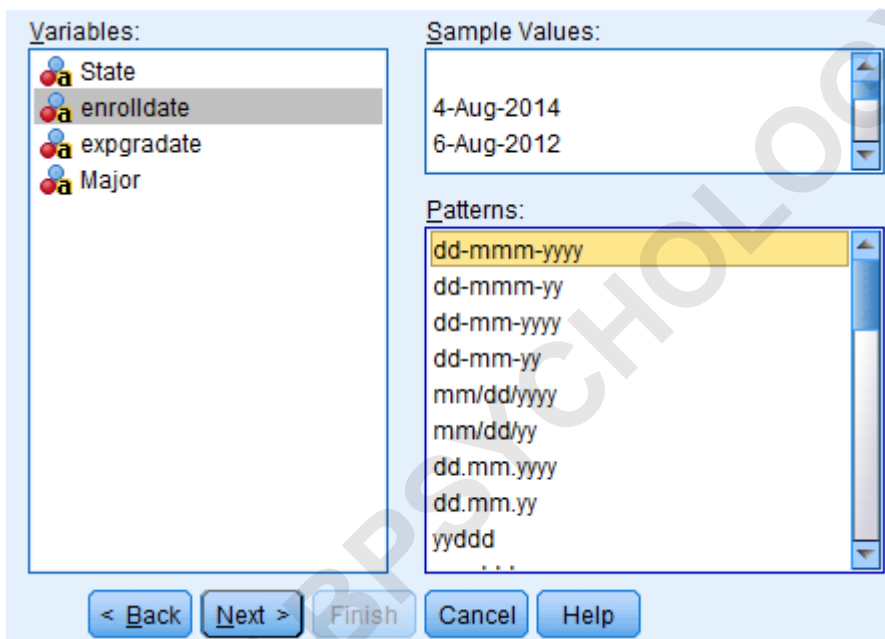
If you have datetime variables in a text or CSV file, SPSS will often read those variables in as string or character variables, instead of treating them as actual dates. In order to have those variables correctly recognized, you'll need to convert them from string to date.

In the sample dataset, the variable enrolldate (date of college enrollment) contains dates in the form *dd-mmm-yyyy*, but was read into the dataset as a string variable. Let's convert that variable from a string to a numeric date.

## Running the Procedure

### Using the Date & Time Wizard

Click **Transform > Date and Time Wizard**. Select **Create a date/time variable from a string containing a date or time**. Then click **Next**. In the Variables box, select variable enrolldate. This will show a preview of the values of the variable in the Sample Values box, so that you can select the correct pattern. In the Patterns box, click **dd-mmm-yyyy**. Then click **Next**.



In the Result Variable box, enter a name for the new date variable; let's call it `date_of_enrollment`. Optionally, you can add a variable label, and if desired, you are able to change the date format used for the output variable.

Complete or edit the specifications for new variable

Input Variable: enrolldate	Input Format: dd-mmm-yyyy
Result Variable: date_of_enrollment	Output Format: dd-mmm-yyyy dd-mmm-yy mm/dd/yyyy
Variable Label: Date of college enrollment	
Execution <input checked="" type="radio"/> Create the variable now <input type="radio"/> Paste the syntax into the syntax window	

< Back   Next >   Finish   Cancel   Help

Click **Finish**.

### Using Syntax

```
COMPUTE date_of_enrollment=number(enrolldate, DATE11).
VARIABLE LABELS date_of_enrollment 'Date of college enrollment'.
VARIABLE LEVEL date_of_enrollment (SCALE).
FORMATS date_of_enrollment (DATE11).
VARIABLE WIDTH date_of_enrollment(11).
EXECUTE.
```

*What's going on in this syntax?*

The first line (`COMPUTE`) actually computes the new date variable using the built-in function `number()`, which converts string variables to numeric variables. The argument `DATE11` tells SPSS that the content of the string variable is in `DATE11` format initially (`dd-mmm-yyyy`). The second line (`VARIABLE LABELS`) applies the variable label "Date of college enrollment" to the new variable. The third line (`VARIABLE LEVEL`) explicitly sets the measurement level of the new variable to `Scale`. The fourth line (`FORMATS`) applies a human-readable date format to the new variable. Here, we tell SPSS to continue using the `DATE11` (`dd-mmm-yyyy`) format for the new variable. The fifth line (`VARIABLE WIDTH`) tells SPSS how wide the column should be. This particular date format always has 11 characters, so the column is set to have width 11.

## Example: Computing Elapsed Time between Two Date-Time Variables

### Problem Statement

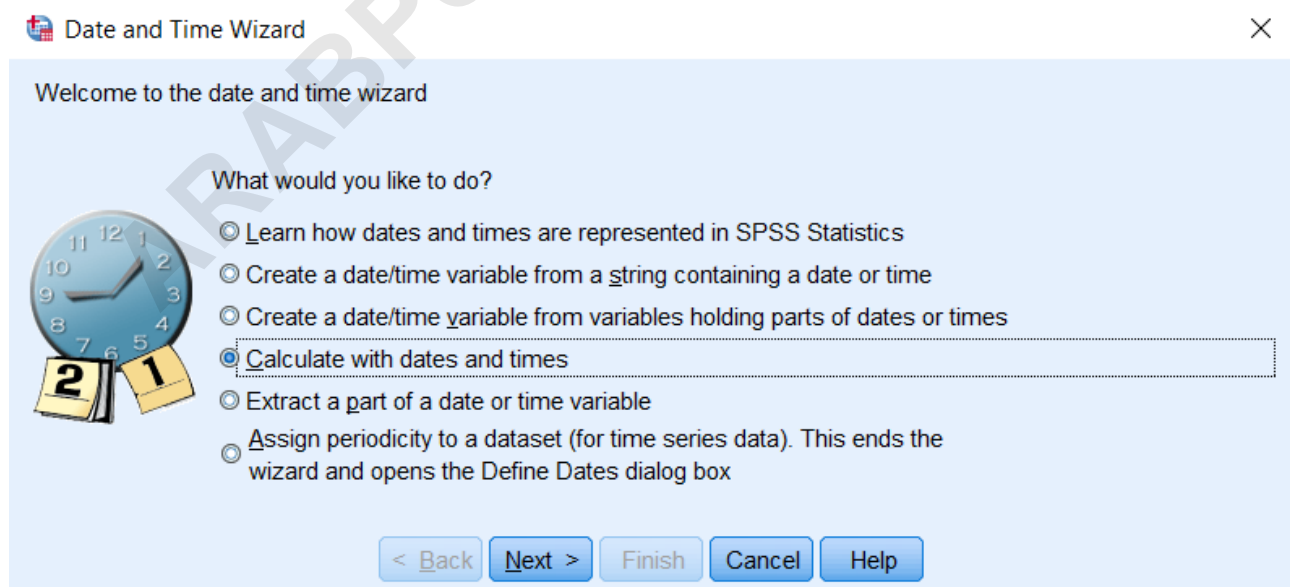
Sometimes you may need to calculate the length of time that has passed between two points in time. For example, you may wish to calculate the ages of people in your sample based on information you have about when they were born and what the current day/time/year is (or another date of your choosing). Any unit of time can be used. This means that you can calculate how many years, months, days, hours, minutes, or even seconds old each person is.

Before we can perform a calculation with dates and times, we first need to make sure that our dataset has at least two variables that represent time points. If you completed the above example, you will now have at least two date variables in the sample dataset: bday (the person's date of birth) and now date\_of\_enrollment (the date the person enrolled in college). We can compute the age that each person was when they enrolled in college using these two time points.

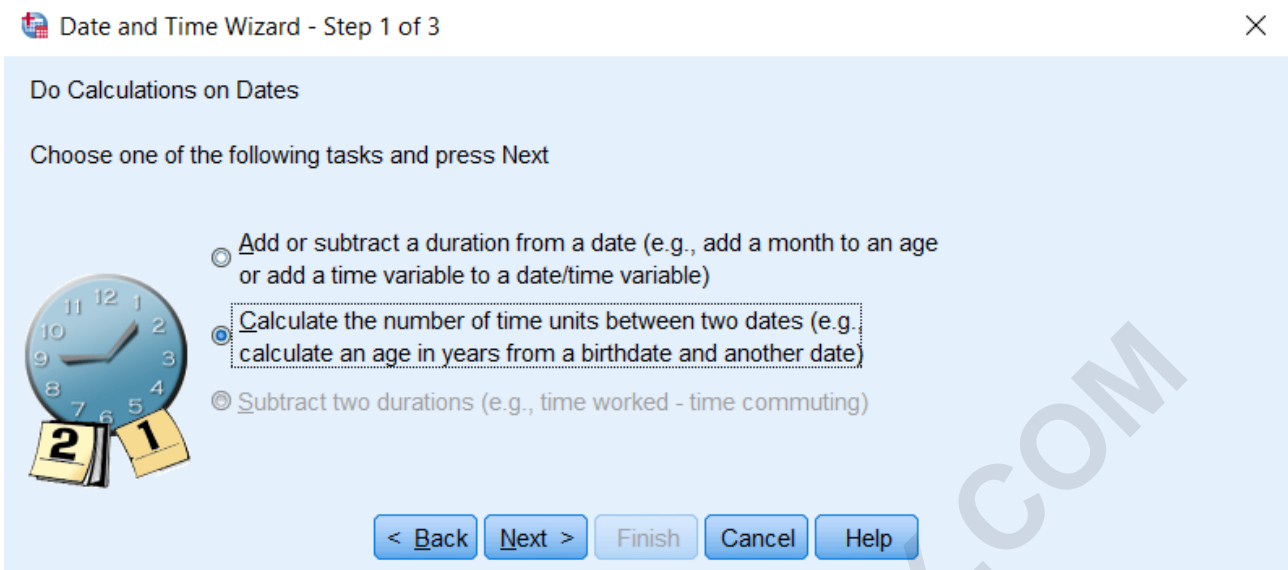
### Running the Procedure

#### Using the Date & Time Wizard

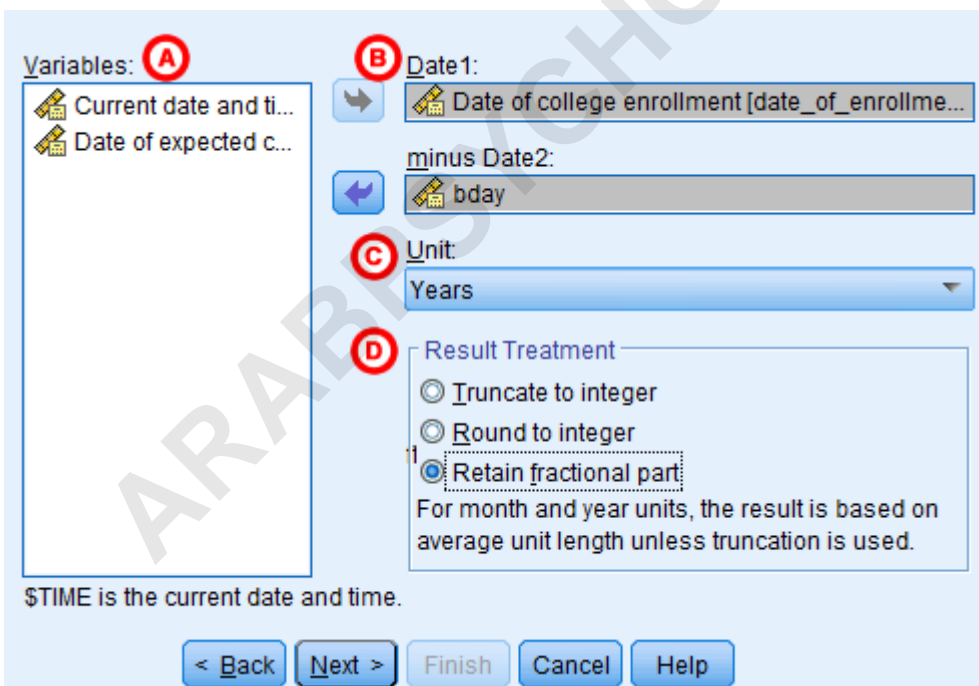
Click **Transform > Date and Time Wizard**. The Date and Time Wizard window will appear. Click **Calculate with dates and times** and then click **Next**.



Click **Calculate the number of time units between two dates**. Click **Next**.



We will now specify how the new variable should be computed from our existing date variables in this calculation: `date_of_enrollment` (the date at which a person enrolled in college) and `bday` (the date of birth).

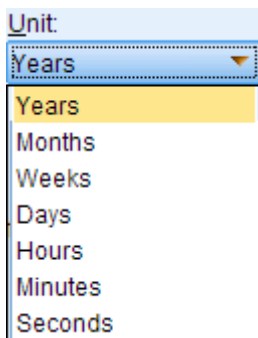


**A Variables:** Lists all of the available date and time variables in your dataset. It also includes a variable called "\$TIME" which represents the current date and time.

**B Date1:** The right half of the dialog box is where we will specify which variables to use, and how to

set up the calculation. In the **Date1** field, select the variable *date\_of\_enrollment* and in the **minus Date 2** field, select the variable *bday*. This specifies that SPSS should calculate *date\_of\_enrollment* minus *bday*, which will yield the number of years between when the person was born and when they enrolled in college (i.e., their age at college enrollment).

**CUnit:** The unit of time to use for the variable you are creating. You can choose among Years, Months, Weeks, Days, Hours, Minutes, and Seconds. In this example, select Years from the **Unit** list.



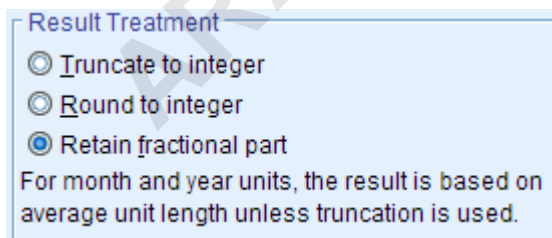
**DResult Treatment:** Specify how to treat the values of the variable that will be calculated. You can choose to *truncate to integer*, *round to integer*, or *retain fractional part*.

**Truncate to integer** means dropping the fractional part (e.g., 1.3 would become 1, and 1.6 would become 1).

**Round to integer** bumps the number to the nearest integer (e.g., 1.3 would round to 1, but 1.6 would round to 2).

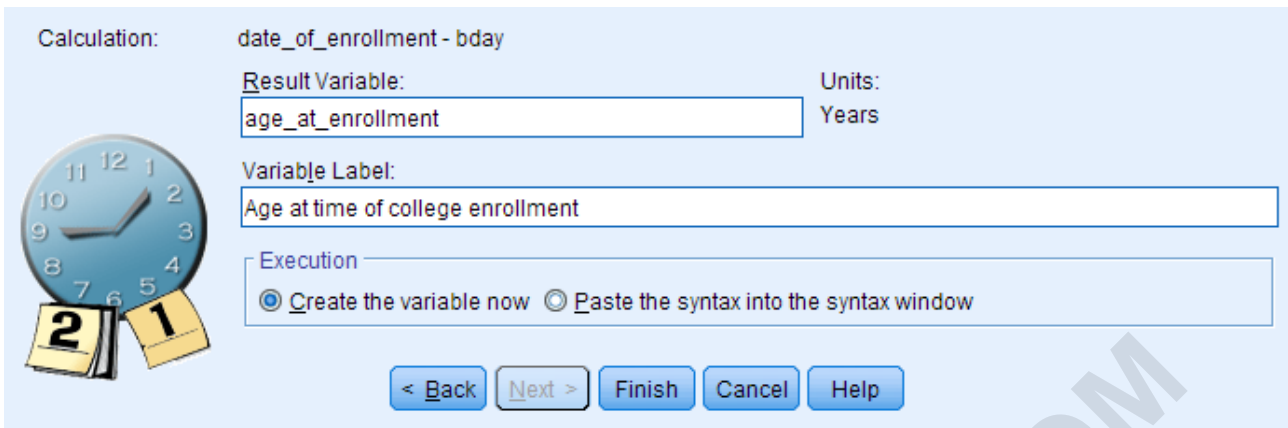
**Retain fractional part** means that the fraction will remain (e.g., 1.6 remains 1.6).

In this example, fractions will be retained so that the values for the new variable reflect the individual's exact age (in years) when they enrolled in college.



When you are finished setting up the calculation, click **Next**.

The final window asks you to name the variable you are creating in the **Result Variable** field and to provide a label in the **Variable Label** field. Here, the new variable is called *age\_at\_enrollment* and is labeled "Age at time of college enrollment".



Calculation: date\_of\_enrollment - bday

Result Variable: age\_at\_enrollment Units: Years

Variable Label: Age at time of college enrollment

Execution

Create the variable now  Paste the syntax into the syntax window

< Back Next > Finish Cancel Help

The **Execution** area allows you to choose how to create the new variable. You can have SPSS **Create the variable now**, which will immediately create the new variable in your dataset. Alternatively, select **Paste the syntax into the syntax window**, which will have SPSS write the syntax (command language) that will create the variable whenever you choose to run the syntax command in the future. This latter option will not create the new variable until you run the syntax.

When you are finished, click **Finish**.

Once your new variable has been created, it is always a good idea to check that the calculation was accurate. You can do this by spot-checking some of the rows in your data. You can manually calculate the time between *date\_of\_enrollment* and *bday* for some of the cases in the data and then compare the manual calculation to the value SPSS created in the new variable *age\_at\_enrollment*.

## Using Syntax

```
COMPUTE age_at_enrollment=(date_of_enrollment - bday) / (365.25 * time.days(1)).
VARIABLE LABELS age_at_enrollment "Age at time of enrollment (years)".
VARIABLE LEVEL age_at_enrollment (SCALE).
FORMATS age_at_enrollment (F8.2).
VARIABLE WIDTH age_at_enrollment(8).
EXECUTE.
```

*What's going on in this syntax?*

The first line (`COMPUTE`) performs the calculation of the elapsed times. Notice that the calculation isn't simply the difference of the two date variables: in the denominator, the term `(365.25*time.days(1))` corrects for different year lengths due to leap years. The second line (`VARIABLE LABELS`) applies the variable label "Age at time of enrollment (years)" to the new

variable. The third line (`VARIABLE LEVEL`) explicitly sets the measurement level of the new variable to Scale. The fourth line (`FORMATS`) tells SPSS that the computed variable is a numeric variable that has two decimal places and is at most 8 characters wide. The fifth line (`VARIABLE WIDTH`) sets the width of the variable to 8 characters. The last line (`EXECUTE`) tells SPSS to carry out the computation and add the new variable to the active dataset. (Without this line, SPSS will create the variable in the computer's memory but not actually add it to the dataset.)

## For More Information

[Datetime Variables Basics - SPSS-Tutorials.com](http://SPSS-Tutorials.com)

ARABPSYCHOLOGY.COM