



AGILITY
CLINICAL

Keyboard Macros - The most magical tool you may have never heard of - You will never program the same again (It's that amazing!)

Steve Black M.S.P.H

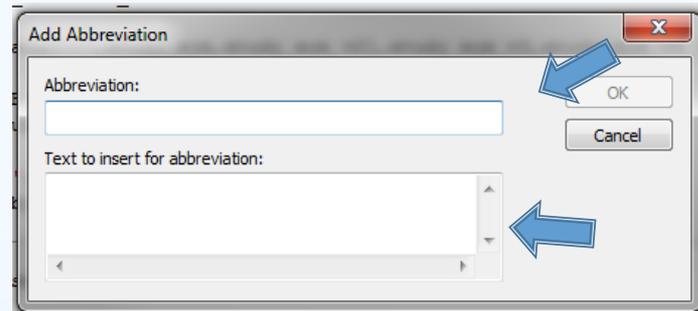
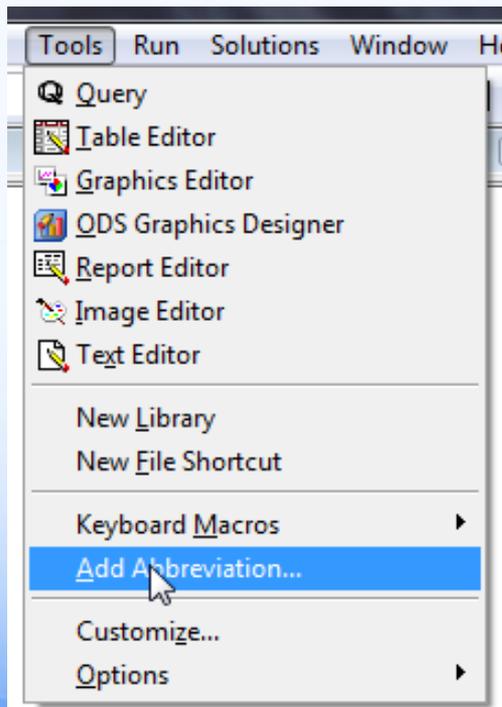
What are They?

- Simply put it is a way to save code and easy way you recall that saved code.
- Sometimes these keyboard macros are called keyboard abbreviations.
- They are REALLY helpful once you build them into your routine.
- They are magical, like a
- Rememberall!



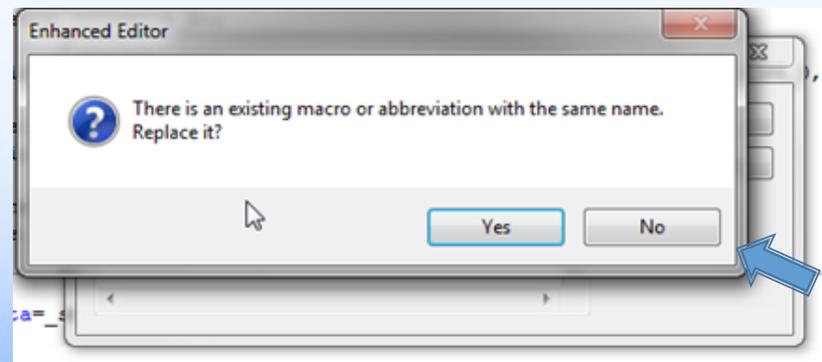
How to Create Them.

- Tools > Add or Shift + Ctrl + A
- Best if you have the code copied before starting to create the abbreviation.



Provide abbreviation here

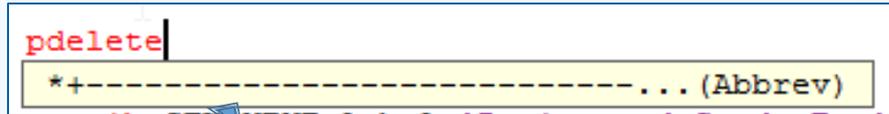
Paste/Type Code Here



Warning if it already exists

Calling a Keyboard Macro

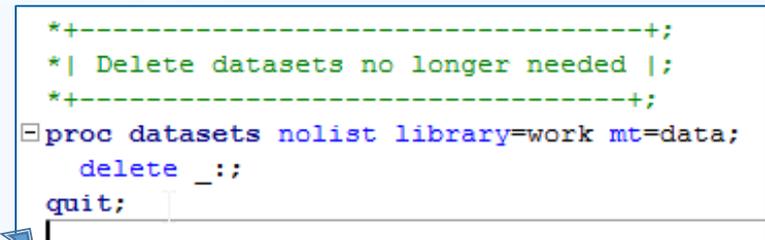
- Type in the pre-specified text/abbreviation into the enhanced editor window.



pdelete

*+-----... (Abbrev)

Abbreviation Description

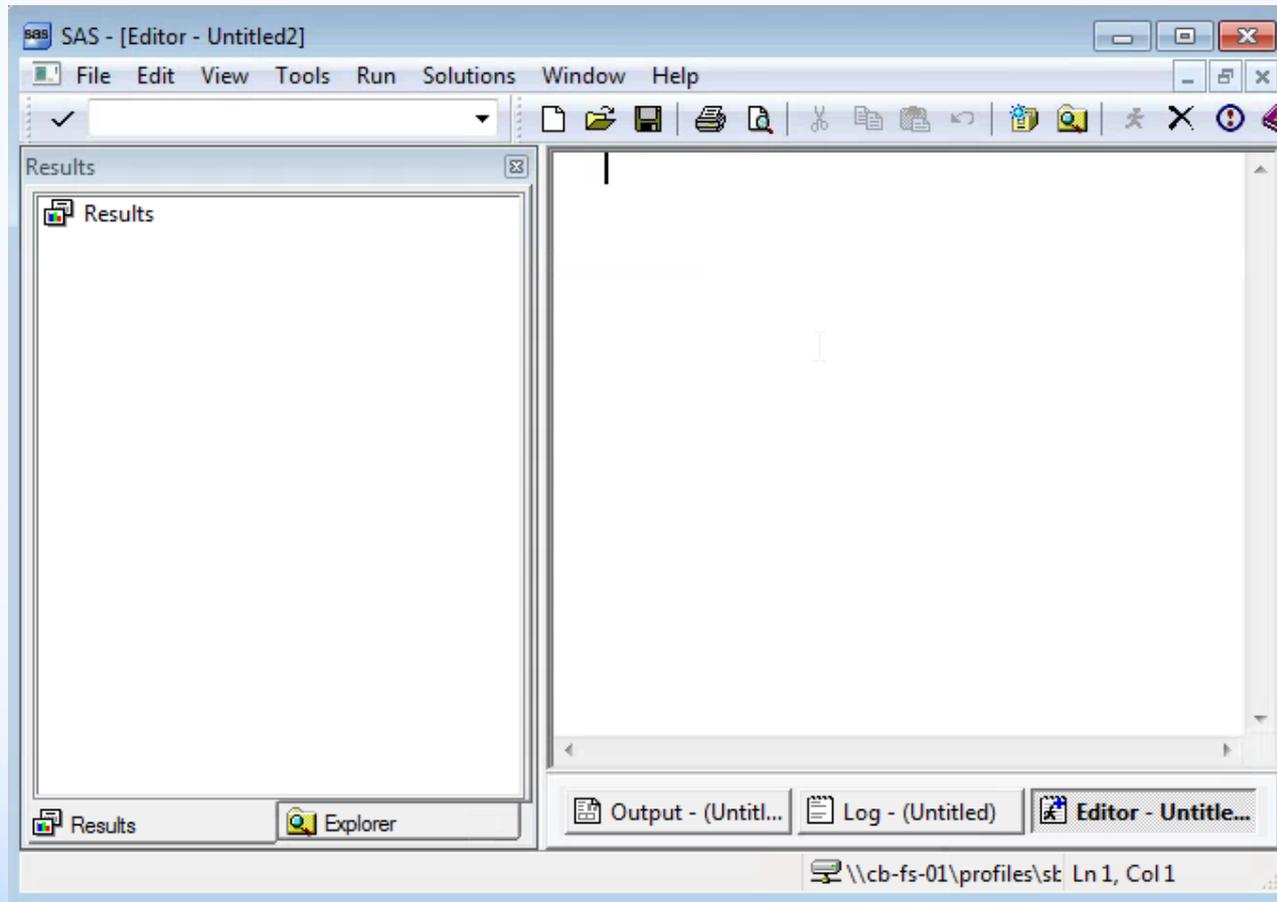


```
*+-----+;  
*| Delete datasets no longer needed |;  
*+-----+;  
proc datasets nolist library=work mt=data;  
  delete _;;  
quit;
```

Full Saved Code

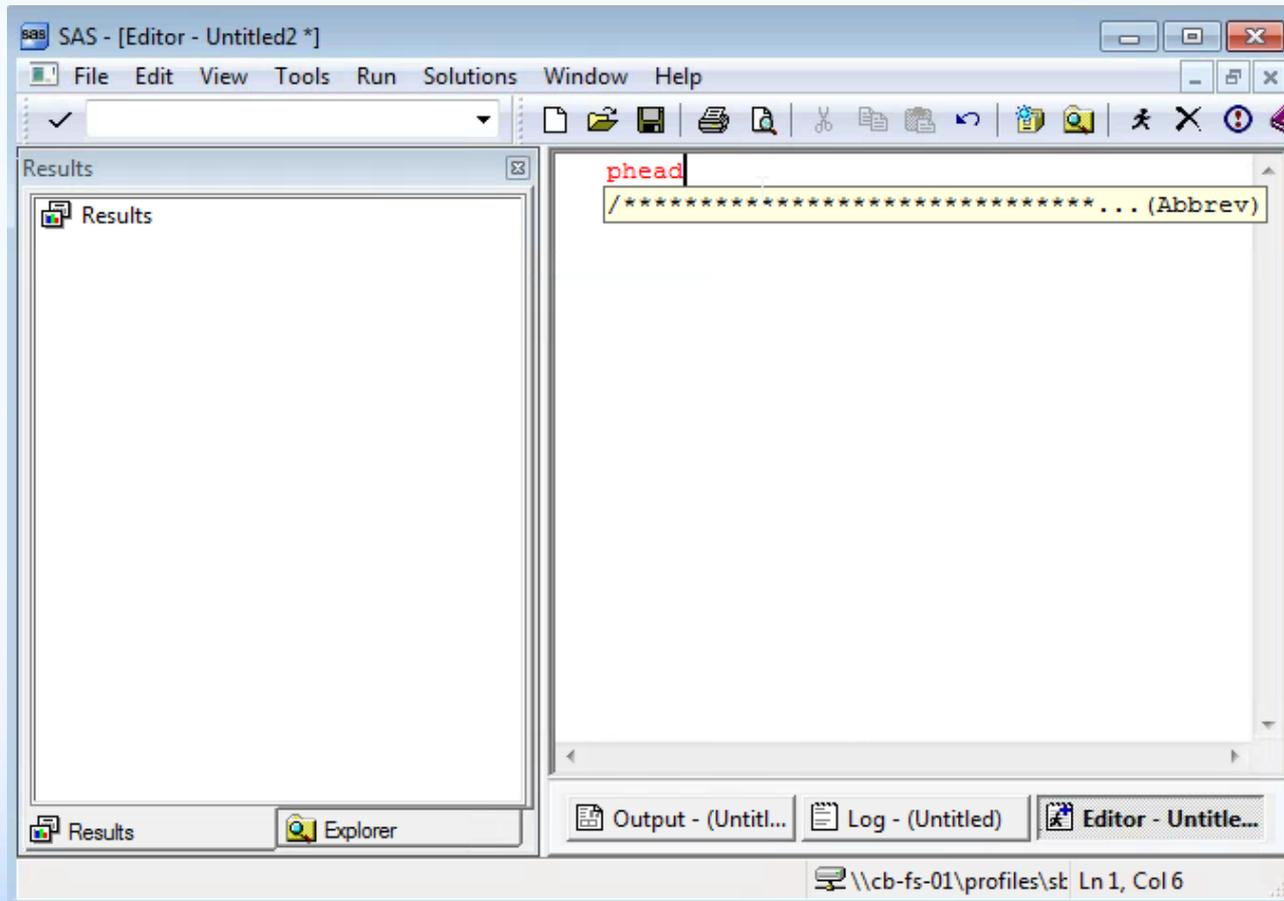
- Once text appears hit tab or return and the code gets placed in the window.

Let's See It Work

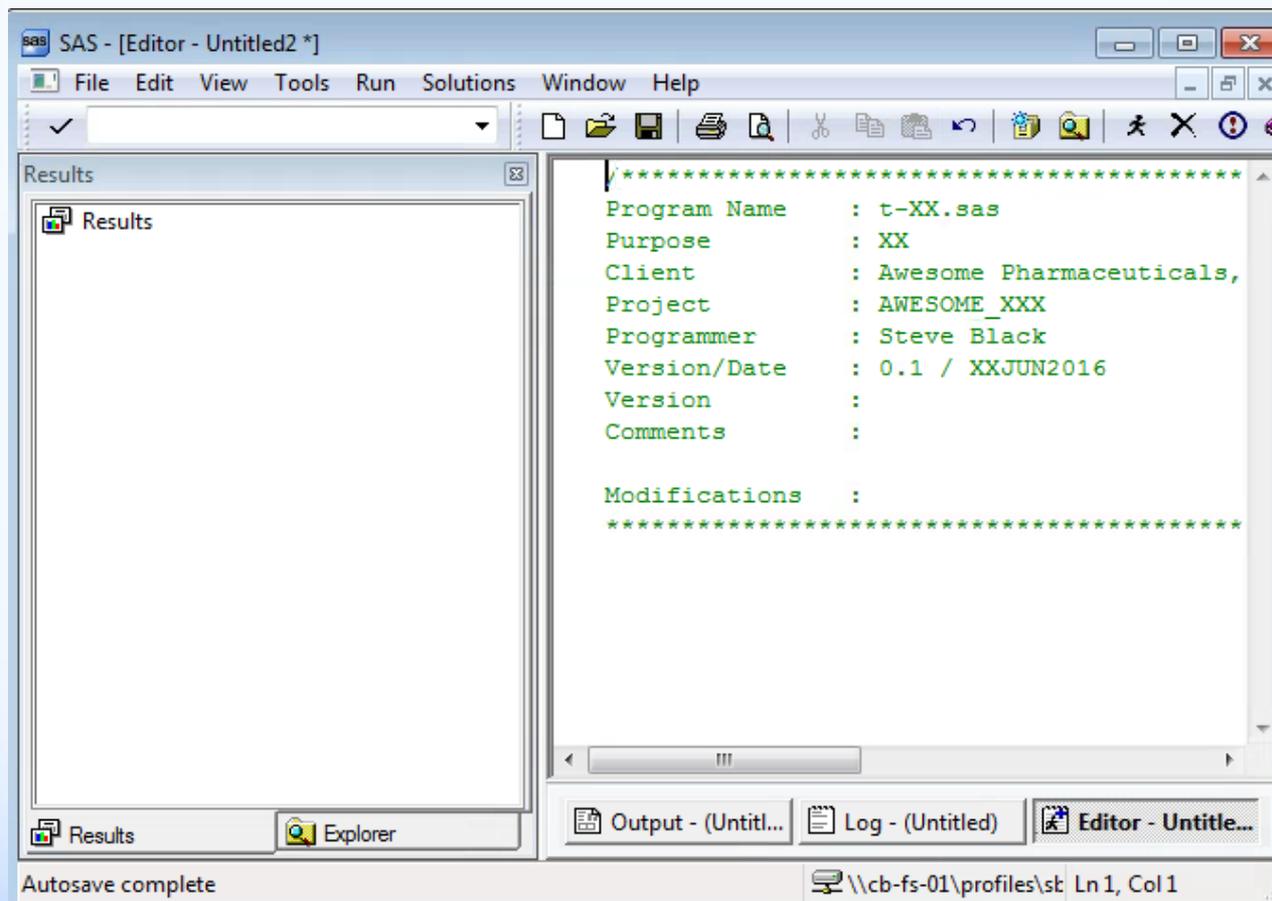


Blank SAS Editor Window

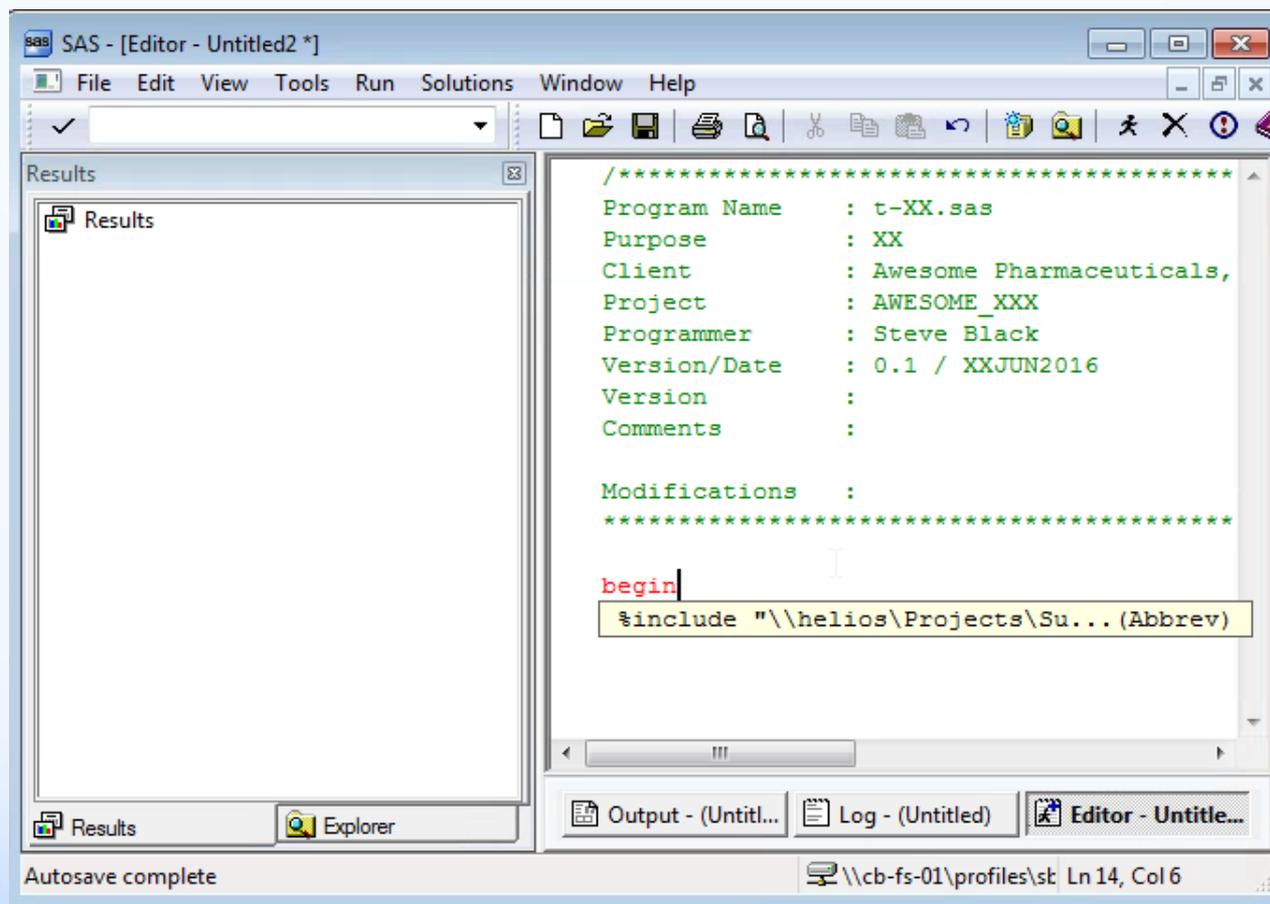
Add the Header



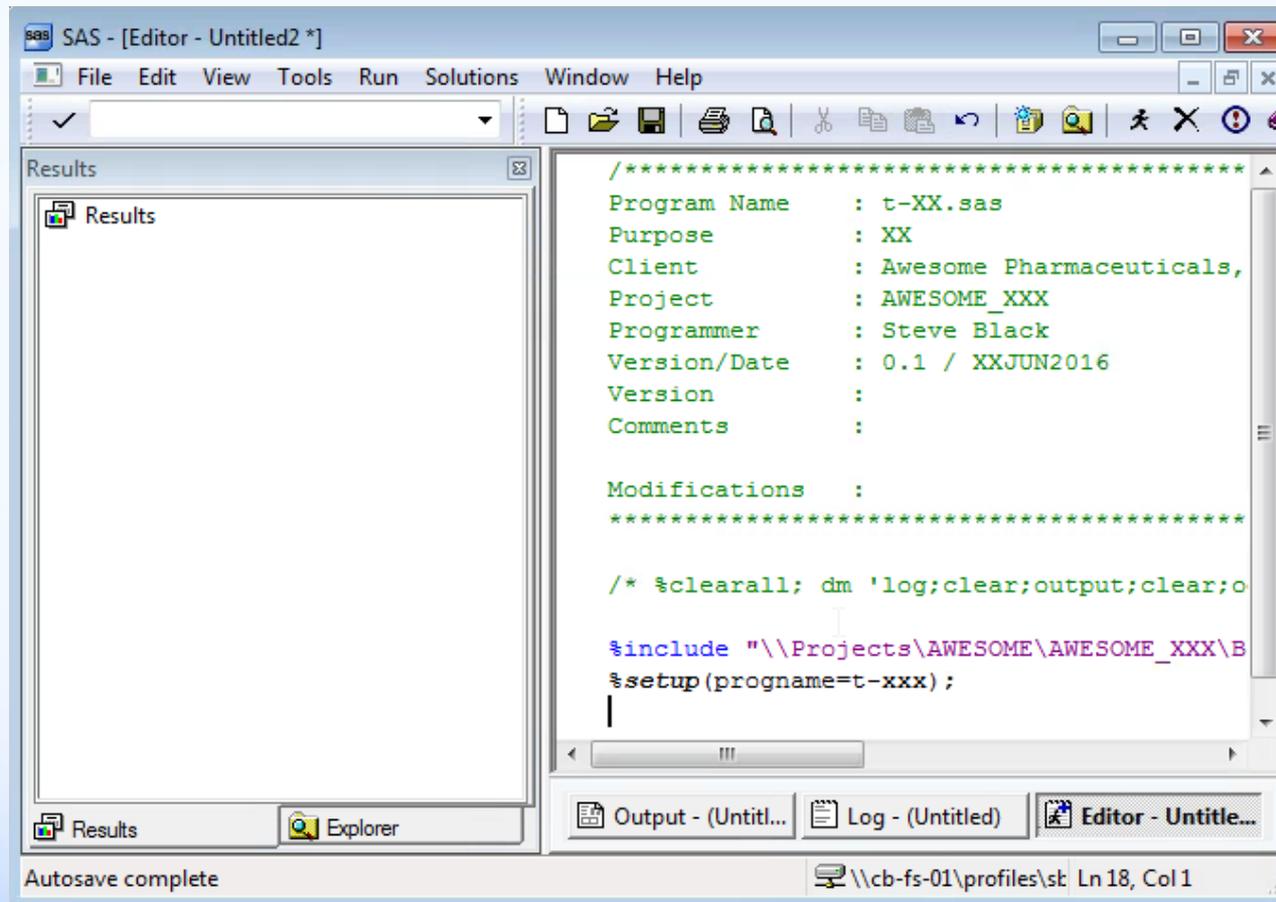
Done – Now add the details



Pull in the libraries and setup stuff



Done – now add the details



Create the frequency

```
/******  
Program Name      : t-XX.sas  
Purpose          : XX  
Client           : Awesome Pharmaceuticals, Inc.  
Project          : AWESOME_XXX  
Programmer       : Steve Black  
Version/Date     : 0.1 / XXJUN2016  
Version          :  
Comments         :  
  
Modifications    :  
*****  
  
/* %clearall; dm 'log;clear;output;clear;odsresults;clear'; */  
  
%include "\\Projects\AWESOME\AWESOME_XXX\Biostats\programs\Set-up\init.sas";  
%setup(progname=t-xxx);  
  
pfreq  
*+-----... (Abbrev)
```

Done – just tweak

```
Project       : AWESOME_XXX
Programmer    : Steve Black
Version/Date  : 0.1 / XXJUN2016
Version      :
Comments     :

Modifications :
*****

/* %clearall; dm 'log;clear;output;clear;odsresults;clear'; */

%include "\\Projects\AWESOME\AWESOME_XXX\Biostats\programs\Set-up\init.sas";
%setup(progname=t-xxx);

*+-----+;
*| Access necessary fields from SASDATA.A_### |;
*+-----+;

*** output frequency data ***;

ods output crosstabfreqs=_freq_out;

*** create frequency data ***;

proc freq data=sasdata.a_pre_proc;
  tables (study pre_ruth limb)*txgroup /norow ;
run;
|
```

Prepare the freq output

```
*** output frequency data ***;  
  
ods output crosstabfreqs=_freq_out;  
  
*** create frequency data ***;  
proc freq data=sasdata.a_pre_proc;  
  tables (study pre_ruth limb)*txgroup /norow ;  
run;  
  
dataprep  
  *** get data into table format...(Abbrev)
```

Done – just add in the guts

```
*** get data into table format ***;

data _data_prep;
set _freq_out;

*** create total number of subjects ***;

if index(table,'study') and _type_ in ('11','10') then y_axis=0;

*** create variable 1 ***;

else if index(table,'###') then do;
    if _type_ in ('01','00') then y_axis=10;
else y_axis=compress(1||###)*1;
end;

else if index(table,'###') then do;
    if _type_ in ('01','00') then y_axis=20;
else y_axis=compress(2||###)*1;
end;

*** create x_axis ***;
if txgroup=. then txgroup=3;
x_axis=txgroup;

*** create display ***;

if y_axis in (0,10,20) then display=compbl('\b N='||trim(left(frequency)));
else if _type_='11' then display=compbl(compress(frequency)||" ("||trim(left
else if _type_='10' then display=compbl(compress(frequency)||" ("||trim(left

*** delete all non-essential data ***;

if y_axis ne .;

run;
```

Transpose the data

```
if y_axis in (0,10,20) then display=compbl('\b N='||trim(left(fr
else if _type_='11' then display=compbl(compress(frequency)||" (
else if _type_='10' then display=compbl(compress(frequency)||" (
*** delete all non-essential data ***;

if y_axis ne .;

run;

ptransp|
*** sort by y_axis x_axis ***;... (Abbrev)
```

Done

```
*** sort by y_axis x_axis ***;  
|  
proc sort data=_data_prep;  
by y_axis x_axis;  
run;  
|  
*** transpose data ***;  
|  
proc transpose data=_data_prep out=_trans_data;  
by y_axis;  
id x_axis;  
var display;  
run;  
|
```

Add in continuous stats

```
*** transpose data ***;  
  
proc transpose data=_data_prep out=_trans_data;  
by y_axis;  
id x_axis;  
var display;  
run;  
  
pmeans|  
proc means data=_pre_freq2 nop... (Abbrev)
```

Done – just modify per data

```
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;

*** create axis and display variables ***;

data _means (keep=y_axis display x_axis); length display $15;
  set
  _stats (in=a where=( _type_=1));

  *** create x_axis ***;
  x_axis=trt01an;

  *** create y_axis for var1 ***;

  if a then do;
    y_axis=10;
    display=strip(put(n,best.));
    output;
    y_axis=11;
    display=strip(put(mean,9.1)||" ("||strip(put(std,9.2))||"");
    output;
    y_axis=12;
    display=compress(put(median,9.0));
    output;
    y_axis=13;
    display= strip(put(min,9.0))||", "||strip(put(max,9.0));
    output;
  end;

run;
```

Transpose mean data

```
display=strip(put(mean,9.1))||" ("||strip(put(std,9.2))||")";  
output;  
y_axis=12;  
display=compress(put(median,9.0));  
output;  
y_axis=13;  
display= strip(put(min,9.0))||", "||strip(put(max,9.0));  
output;  
end;  
  
run;  
  
ptranspm  
*** sort by y_axis x_axis ***;... (Abbrev)
```

Done

```
run;
```

```
*** sort by y_axis x_axis ***;
```

```
proc sort data=_means;  
by y_axis x_axis;  
run;
```

```
*** transpose data ***;
```

```
proc transpose data=_means out=_trans_means;  
by y_axis;  
id x_axis;  
var display;  
run;
```

Create a blank template for missing data

```
proc transpose data=_means out=_trans_means;  
  by y_axis;  
  id x_axis;  
  var display;  
run;
```

blanks|

```
*** create blanks for report a... (Abbrev)
```

Done – just modify per data

```
*** create blanks for report and labels ***;

data _blanks;
  do y_axis=9 to 19 by 10;
    output; end;
  do y_axis=10 to 13;
    output; end;
  do y_axis=20 to 22;
    output; end;
run;

proc sort data=_blanks;
  by y_axis;
run;
```

Finalize program code

```
do y_axis=9 to 19 by 10,  
output; end;  
do y_axis=10 to 13;  
output; end;  
do y_axis=20 to 22;  
output; end;  
run;
```

```
proc sort data=_blanks;  
by y_axis;  
run;
```

```
programc  
*** create linedesc and set in...(Abbrev)
```

Done – just put in the details

```
*** create linedesc and set in blanks ***;

data _program;
length linedesc $200 _1 _2 _3 $17;
merge _trans_data (in=a) _blanks (in=b) _trans_means;
by y_axis;

*** create linedesc ***;

if y_axis=0 then linedesc='\b Number of Subjects Enrolled';

else if y_axis=80 then linedesc='\b Study Limb';
else if y_axis=81 then linedesc=' Right';
else if y_axis=82 then linedesc=' Left';

else if y_axis=10 then linedesc='\b Creatinine (mg/dL)';
else if y_axis=20 then linedesc='\b WBC (103 cells/ $\mu$ L)';
else if y_axis=30 then linedesc='\b RBC (Million cells/ $\mu$ L)';
else if y_axis=40 then linedesc='\b Platelet Count (103/ $\mu$ L)';
else if y_axis=50 then linedesc='\b Resting Ankle Brachial Index (ABI)';
else if y_axis=60 then linedesc='\b Toe Brachial Index (TBI)';

else if y_axis in (11,21,31,41,51,61) then linedesc=' Mean (Std Dev)';
else if y_axis in (12,22,32,42,52,62) then linedesc=' Median';
else if y_axis in (13,23,33,43,53,63) then linedesc=' Min - Max';

else if y_axis=70 then linedesc='\b Baseline Rutherford Category';
else if y_axis=71 then linedesc=' Category 2 - Moderate Claudication';
else if y_axis=72 then linedesc=' Category 3 - Severe Claudication';
else if y_axis=73 then linedesc=' Category 4 - Ischemic Rest Pain';

*** fill in for missing data ***;

if y_axis in (71,72,73,81,82) and _1='' then _1='0 (0.0%)';
if y_axis in (71,72,73,81,82) and _2='' then _2='0 (0.0%)';
if y_axis in (71,72,73,81,82) and _3='' then _3='0 (0.0%)';
```

Create the report

```
if y_axis in (71,72,73,81,82) and _1='' then _1='0 (0.0%)';  
if y_axis in (71,72,73,81,82) and _2='' then _2='0 (0.0%)';  
if y_axis in (71,72,73,81,82) and _3='' then _3='0 (0.0%)';  
  
*** rename _1 _2 _3 ***;  
  
rename _1=ocol1 _2=ocol2 _3=ocol3;  
  
drop _name_;  
run;  
  
preport  
*+-----... (Abbrev)
```

Done – just change what's needed.

```
*+-----+;  
*| Generate Final Table in RTF format |;  
*+-----+;  
  
ods escapechar='~';  
ods listing close;  
ods tagsets.rtf file = "&outpath" &repopt;  
  
title3 "&ttlnum";  
title4 "&ttldesc";  
title5 "&ttlpop";  
  
footnote1 "Abbreviations: MedDRA, Medical Dictionary for Regulatory Activities";  
footnote3 "Note: Group 1, Drug A 4.0 mg / Sham; Group 2, Drug B 0.5 mg / Lixit 0.5 mg or Laxit 1.25  
footnote5 "~{super [1]} Medical history diagnoses are coded to system organ classes using MedDRA ver:  
footnote7 &fnref;  
footnote8 &fnpath;  
  
proc report data=_program headline headskip nowindows missing split = "@"  
style(header column) = [protectspecialchars = off] style(report) = [bordertopwidth = 1.5];  
columns (extra ord MHSOC _1 _2 _3 _4 _5);  
define extra / order noprint;  
define ord / order descending noprint;  
define MHSOC / "System Organ Class~{super [1]}" flow style={just=left cellwidth=4.0in};  
define _1 / "Group 1~n(N=&tx1)" style={just=center cellwidth=0.8in};  
define _2 / "Group 2~n(N=&tx2)" style={just=center cellwidth=0.8in};  
define _3 / "Group 3~n(N=&tx3)" style={just=center cellwidth=0.8in};  
define _4 / "Group 4~n(N=&tx4)" style={just=center cellwidth=0.8in};  
define _5 / "Total~n(N=%trim(&tx5))" style={just=center cellwidth=0.8in};  
compute before extra;  
line put ' ';  
endcomp;  
ods tagsets.rtf close;  
ods listing;
```

We just created a table in like 1 minute!!

- Shhhhh... don't let them know.
- Think of the possibilities!
- Instead of complicated templates just do it piece by piece.
- What else can we do?



A few more, because I like them.

```
1
❏ %macro autofmt;
proc sql noprint;
select unique paramn, param into: num1 - : num99, : text1 - : text99
from adam.adlb where parcat1n=4 and anl01fl='Y';
%let numparm=&sqllobs;
quit;

proc format;
value paramns
%do x = 1 %to &numparm;
  &&num&x = "&&text&x";
%end;
;
run;

%mend;
%autofmt;

*** run primary program ***;
%let batchpath=P:\&SPONSOR_PATH\&PROJECT_PATH\Biostats\programs\TLFs;
❏ proc sql noprint;select program into: program from psd;quit;
%batch(&batchpath,inc="&program");

*** QC proc compare ***;
❏ data _null_;length npath $200;
npath="&qcpath";
call symput('new_qcpath',tranwrd(npath,scan(npath,-1,'\'),'QC_documentation\'||scan(npath,-1,'\')));
run;

title3 "&TTLQC";
footnote "&QCFNPATH";
ods listing close;
ods rtf file = "&new_qcpath" style=rtf_tnr_9pt;

*** compare against primary data ***;
❏ proc compare base=final compare=_program;
var text2 _1 _2 _3 _4 _5;
run;
ods rtf close;
ods listing;

%checkprog;

❏ PROC IMPORT OUT= WORK._C
DATAFILE= &d
DBMS=EXCEL R
RANGE="crfs$a3:e60"
GETNAMES=YES;
MIXED=NO;
SCANTEXT=YES;
USEDATE=YES;
SCANTIME=YES;

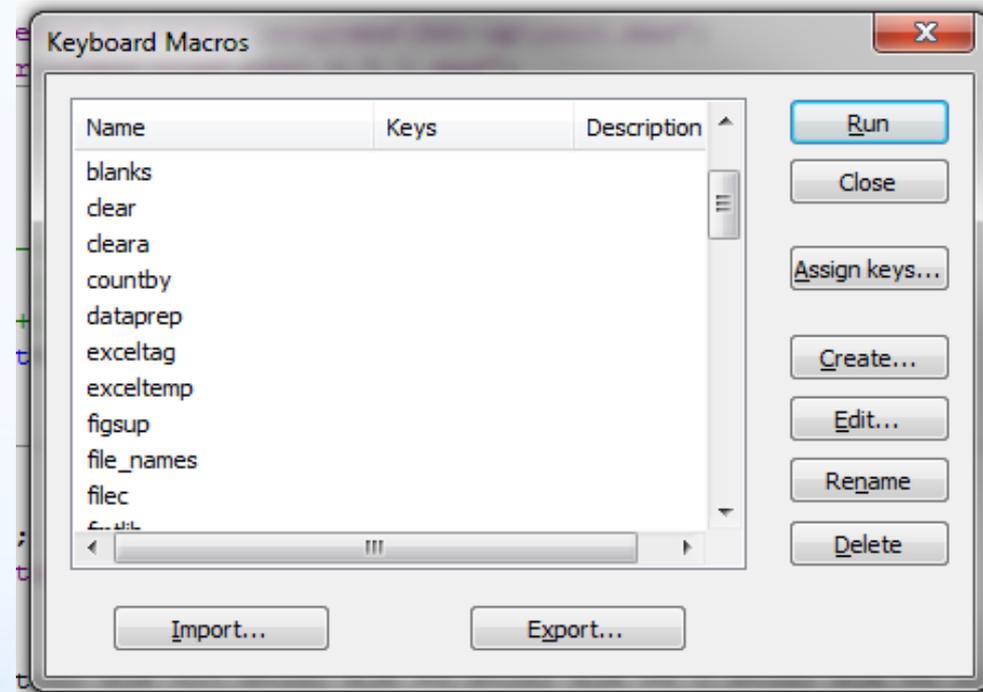
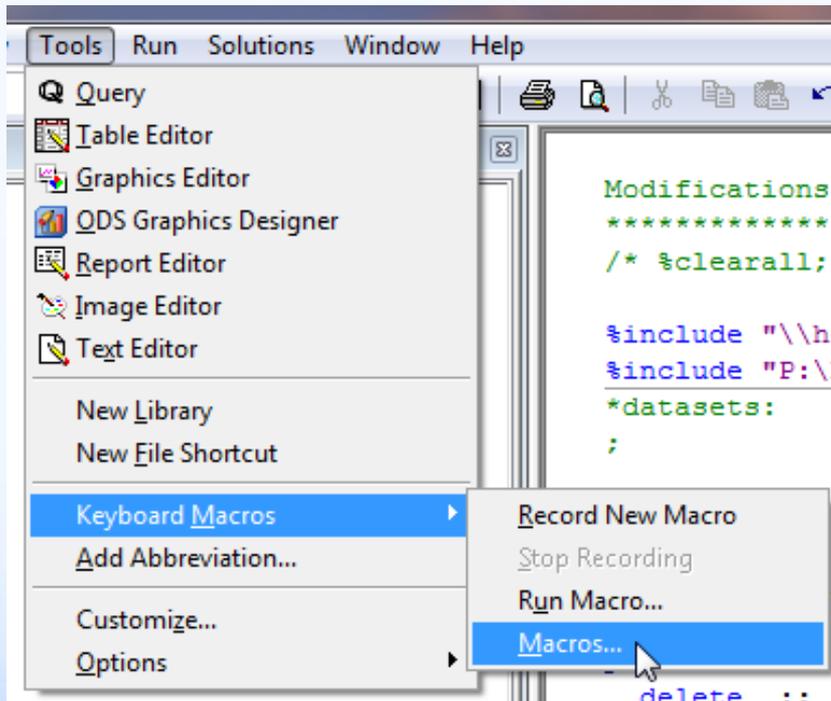
RUN;|
```

Effective Uses

- Not comprehensive in any way, but to provide a number of uses of the keyboard macros that I've found effective.
- Creating array templates, code for preparing data prior to a PROC FREQ, common code for PROC TEMPLATE, EXCELXP Tagsets, X commands, FMTLIB code in PROC FORMAT, PROC IMPORT, PROC FREQ, PROC MEANS, PROC UNIVARIATE, Capturing P-values, PROC SQL, PROC TRANPOSE, dataset templates, PROC REPORT, PROC COMPARE, date/time conversions, company log warnings, programmer name.

Managing the Keyboard Macros

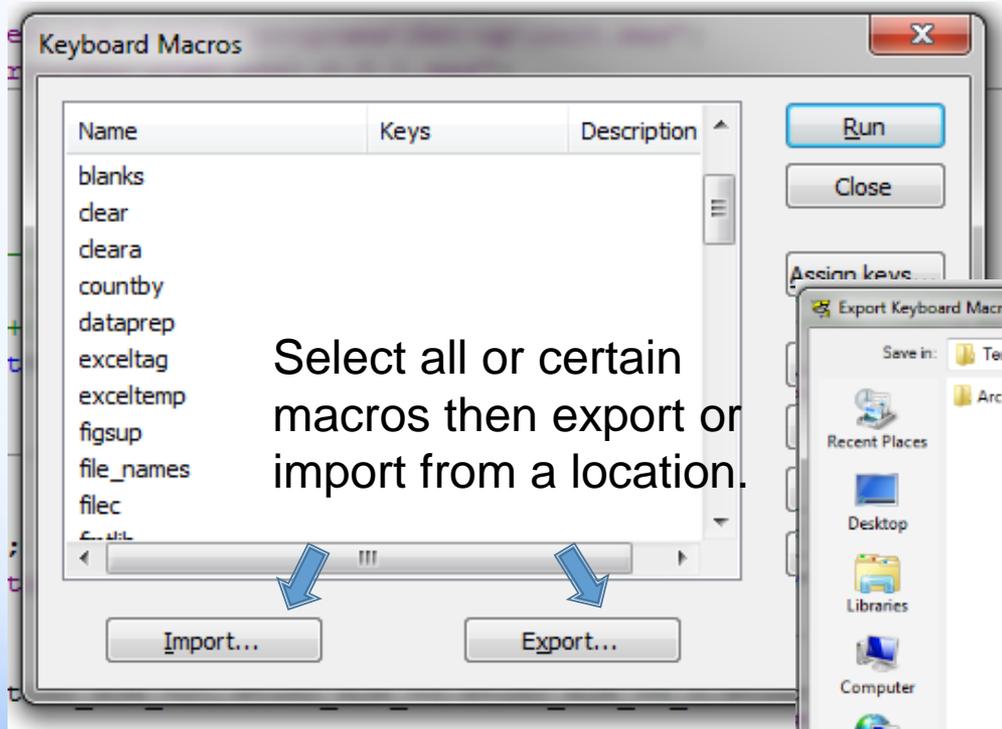
- Tools > Keyboard Macros > Macros or Ctrl + Shift + M.



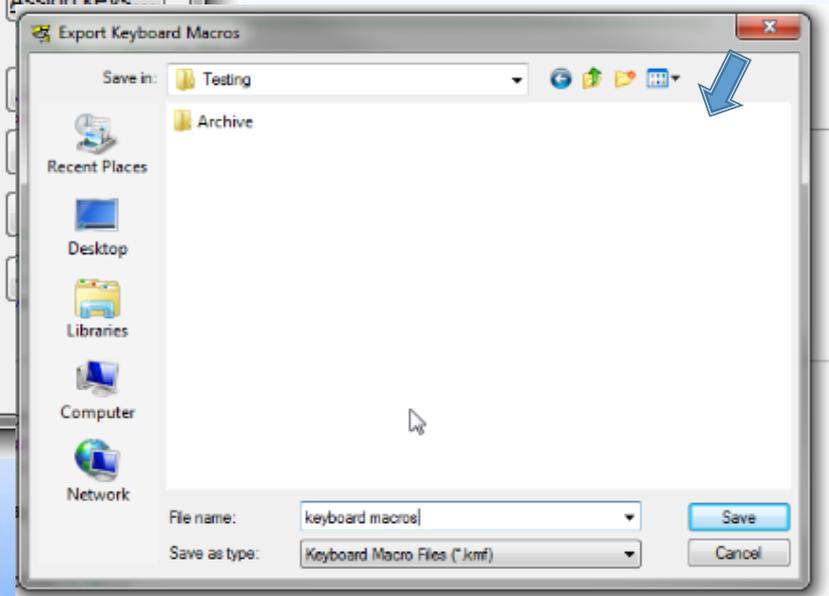
Main Keyboard
Macro Window

Import/Export them

- In case you switch computers, move to a server based system or you decide to work for us.

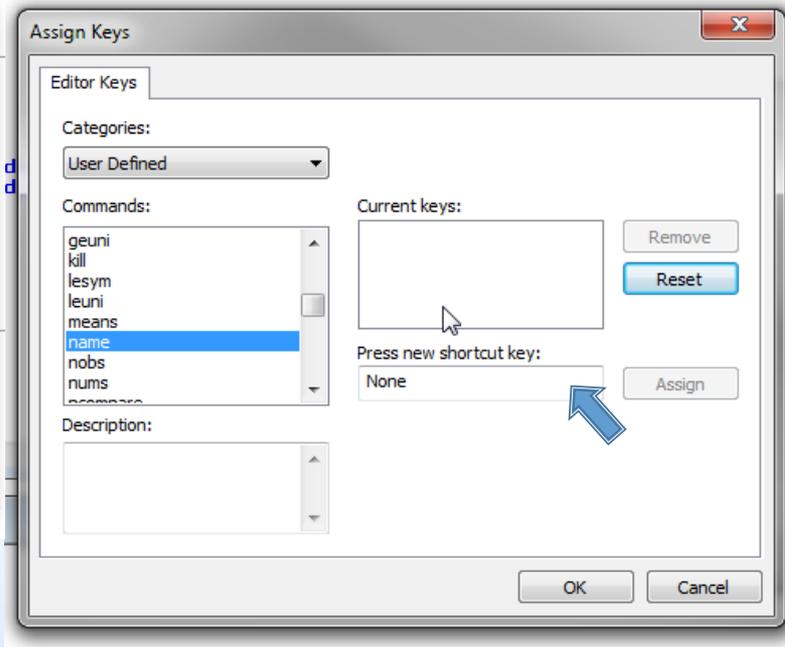
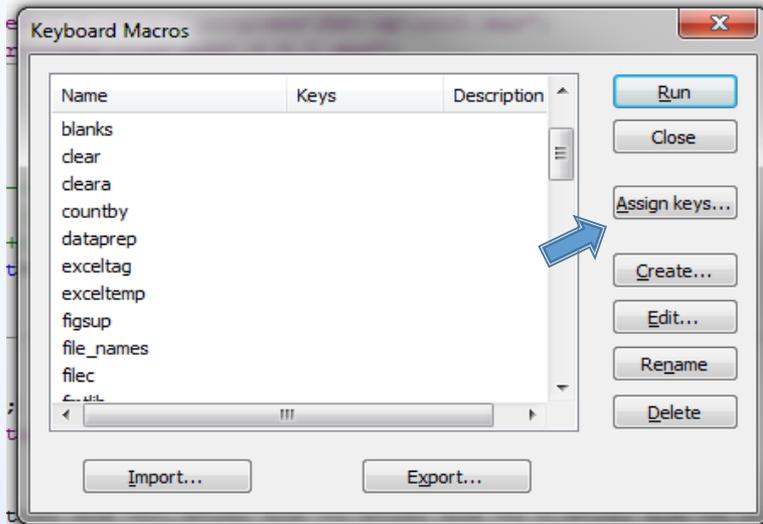


Browse Window for Importing/Exporting Keyboard Macros



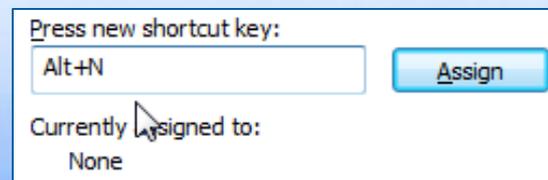
Creating a Shortcut Key

- Instead of typing in an abbreviation an even simpler way is to create a shortcut key.

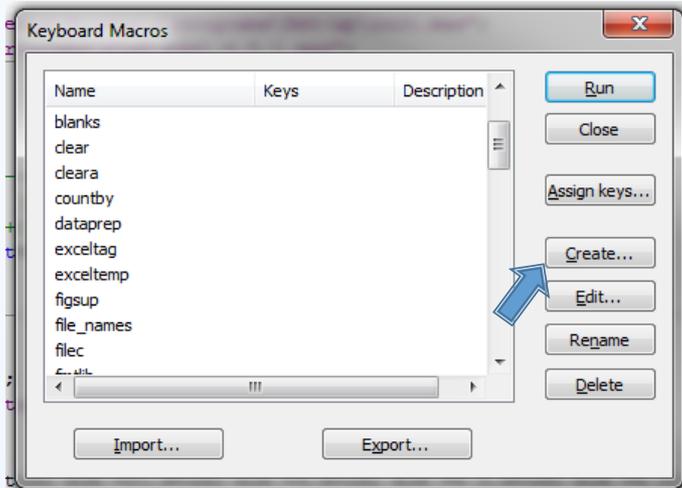


Type in
shortcut key
wanted

Main Keyboard
Macro Window
Select Assign Keys



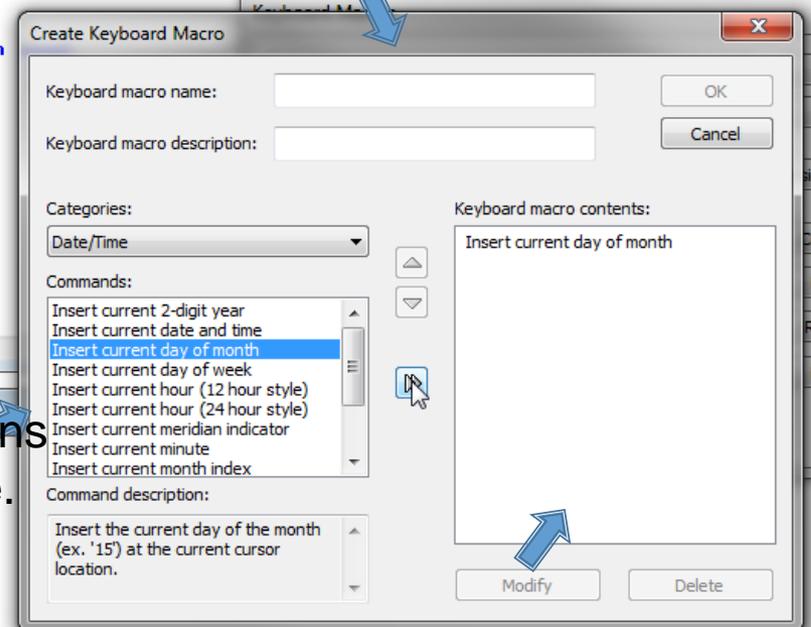
Using System Commands



Simply select
Create..

Lots of options
for date/time.

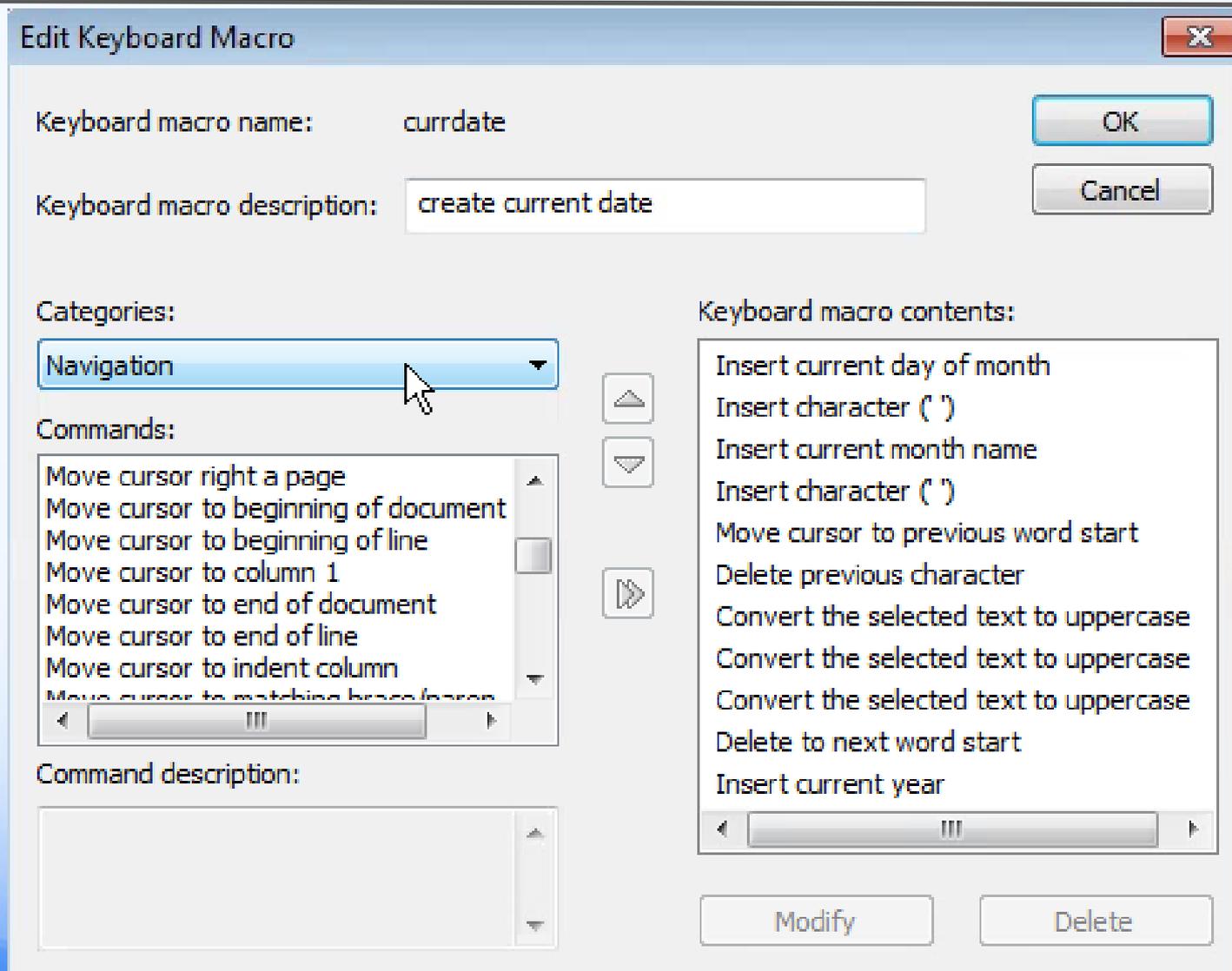
Add macro name



There are many command
options when the Category
'All' is selected. Have fun
Exploring!

Once selected you
can re-organize if
needed

Create Current Date in DDMMYYYY format



Summary

- How to create an abbreviation
- Ideas for abbreviations
- How to manage abbreviations
- Creating a shortcut key
- Using system commands
- Take the initiative create one soon and begin programming even better.

Questions?



Contact Me!

Name: Steve Black

Organization: Agility Clinical Inc.

Address: 6005 Hidden Valley Rd.

City, State ZIP: Carlsbad, CA 92011

Work Phone: 760-658-5919

E-mail: sblack@agility-clinical.com