

COMMAND Your Session:

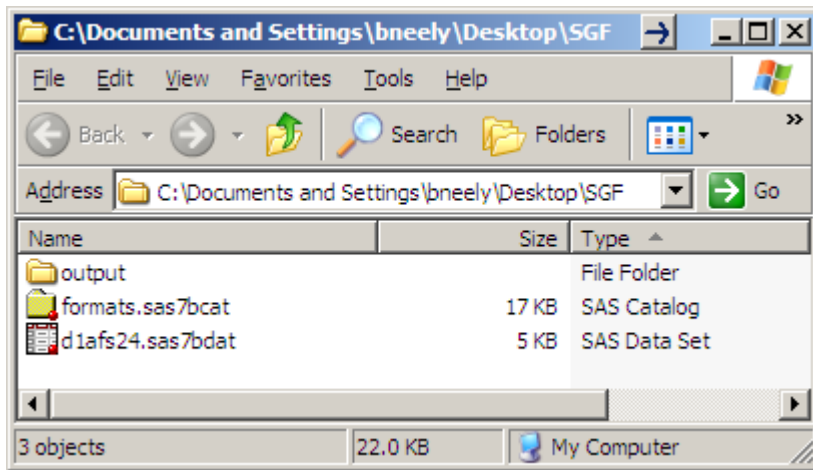
SAS® Commands and Command-Style Macros

Ben Neely

RTI International

Motivation

Provided Information



First Look at Data

Alphabetic List of Variables and Attributes

#	Variable	Type	Len	Format	Label
1	UserDefinedDichotomous	Num	8	Y1N0F.	Dichotomous Variable

Motivation

Data with Y1NOF Format

SAS - [FSVIEW: WORK.D1AFS24 (B)]

Command ===>

Obs	UserDef inedD ichotomous
1	YES
2	YES
3	NO
4	NO
5	NO
6	NO
7	NO
8	YES
9	YES

Attempt Data Subset

SAS - [FSVIEW: WORK.D1AFS24 (B)]

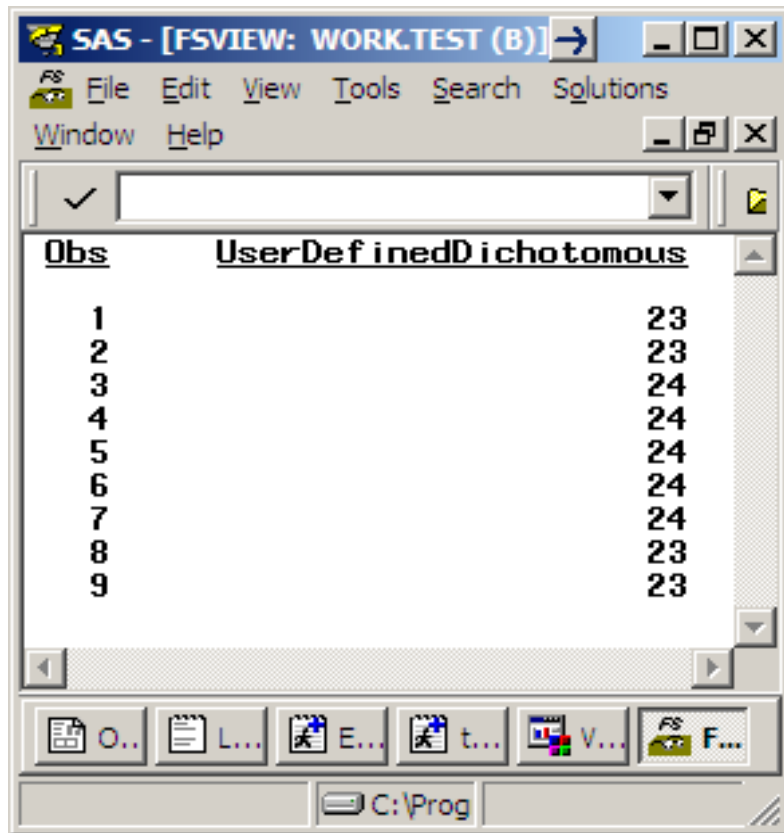
Command ===>

Obs	UserDef inedD ichotomous
1	YES
2	YES
3	NO
4	NO
5	NO
6	NO
7	NO
8	YES
9	YES

ERROR: WHERE clause operator requires compatible variables.

Motivation

Data without Y1NOF Format



The screenshot shows the SAS interface with a window titled 'SAS - [FSVIEW: WORK.TEST (B)]'. The menu bar includes File, Edit, View, Tools, Search, Solutions, Window, and Help. The data table below has the following content:

Obs	UserDefinedDichotomous
1	23
2	23
3	24
4	24
5	24
6	24
7	24
8	23
9	23

Command Wish List

- Crosswalk from values to format labels
- Allow for partial format name input
- Provide the number of formats that match partial input
- Delivery Method – Output Window

Outline



- Terminology
- Command Submission Locations (CSL)
- Find the Right Command
- Create Highly Customized Commands using Command-Style Macros

Terminology | SAS Command vs. SAS Syntax

- Often SAS command and SAS Syntax are used interchangeably, but...
- SAS Command
 - ⊗ Ex| Pressing the Print Icon (Submits the command DMPRINT).
- SAS Syntax
 - ⊗ Ex| Data Step / Procedure

Terminology|

SAS Command Line vs.
Operating System (OS) Command Line

SAS Command Line

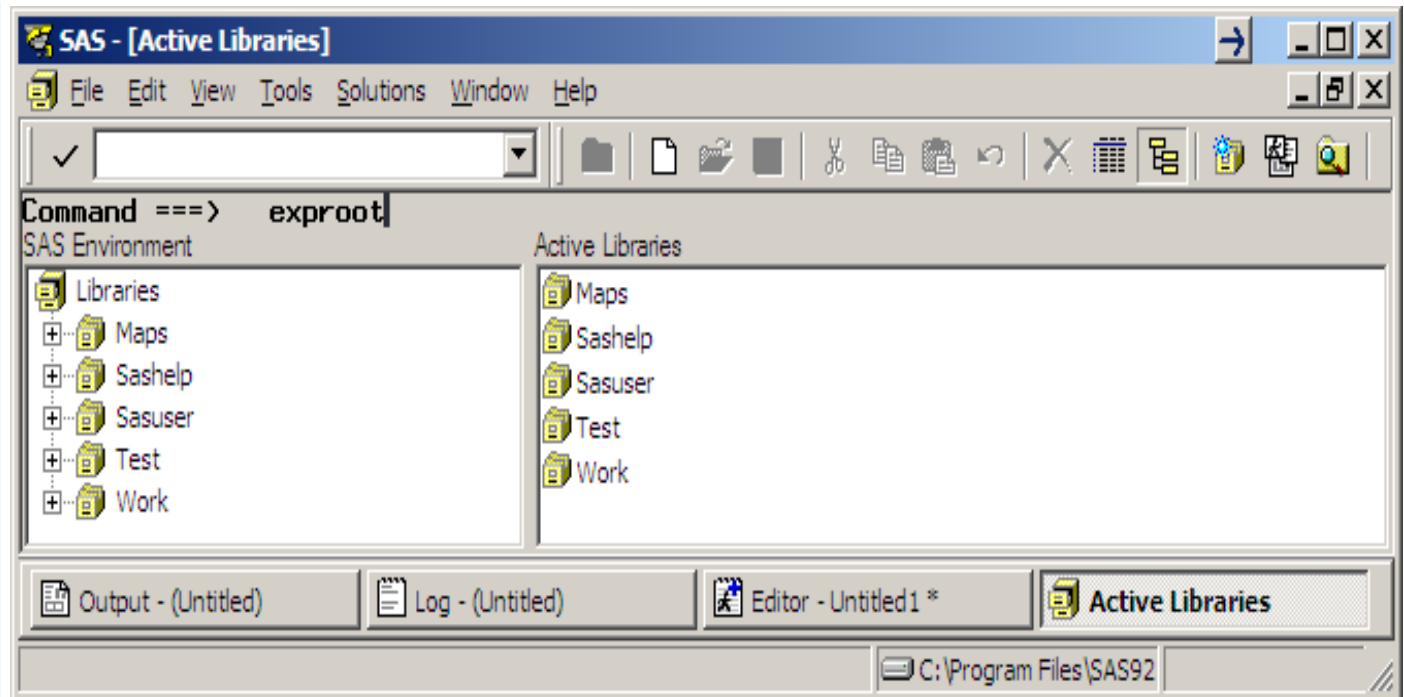
- Execute Tasks within an Interactive SAS session.
- Customizable via Options

OS Command Line

- Execute SAS programs in batch within the OS.
- Customizable via Options.

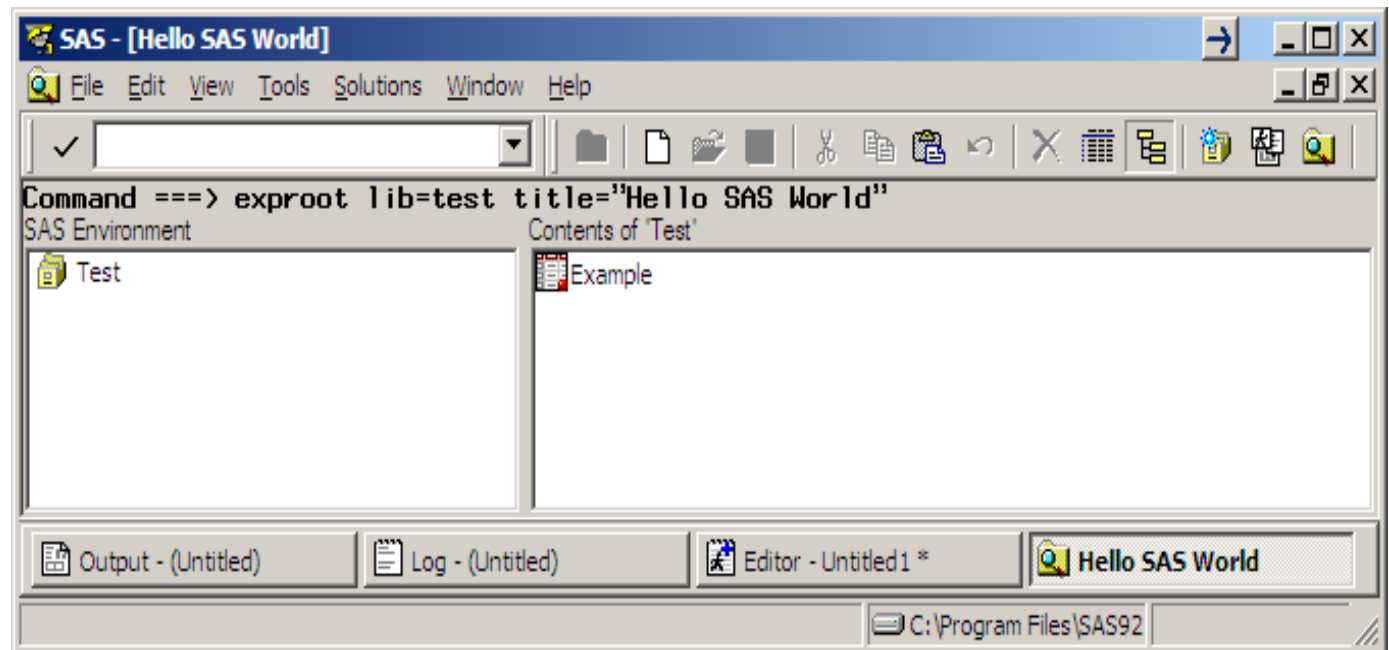
Terminology | SAS Command Line Options

- Example: Command w/o options.

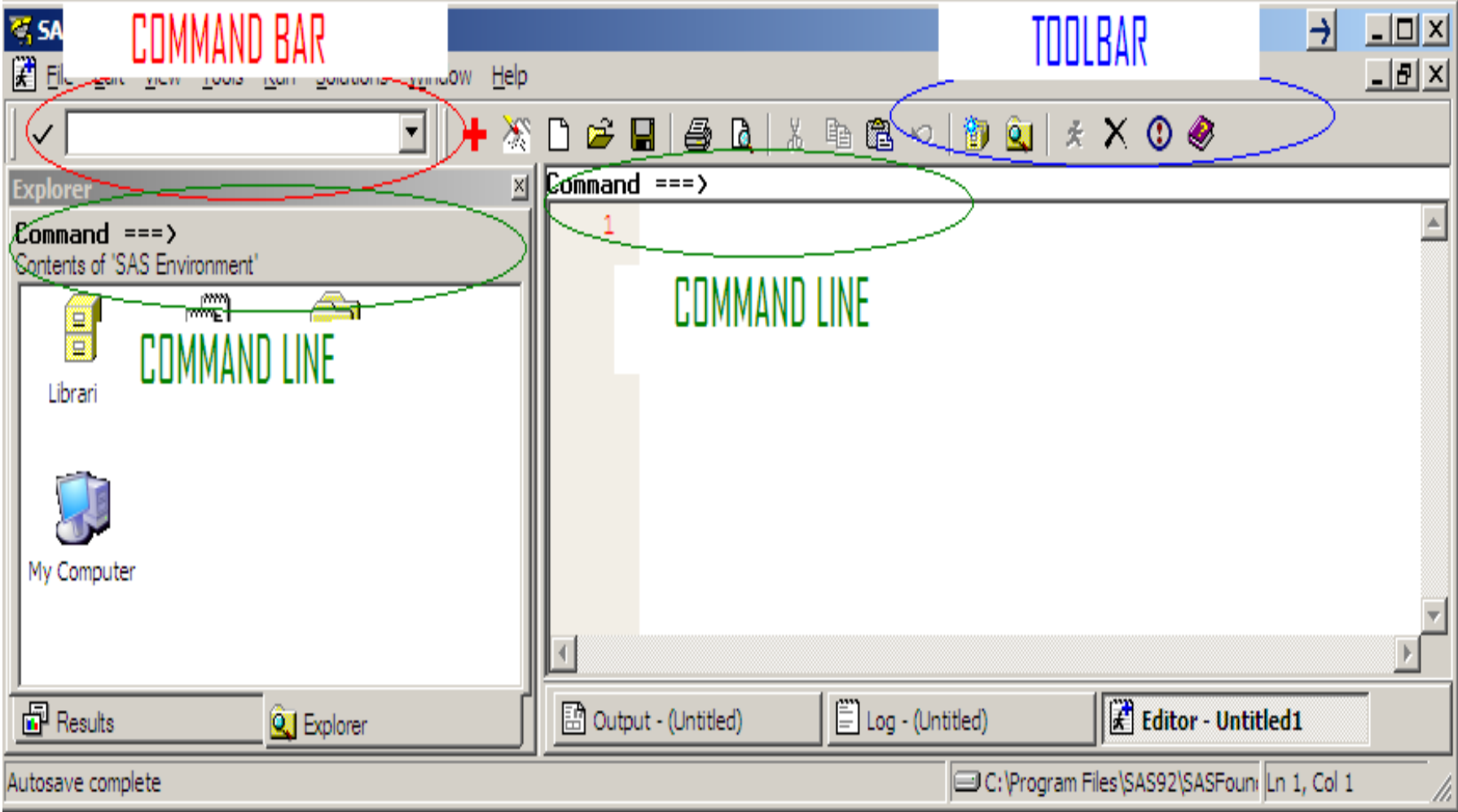


Terminology | Command Line Options

- Options that require user input.
- Enter Options on the right side of the command.
- Offer additional flexibility and efficiency to commands.



Command Submission Locations (CSL)



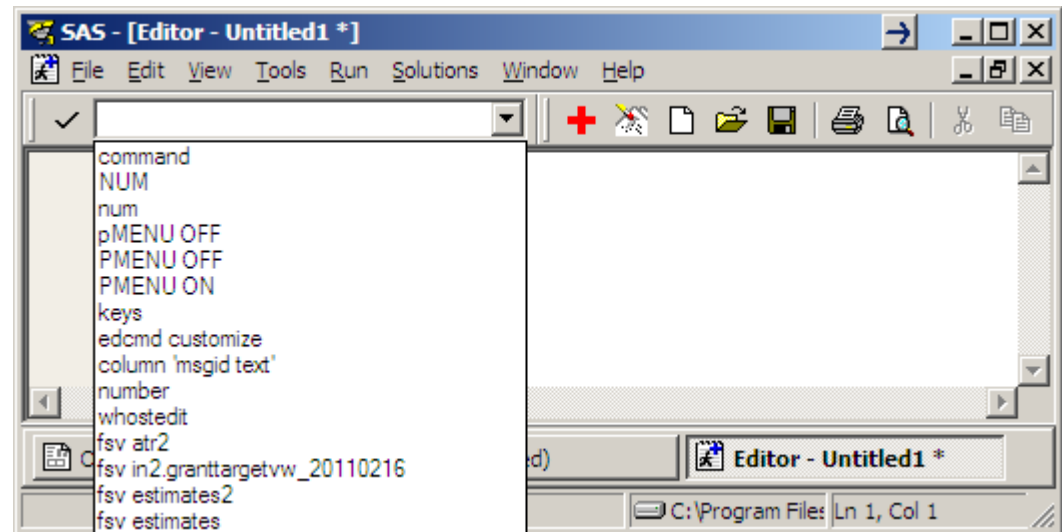
CSL 1 | Command Bar

PROS

- .. Issue Commands at Startup
- .. Drop-Down Menu

CONS

- .. Cognizant of active window



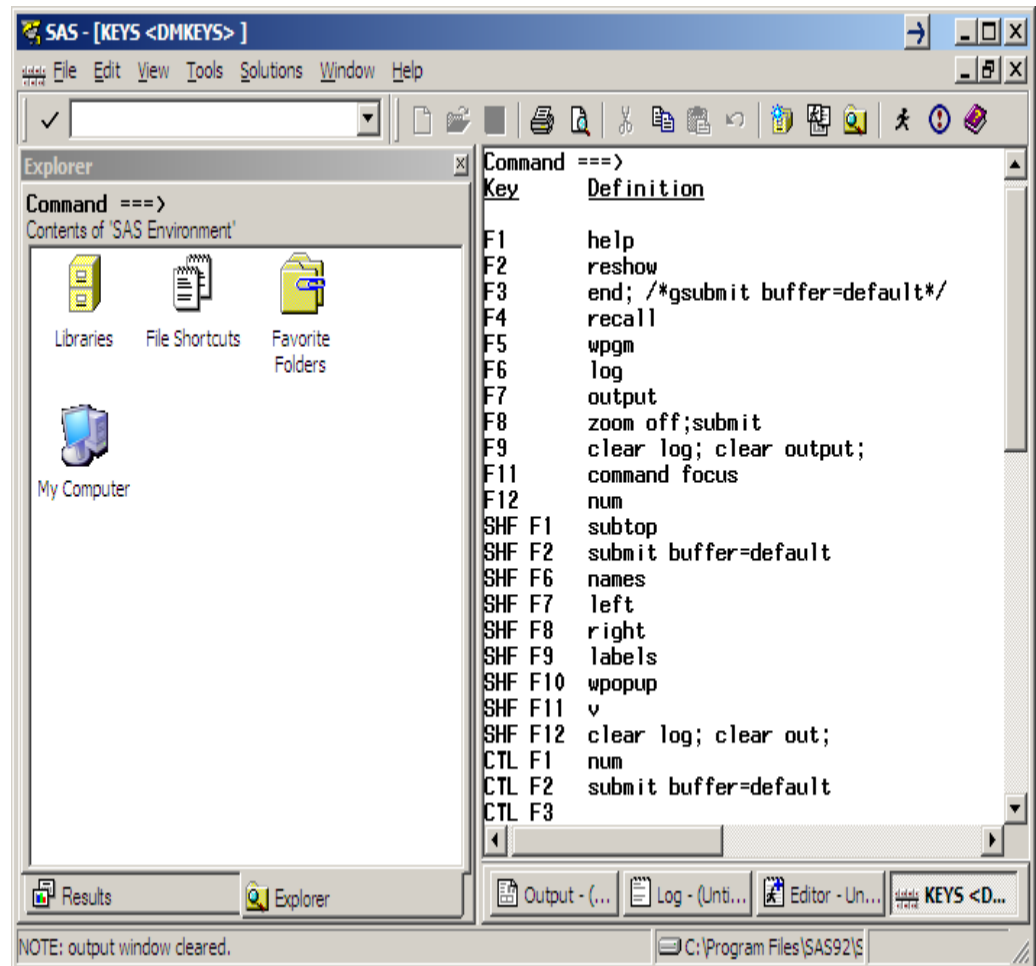
CSL 2 | Keyboard

PROS

- .. Commands Executed Quickly

CONS

- .. Need to remember command/key relationship
- .. Cognizant of Active Window
- .. F Lock



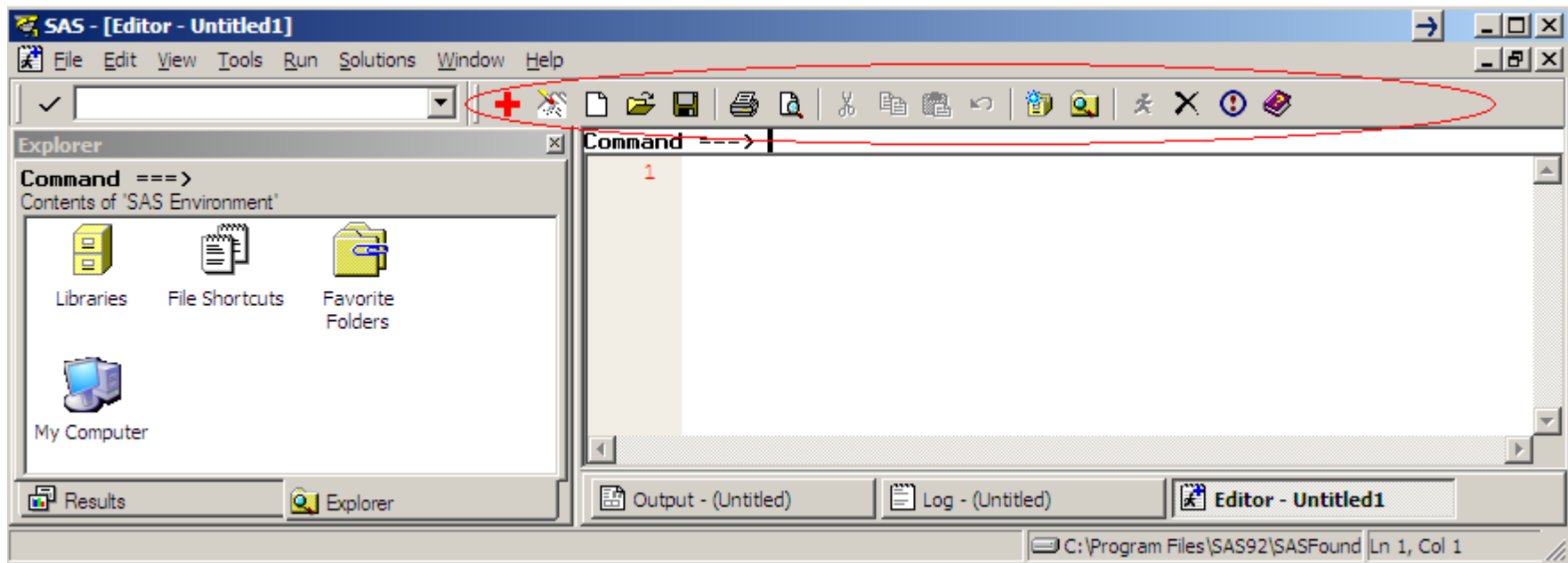
CSL 3 | Toolbar

PROS

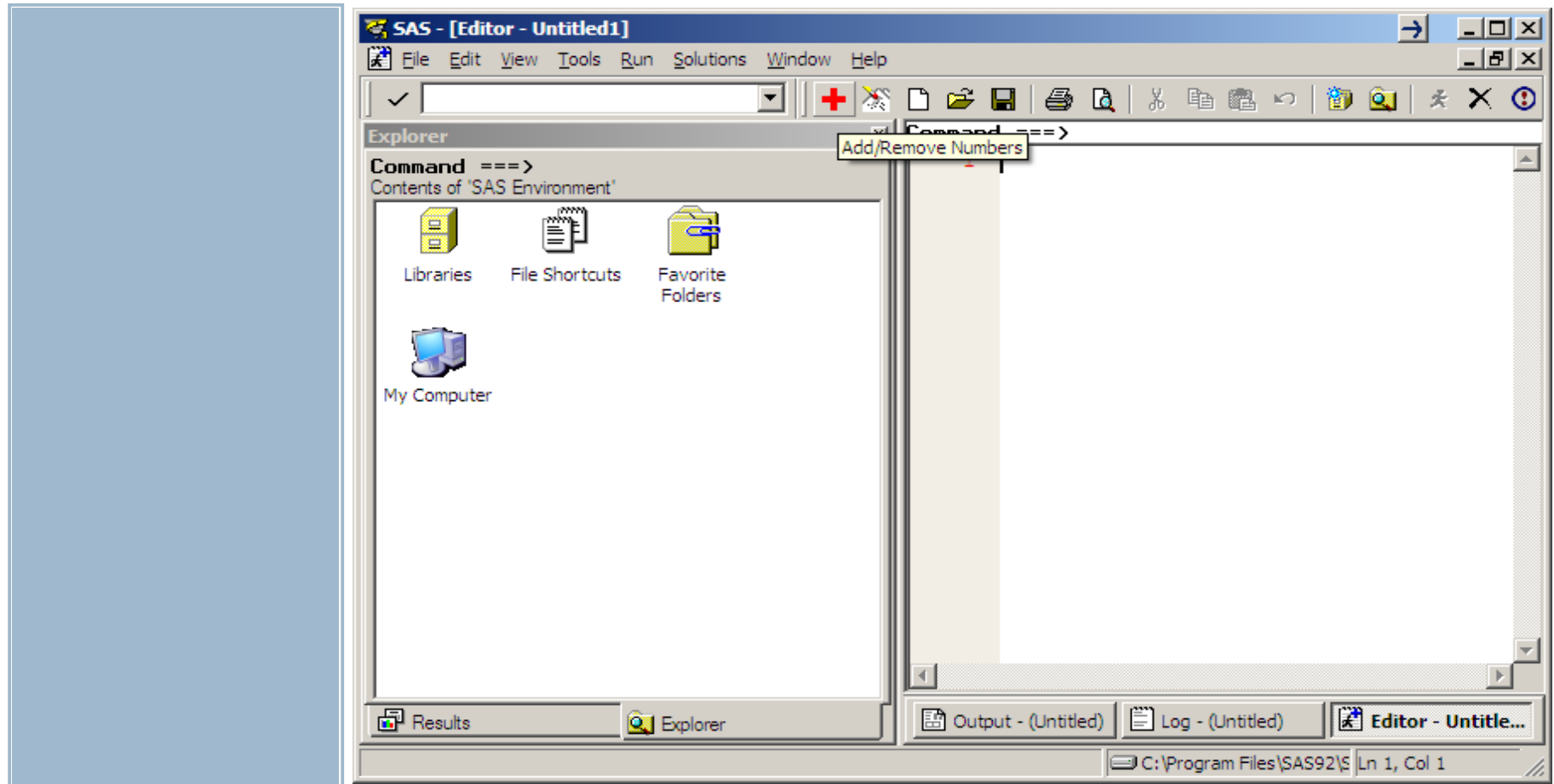
- .. Icon
- .. Hover Text

CONS

- .. Limited by the space on the toolbar

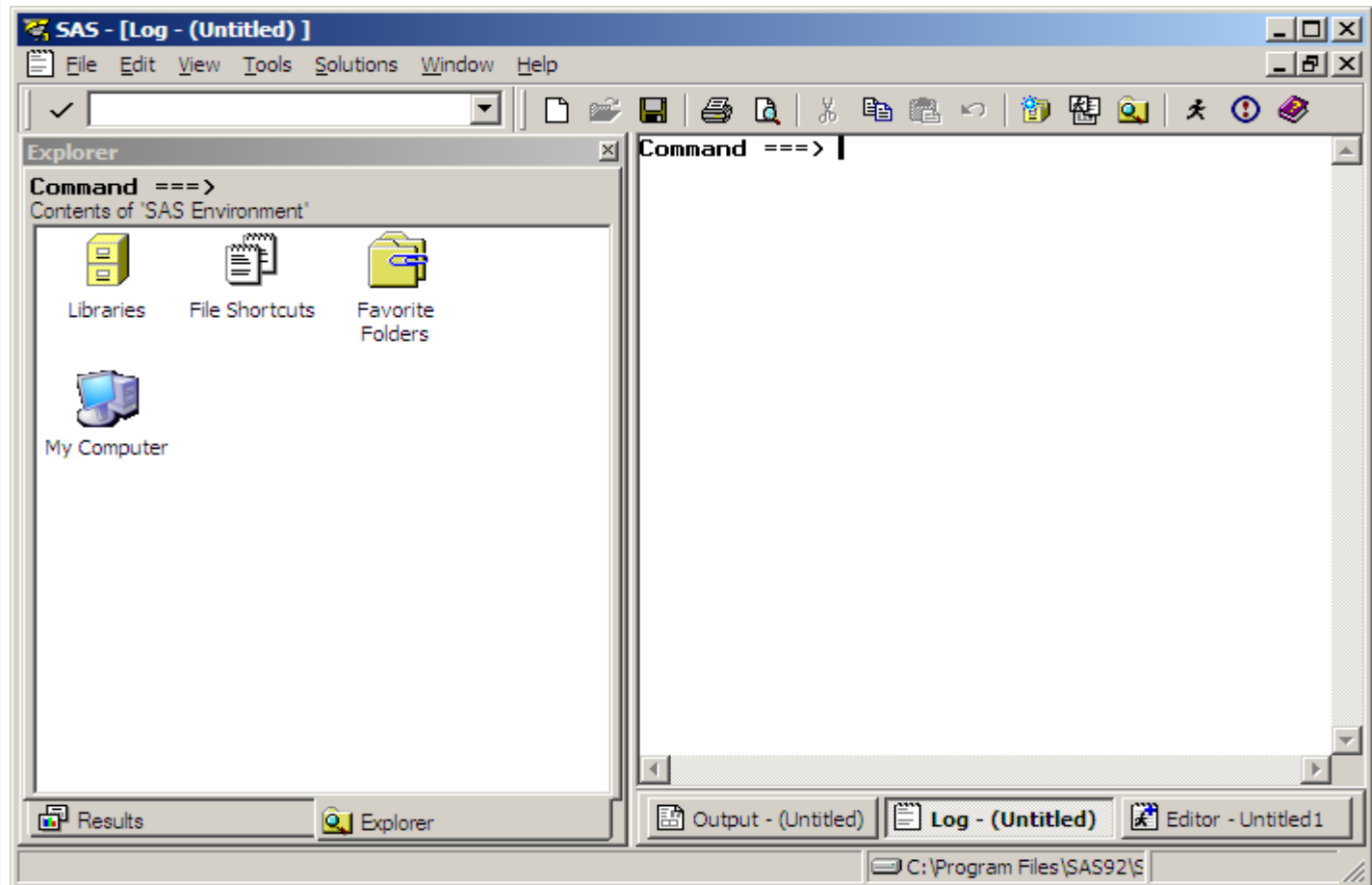


CSL 3 | Toolbar



CSL 3 | Toolbar

- Example:
Change the
Active Window



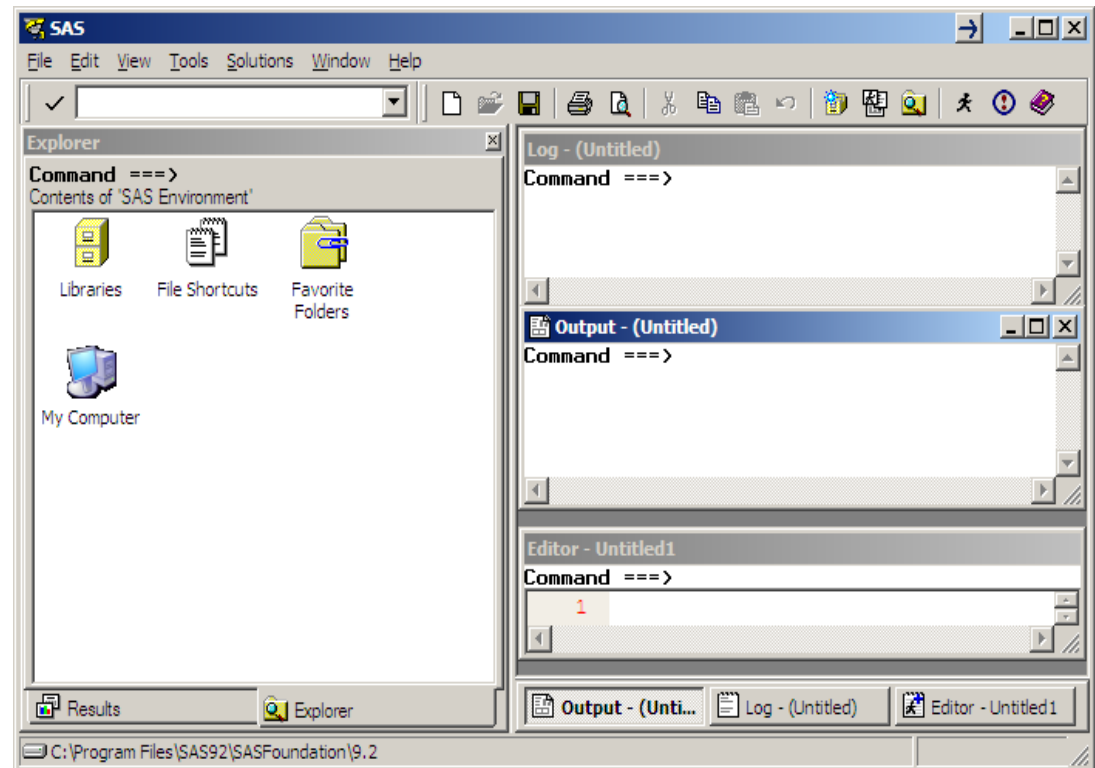
CSL 4 | Command Line

PROS

- .. PREVCMD command accepted

CONS

- .. Cognizant of active window

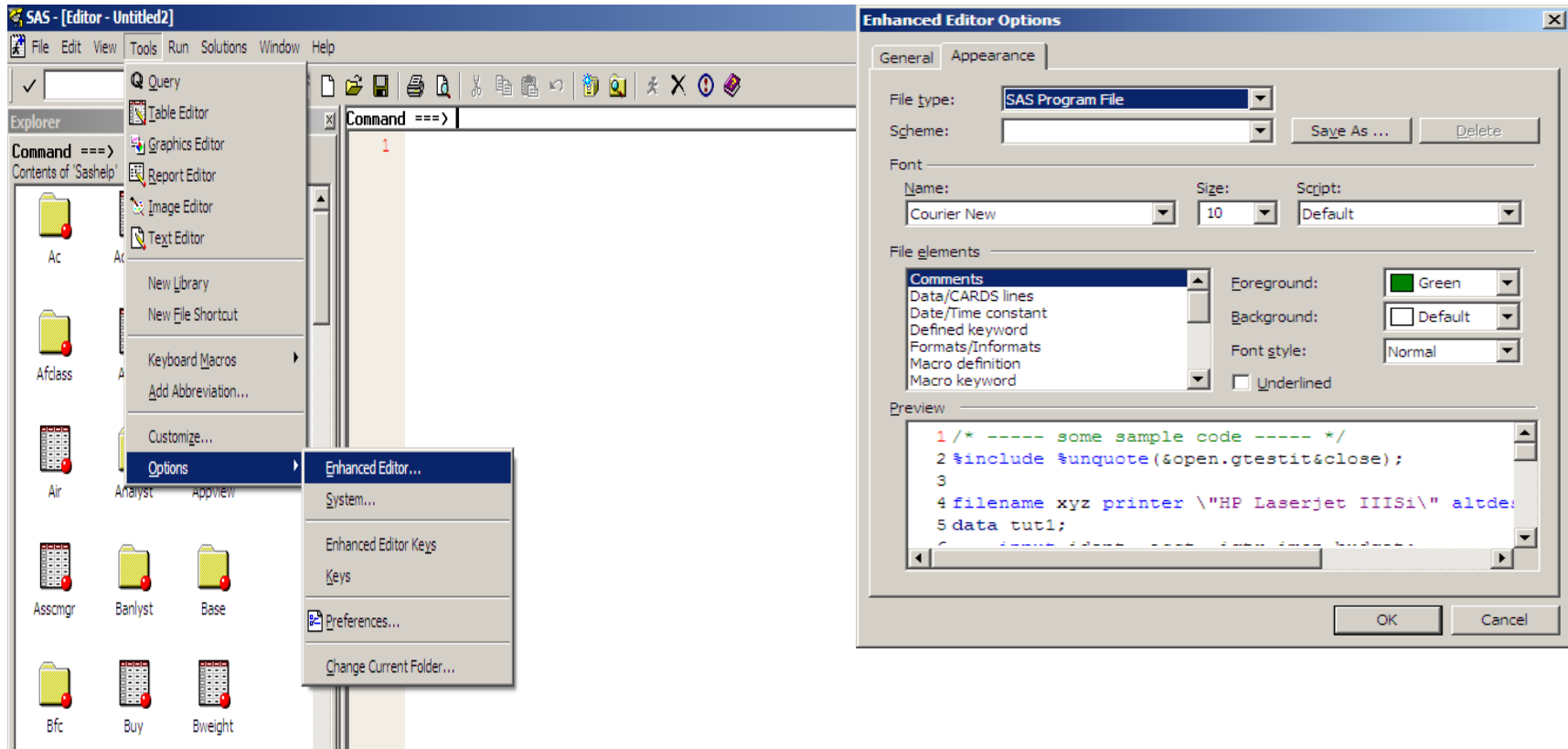


Find the Right Command

- HELP -> SAS Help and Documentation -> SAS Products -> Base SAS -> SAS Command Reference
- HELP -> SAS Help and Documentation -> SAS Products -> Base SAS -> SAS Window Reference
- PREVCMD / ? Command

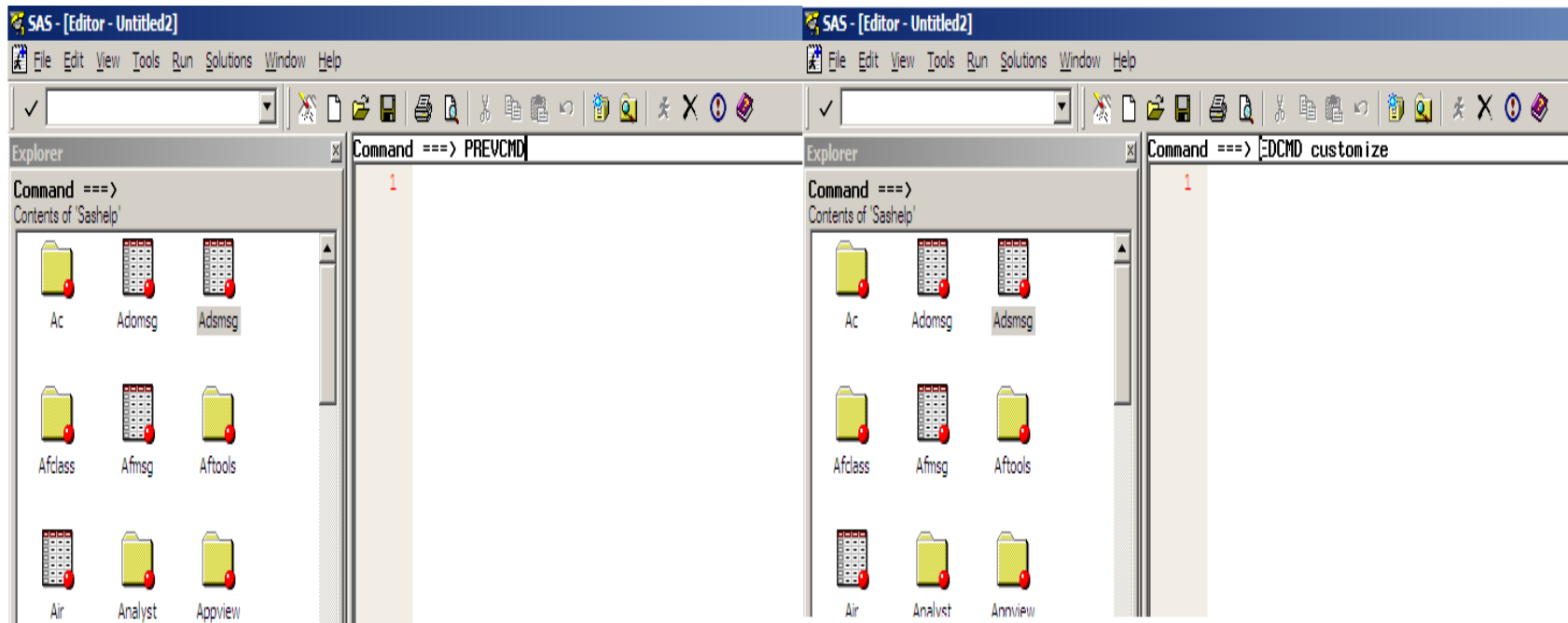
Find the Right Command|

PREVCMD / ? Command



Find the Right Command|

PREVCMD / ? Command



Command-Style Macro|

Introduction

Why is it Useful?

- Macro facility used to create highly-customized commands
- Almost no limits with the SAS command interface

What is it?

- Macro designed to be called from the command line.
- Must use the CMD macro option
- Must use the CMDMAC system option
- Macro needs to be compiled or create an AUTOCALL library before use

Command-Style Macro |

Motivation Revisited

Command Wish List

- .. Crosswalk from values to format labels
- .. Allow for partial format name input
- .. Provide the number of formats that match partial input
- .. Delivery Method – Output Window

SAS Statement Output

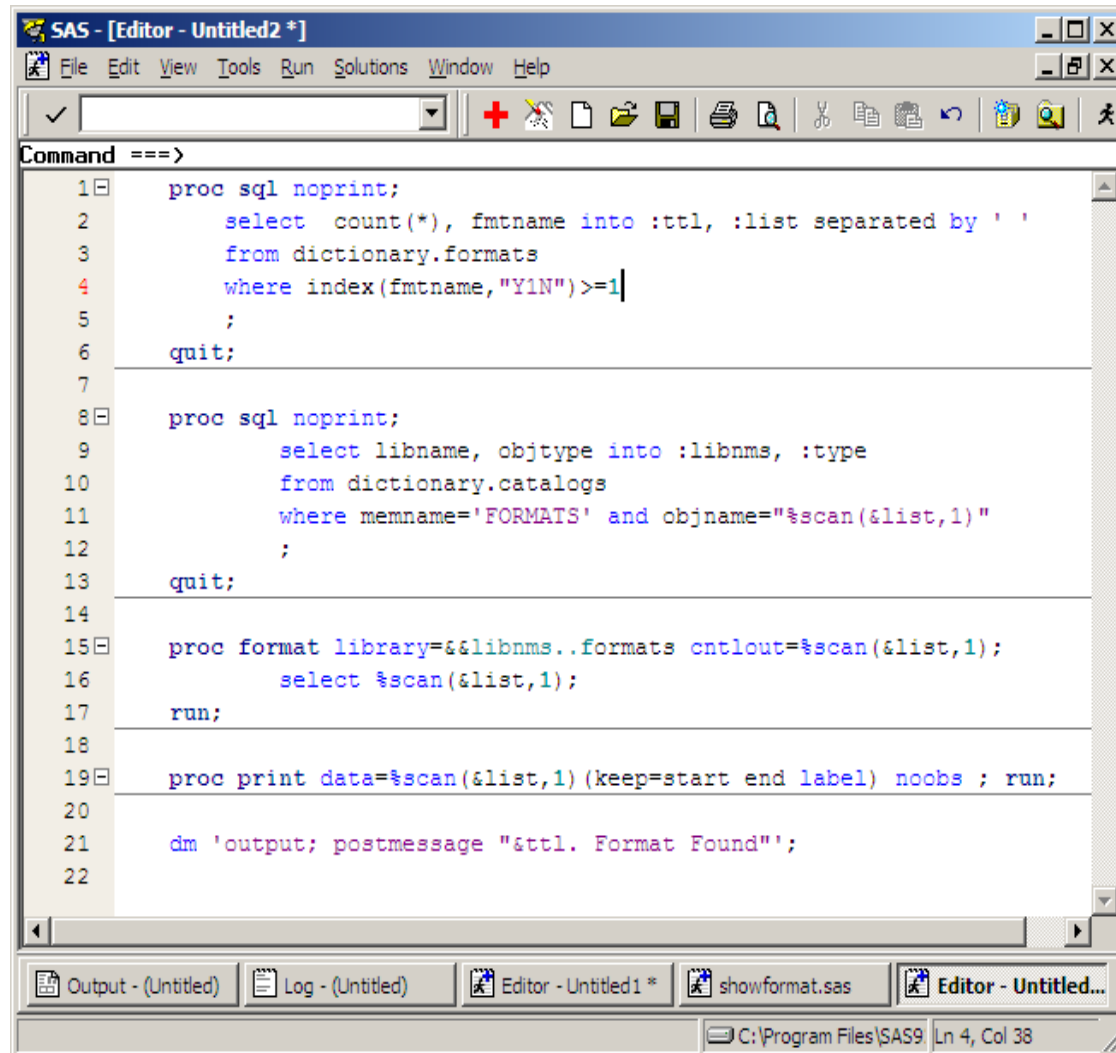


START	END	LABEL
.	.	MISSING DATA
.A	.A	MISSING DATA
.E	.E	MISSING DATA
.G	.G	MISSING DATA
.H	.H	MISSING DATA
.I	.I	MISSING DATA
-9	-9	MISSING DATA
-8	-8	MISSING DATA
-7	-7	MISSING DATA
-5	-5	MISSING DATA
-1	-1	MISSING DATA
23	23	YES
24	24	NO



Command-Style Macro

Motivation Revisited



The screenshot shows the SAS Editor window with the following code:

```
Command ==>
1 proc sql noprint;
2     select count(*), fmtname into :ttl, :list separated by ' '
3     from dictionary.formats
4     where index(fmtname,"Y1N")>=1
5     ;
6 quit;
7
8 proc sql noprint;
9     select libname, objtype into :libnms, :type
10    from dictionary.catalogs
11    where memname='FORMATS' and objname="%scan(&list,1)"
12    ;
13 quit;
14
15 proc format library=&&libnms..formats cntlout=%scan(&list,1);
16     select %scan(&list,1);
17 run;
18
19 proc print data=%scan(&list,1) (keep=start end label) noobs ; run;
20
21 dm 'output; postmessage "&ttl. Format Found"';
22
```

The window title is "SAS - [Editor - Untitled2 *]". The menu bar includes File, Edit, View, Tools, Run, Solutions, Window, and Help. The status bar at the bottom shows "C:\Program Files\SAS9 Ln 4, Col 38".

Command-Style Macro|

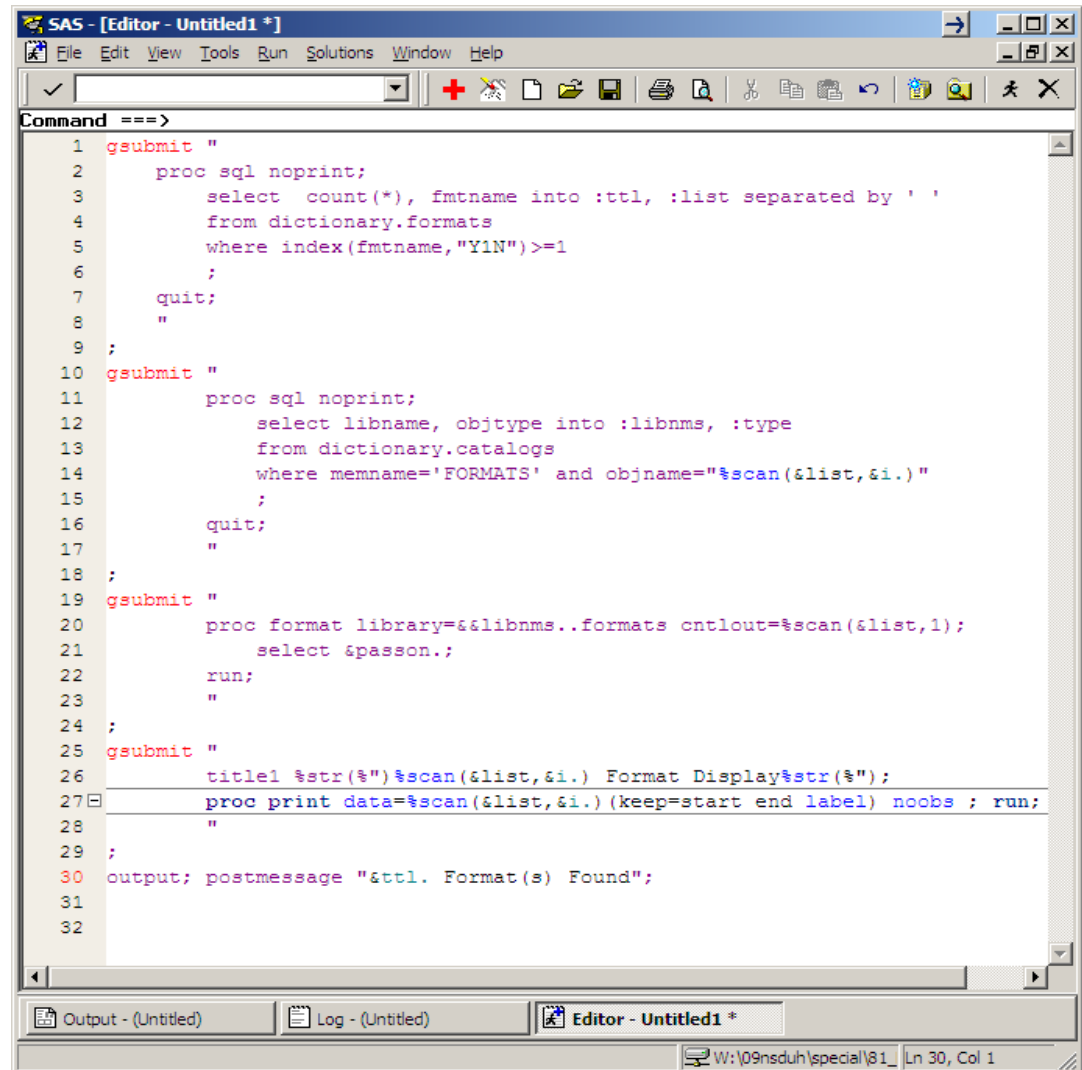
Designed to be Command Line Executable

- Must use one of the following (or a combination)
 - ⊗ Macro Language
 - ⊗ Existing SAS Commands
 - ⊗ SAS statement(s) wrapped in a GSUBMIT command

Command-Style Macro

Designed to be Command Line Executable

- .. SAS Statement(s) wrapped in GSUBMIT command
- .. Syntax: gsubmit " ";
- .. Remove DM statement

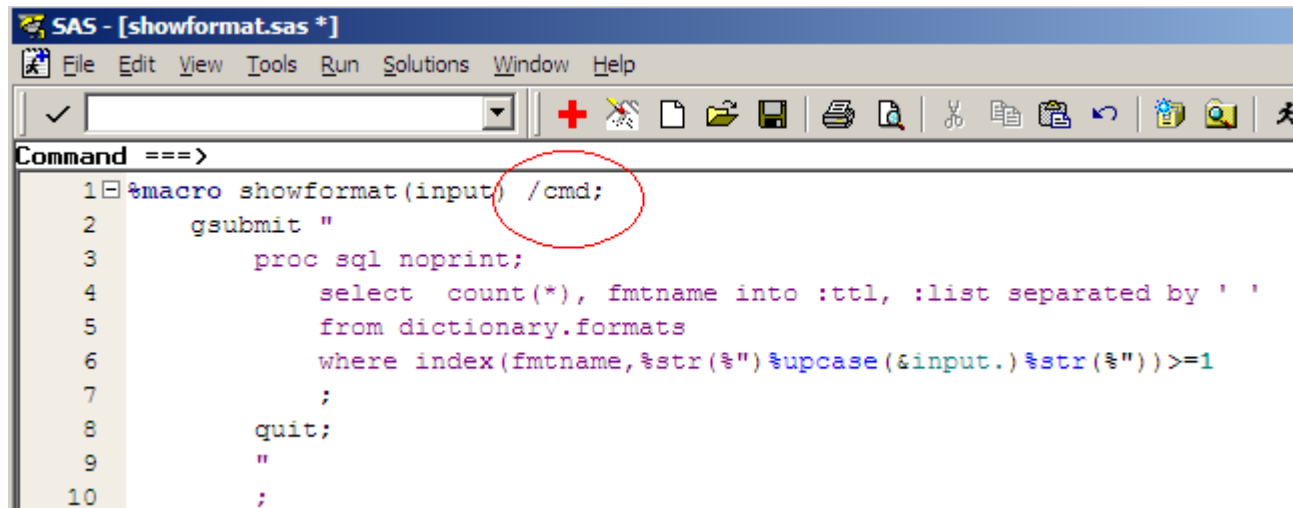


```
SAS - [Editor - Untitled1 *]
File Edit View Tools Run Solutions Window Help
Command ==>
1 gsubmit "
2   proc sql noprint;
3     select count(*), fmtname into :ttl, :list separated by ' '
4     from dictionary.formats
5     where index(fmtname,"Y1N")>=1
6   ;
7   quit;
8   "
9 ;
10 gsubmit "
11   proc sql noprint;
12     select libname, objtype into :libnms, :type
13     from dictionary.catalogs
14     where memname='FORMATS' and objname="%scan(&list,&i.)"
15   ;
16   quit;
17   "
18 ;
19 gsubmit "
20   proc format library=&libnms..formats ontlout=%scan(&list,1);
21     select $passon.;
22   run;
23   "
24 ;
25 gsubmit "
26   title1 %str(%)%scan(&list,&i.) Format Display%str(%) ;
27   proc print data=%scan(&list,&i.) (keep=start end label) noobs ; run;
28   "
29 ;
30 output; postmessage "&ttl. Format(s) Found";
31
32
```


Command-Style Macro

CMD Macro Option

- “Specifies that the macro can accept either name-style or command-style invocation”.
 - Command==> ShowFormat Y1N
 - Command==> %ShowFormat Y1N



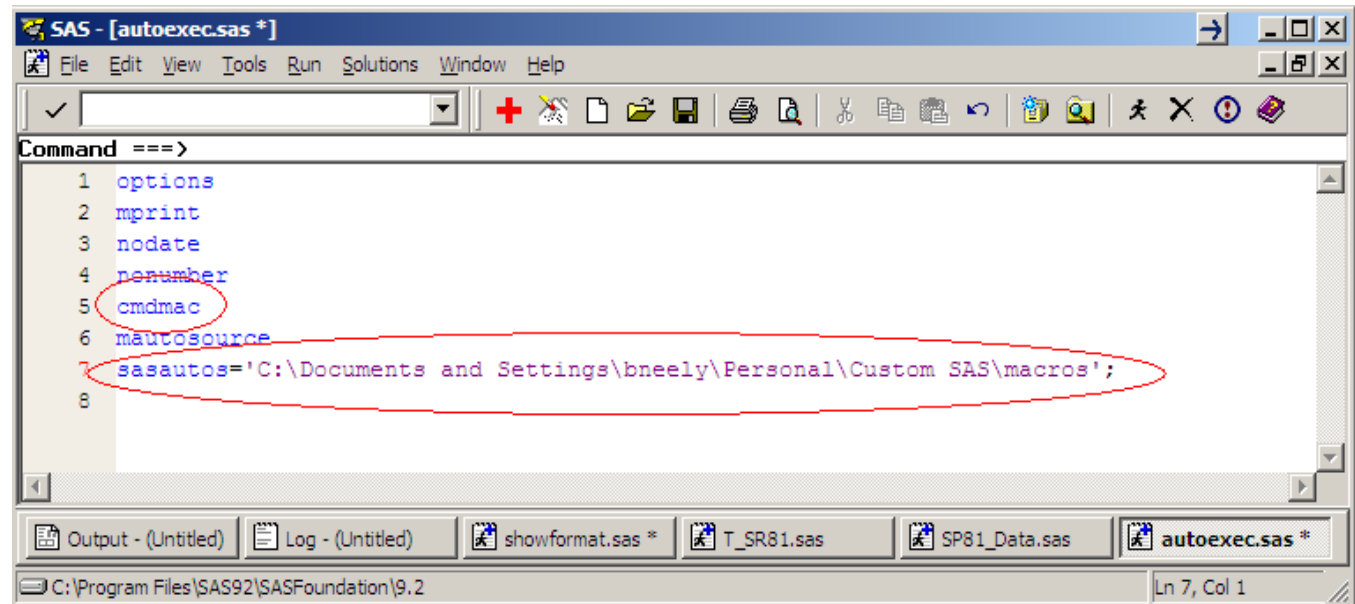
```
SAS - [showformat.sas *]
File Edit View Tools Run Solutions Window Help
Command ==>
1 %macro showformat(input) /cmd;
2   gsubmit "
3     proc sql noprint;
4       select count(*), fmtname into :ttl, :list separated by ' '
5       from dictionary.formats
6       where index(fmtname, %str("%" %supcase(&input.) %str("%")) >= 1
7     ;
8   quit;
9   "
10  ;
```

Command-Style Macro|

CMDMAC System Option / Autocall Library Option

- CMDMAC system option – macro processor scans the 1st word of every SAS command

- Autocall Library and mautosource, manually compiling the macro is unnecessary.



Before we try ShowFormat....

Questions?