

Keyboard Macros! An awesome tool you may have never heard of - once you do, you will never program the same again (It's that amazing!)

Steve Black, Precision for Medicine, Oncology and Rare Disease

ABSTRACT

Stop cutting and pasting old code into new programs, stop worrying about forgetting or misplacing really neat code, let SAS® remember it for you! This can be done quickly and easily through Keyboard Macros! The term Keyboard macro is a mild misnomer as they are not really macros or at least not in the way that most SAS programmers think of macros. They should really be called Keyboard Shortcuts or Keyboard Awesomeness. As these allow you to bring back all that awesome code you wrote. The title of the paper may be an exaggeration but I have found these to really speed up my programming. In this paper I will demonstrate how to create, use, and transfer keyboard macros. I will also discuss a few of the more in-depth aspects of Keyboard Macros with the intent that the reader will use and make them a part of their programming repertoire.

WHAT IS A KEYBOARD MACRO?

If you have ever searched through your programs looking for some useful code or find yourself copy/pasting large chunks of code from one program to another to save time but then end up having to tweak every little thing to get it to work. Keyboard macros are the solution you have been looking for. A keyboard macro, also known as a keyboard abbreviation, stores selected code in SAS and will place this saved code into your enhanced editor window when the proper abbreviation is typed in. A quick example is the program header required by various employers to state the program name, purpose, programmer, and change history. Once stored as a keyboard macro/abbreviation it's easy to place this code on the top of every program then fill in the needed items.

KEYBOARD SHORTCUTS

You may already be using these and not even know it. Common Keyboard shortcuts are Ctrl + C for copying text, and Ctrl + V for pasting copied text. Basically it is a saved process which is called by typing certain characters in the correct sequence. In the SAS windows environment, there are several SAS specific keyboard shortcuts that you may or may not be aware of, here is a list of ones that I have found super helpful.

Common Shortcuts:

- Find text Ctrl + F
- Find and replace text Ctrl + H
- Copy Selection Ctrl + C
- Paste Ctrl + V
- Save Ctrl + S

Not as Common:

- Comment the selected code (/) Ctrl + /
- Uncomment the selected code (/) Ctrl + Shift + /
- Convert selected text to upper case Ctrl + Shift + U
- Convert selected text to lower case Ctrl + Shift + L
- Stop Processing or Cancel Submitted Statement Ctrl + Break (near print screen)

Even less common but still very helpful:

- To move cursor to the matching DO/END statement Alt + [or Alt +]
- To move cursor to matching brace/parentheses Ctrl + [or Ctrl +] or Ctrl + (or Ctrl +)
- To close the active window CTRL+F4
- Collapse all folding blocks Alt + Ctrl + Number pad -
- Expand all folding blocks Alt + Ctrl + Number pad +
- Get Help for a SAS procedure Highlight the procedure name and press F1
- Clean up white space Ctrl + Shift + W
- Next Window Ctrl + Tab or Ctrl + F6
- Tile Windows Shift + F4
- Cascade Windows Shift + F5
- Selecting text vertically ALT + Left click (Highlighting) with the cursor

In SAS you can also create a few of your own simple Keyboard Shortcuts that maybe super helpful.

By pressing F9 you can see a list of a few of the system shortcuts sometimes referred to as DMKEYS that have been created. I created the clear command for F12 which clears whatever window I am in (log, results, output window) which is the same as Ctrl + E but faster. So instead of dragging the cursor up to the clear icon in SAS, I simply hit F12 and it's all cleared up.

<u>Key</u>	<u>Definition</u>
F1	help
F2	reshow
F3	end; /*gsubmit buffer=default*/
F4	recall
F5	wpgm
F6	log
F7	output
F8	zoom off;submit
F9	keys
F11	command focus
F12	clear
SHF F1	subtop

CREATING A KEYBOARD MACRO

Now that we see how SAS can save a process and make it easy to call that process, you can do the same with code that you have created. Simply type in the keyboard abbreviation or shortcut create and the code will appear in the SAS editor window. These can be really helpful once you build them into your programming routine.

So creating one is super simple.

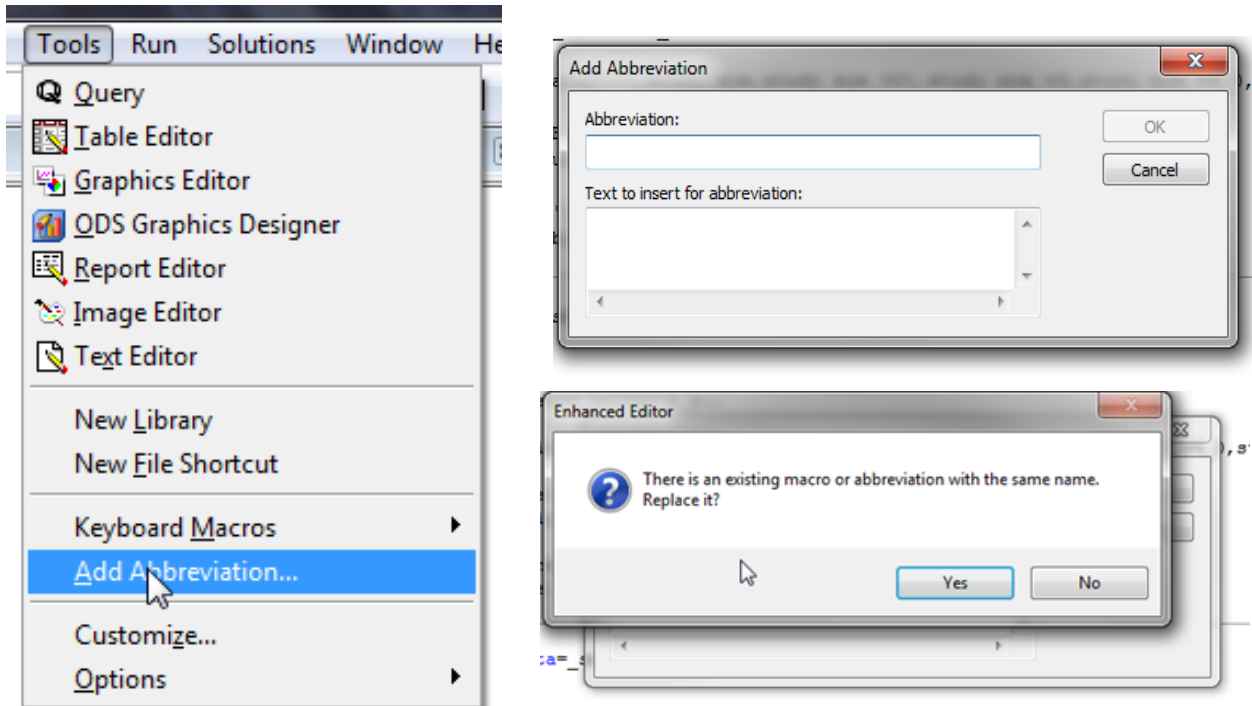
Step 1: Copy some code you think is worth saving.

Step 2: On the top bar go to **Tools** then go to **Add Abbreviation** or press **Ctrl + Shift + A**. This opens up a new window for you to place in your code.

Step 3: Paste in your code into the “Text to insert for abbreviation” box.

Step 4: Type in an abbreviation in the “Abbreviation” box to pull up this saved code. This can be most anything, I try to use an abbreviation that will not be commonly typed while programming such as “phead” for a header or “pmeans” for my template PROC MEANS code.

Step 5: click OK. If you have already created an abbreviation with that same abbreviation text, SAS will put up a warning, which is helpful, you can then overwrite the original abbreviation or modify your abbreviation text to something different.



CALLING A KEYBOARD MACRO

In the SAS editor Window simply type in the abbreviation created, once the last character is typed a small box appears with a small sample of the abbreviation code. This is helpful as you can recognize whether this is the abbreviation wanted or not.

```
pmeans|
proc means data=_pre_freq2 nop... (Abbrev)
```

If this is the abbreviation wanted, simply hit ENTER or hit the TAB key and the full code will be placed into your SAS Editor Window.

```
proc means data=_pre_freq2 noprint;
class trt01an;
var age;
output out=_stats n=n mean=mean std=std median=median min=min max=max;
run;
```

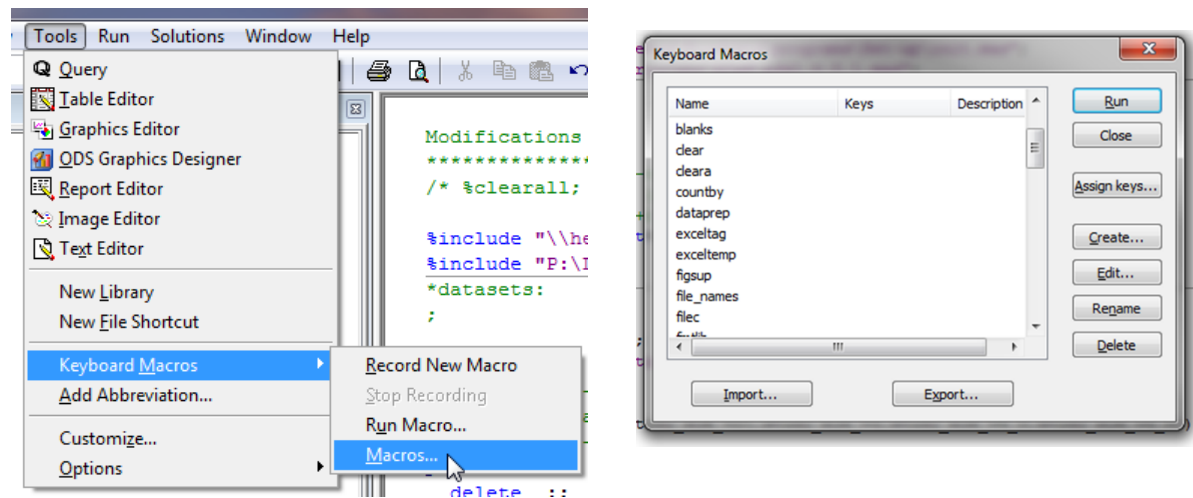
Amazing! I have not found a limit on the size of code that can be copied and pasted which is nice.

EFFECTIVE USES

This is where you can really get creative. I've found that setting these abbreviations up as template pieces to a table program has been amazingly helpful at creating table programs super-fast. I'll have one for my program header, then one for a generic 'bring in the analysis dataset', then one for the PROC FREQ or the PROC MEANS. Then I have one to add in my code to prep the data for a transpose - creating a variable that holds the text value of the statistic. Then one to transpose the frequency and mean data. I'll then have one set up to create a final dataset with any arrays needed. Then a separate one for the PROC REPORT. I also save any special code needed such as Unicode values for special characters, PROC SQL code, and other procedures such as COMPARE, IMPORT, and EXPORT). I also have example code to help me remember how things work such as retain statement or renaming variables when using a where statement. I commonly use some command x functions to bring in file names which I have saved as an abbreviation. So it's really anything you want to use again save it as an abbreviation.

MANAGING THE MACROS

Once these abbreviations have been created it's easy to forget abbreviations you used/assigned or what they do. You can easily see the entire list of abbreviations created by pressing **Ctrl + Shift + M**, or by selection **Tools>Keyboard Macros> Macros**.

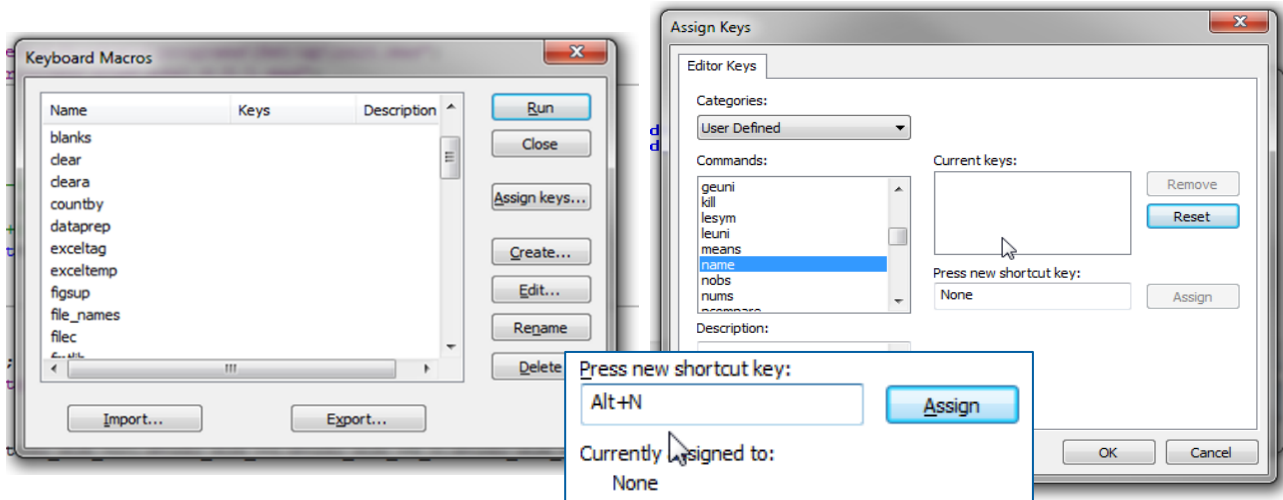


From this screen you can also edit, rename, and or delete previously created abbreviations. Sometimes I forget what a keyboard macro does so I can either select edit or rerun it to see the code, or better yet, create a list of the abbreviations with a brief description and save this list as a keyboard macro for easy recall. Switching between an office-based PC SAS to a server based VPN/ server based SAS has become more common and so it's very handy to be able to move these abbreviations between computers or share them with a co-worker.

To import or export the macros, simply click on the appropriate button in the keyboard macros option box which will open up a file directory box to either retrieve or create a keyboard macro file (.kmf extension). When exporting you also have the ability to name the export file, select which abbreviations you will include and save the file in the desired location. When importing only .kmf files will be allowed to be selected. If your system gets a little spotty it's handy to have your Keyboard Macros stored somewhere safe.

CREATING A SHORTCUT KEY

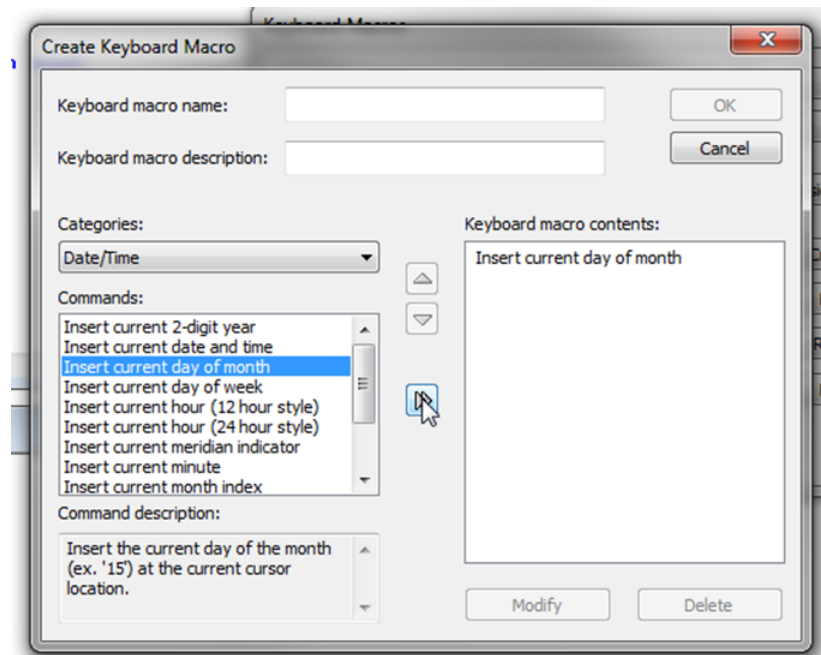
Getting even more fancy instead of typing in an abbreviation you can also create a Shortcut Key. This is done from the Keyboard Macros window where you would select an abbreviation and then select "Assign Keys".



I would assign the shortcut key by pressing my choice of shortcut keys such as Alt + N. By pressing the “Assign” button this would set-up the shortcut key. Now, whenever I want to type my name I can type ALT + N and it appears within the editor window

CREATING THE CURRENT DATE KEYBOARD MACRO

So now you know how to create, and manage keyboard shortcuts and abbreviations, so the last step is to create a Keyboard Macro. In the Keyboard Macros Window (Ctrl + Shift + M) click “Create”. This opens up a new window where you can give the new macro a name and description. Then using the options provided do some really neat things. I’ll illustrate how to create the current date in Date9 format – DDMMYY.



Steps to Insert into the Macro Contents Window:

- Insert current day of the month
- Insert current month name

- Move the cursor to previous word start
- Move cursor right
- Convert the selected text to uppercase
- Move cursor right
- Convert the selected text to uppercase
- Convert the selected text to uppercase
- Delete the next word start
- Insert current year

I assign the keys of ALT + D as a shortcut key and it's perfect. Now whenever I need to insert the current date in my headers or in my changes I log, I simply hit ALT+ D and it's done. If needed you can rearrange the order of the steps by using the up and down arrows in the create keyboard macro box.

CONCLUSION

In this paper, I have illustrated the amazing capability of keyboard macros and have provided a list of potential uses. I have also provided additional insight into the some of the more complicated steps that can be used to quickly insert the current date into their programs. I hope that this paper provides the framework for further use and development of your own keyboard macros.

REFERENCES

Carpenter, Arthur L. 2012 Doing More with the SAS® Display Manager: From Editor to ViewTable - Options and Tools You Should Know” Proceedings of the SAS Global 2012 Conference. Cary, NC: SAS Institute Inc. <http://support.sas.com/resources/papers/proceedings12/151-2012.pdf>

ACKNOWLEDGMENTS

Thanks goes to Albert Ortiz and Scott Davis who both build on ideas and make them better.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Steve Black
Precision for Medicine Inc.
steve.black@precisionformedicine.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.