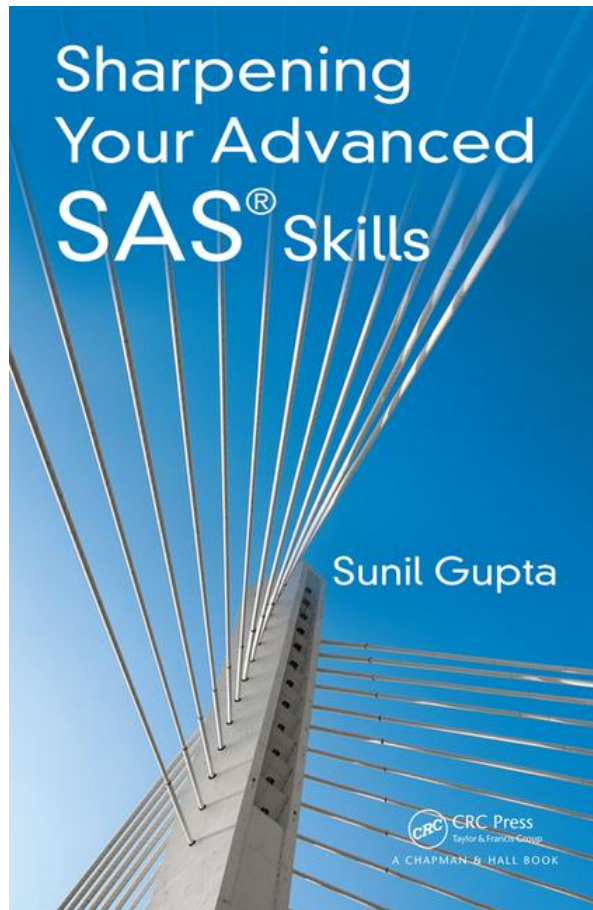


Five Habits of Highly Effective SAS Programmers

Sunil Gupta

Associate Director,
Statistical Programming
Cytel

SASSavvy.com



Listen Carefully and Ask Questions: What exactly are your requirements?



AS PROPOSED BY THE
PROJECT SPONSOR.



AS SPECIFIED IN THE
PROJECT REQUEST.



AS DESIGNED BY THE
SENIOR ANALYST.



AS PRODUCED BY THE
PROGRAMMERS.



AS INSTALLED AT THE
USER SITE.

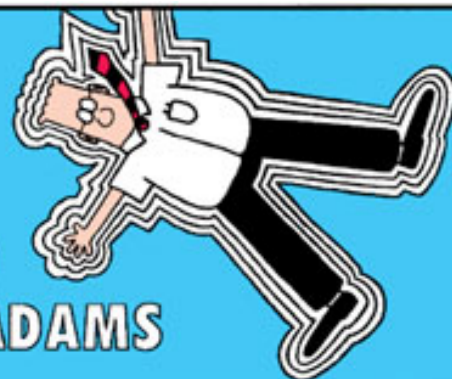


WHAT THE USER WANTED.



DILBERT[®]

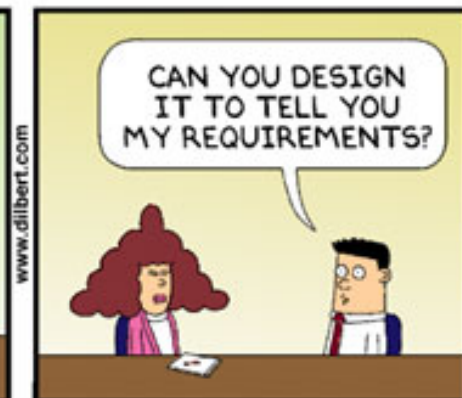
BY
SCOTT ADAMS



E-mail: SCOTTADAMS@AOL.COM



© 2006 Scott Adams, Inc./Dist. by UFS, Inc.



www.dilbert.com

Five Habits of Highly Effective SAS Programmers

**1) Programming / QC
Standards**

**2) Communication /
Organization /
Documentation**

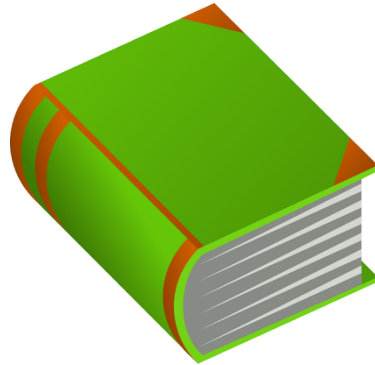
**5) Best
Practices**

**3) Master the Basics /
Continuous Learning**

**4) Passion for
Excellence /
'Outsource' Mentality**

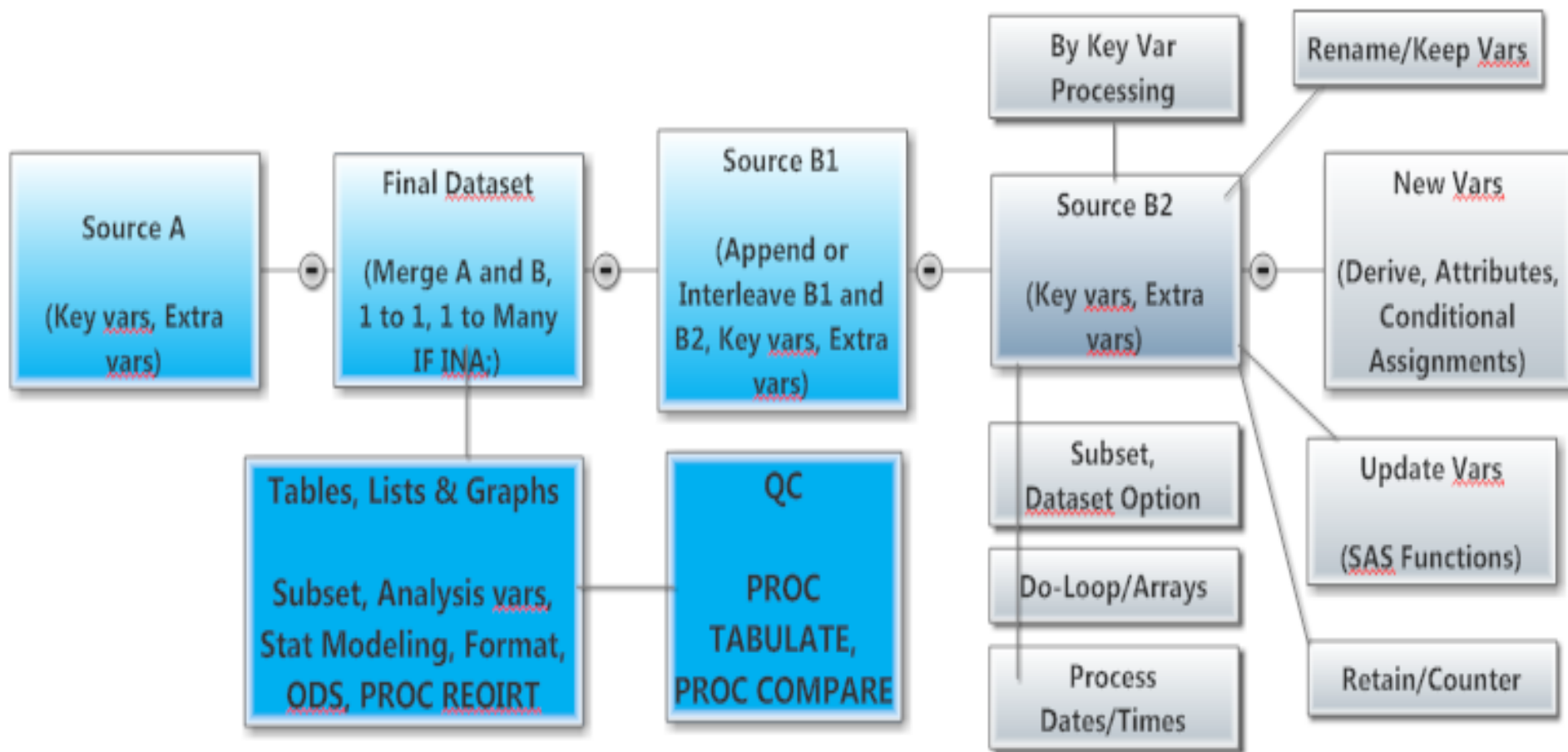
1) Programming / QC Standards:

What makes a good novel/SAS program?



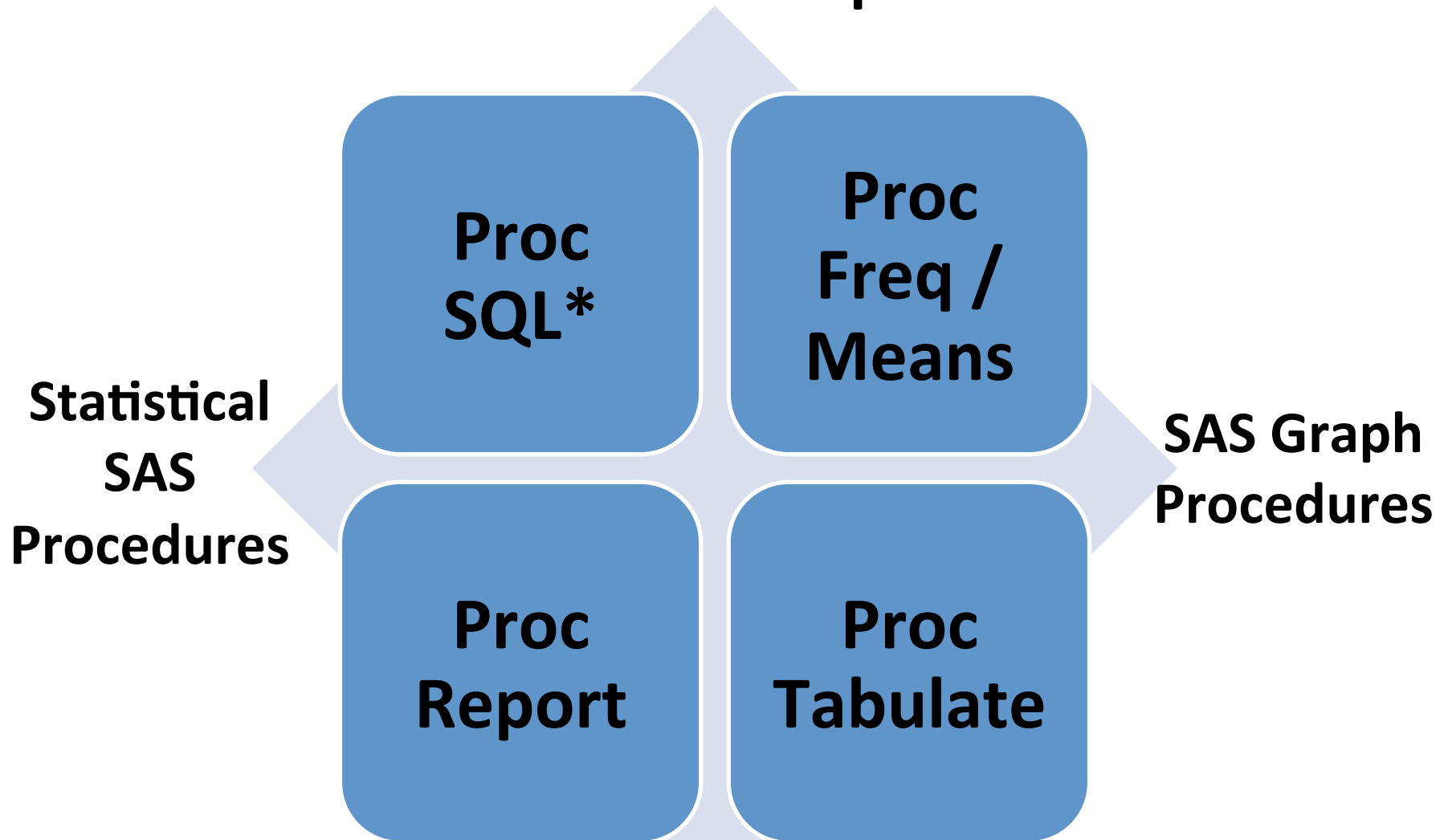
Good Novel	Good SAS Program
Complete Introduction	
Chapters make it easier to understand	
Paragraphs provide the details	
Storyline makes sense from one event to next event	
Conclusion/Summary gives us closure	

The Power of Data Step Programming



See more SAS Mind Maps at SASSavvy.com

Master the Top Four SAS Procedures with common options



* Because Proc SQL may generate incorrect results without any warnings, caution should be used when writing complex, multi-tasking Proc SQL code.

Hidden SAS Gem:

What *is* PROC TABULATE??

Proc Print

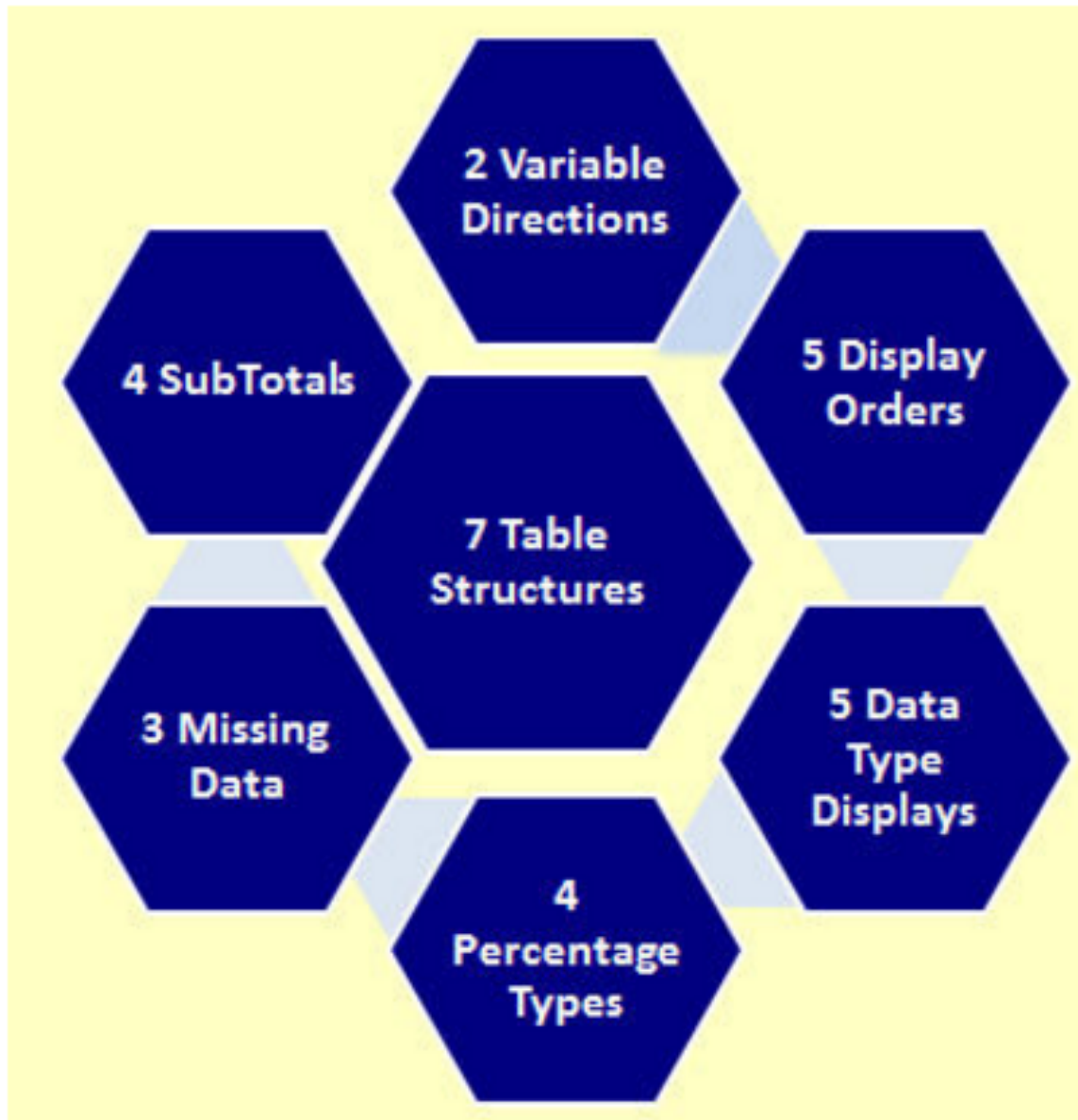
Proc Freq

Proc Tabulate

Proc Means

Proc
Summary

Proc Tabulate Gallery with over 30 SAS Examples



		Total
Age	N	19
	Mean	13.3
	Std	1.5
	Median	13.0
	Min	11.0
	Max	16.0
	Q1	12.0
	Q3	15.0
Sex		
F	N	9
	%	47.4
M	N	10
	%	52.6
Subtotal	N	19
Height	N	19
	Mean	62.3
	Std	5.1
	Median	62.8
	Min	51.3
	Max	72.0
	Q1	57.5
	Q3	66.5
Weight	N	19
	Mean	100.0

Proc Tabulate is great to QC most any

table line both continuous and categorical results in one table

```
proc tabulate data=class order=data missing format=5.1;
```

```
class sex/preloadfmt; var sex_cnt age height weight;
```

```
keylabel N='N' PCTN='%' COLPCTN='%' MEAN='Mean'
MEDIAN='Median' MIN='Min' MAX='Max' STD='Std'
Q1='Q1' Q3='Q3' ALL='Subtotal';
```

```
tables ( sex ), sex_cnt*(n colpctsum)/ rts=75 printmiss
misstext='0';
```

* For variable with unique values that create multiple columns;

```
* tables ( sex ), (&colvar*sex_cnt)*(n colpctsum)/ rts=75
printmiss misstext='0';
```

```
run;
```

SAS Tip: Remove duplicate records, keep non-missing numeric values and rename variables

Address Issues

- * 1 – Process a dataset without hard coding 10 numeric variables that contain summary results;
- * 2 – Remove duplicate records that are not possible with PROC SORT with NODUPKEY option;
- * 3 – Keep non-missing values from each variable;
- * 4 – Keep dataset with the original variable names without hard coding variable names;

Three Step Process

- * 1 - Get non-missing data using PROC MEANS;
- * 2 - Save as macro variables using PROC SQL - original and means names;
- * 3 - Rename variables back to original variable names using macros and do-loop;

SAS Tip: Remove duplicate records, keep non-missing numeric values and rename variables

* 1 - Get non-missing data and use same variable name_MAX and position notation;

```
proc means data=adbm7 nway noprint;  
class usubjid paramcd;  
var basedt -- ttrtd0;  
output out=adbm8 (drop= _type_ _freq_) max=/autoname;  
run;
```

* 2 - Save as macro variables - original and means names from metadata;

```
proc sql;  
select unique name into: onamlst separated by '.' from sashelp.vcolumn  
where libname='WORK' and memname='ADBM7' and name ^in ('USUBJID'  
'PARAMCD');  
select unique name into: mnamlst separated by '.' from sashelp.vcolumn  
where libname='WORK' and memname='ADBM8' and name ^in ('USUBJID'  
'PARAMCD');  
quit;  
%put &onamlst &mnamlst;
```

SAS Tip: Remove duplicate records, keep non-missing numeric values and rename variables

*** 3 - Rename variables back to original variables names with macro do-loop and scan function;**

```
%let rnval=1;
```

```
%macro rnvar;
```

```
data nci_adam.adbmks;
```

```
set adbmks;
```

```
%do %while(%length(%scan(&onamlst, &rnval, %str('.'))) > 0);
```

```
  rename %scan(&mnamlst, &rnval, %str('.')) = %scan(&onamlst, &rnval, %str('.'));
```

```
  %let rnval = %eval(&rnval + 1);
```

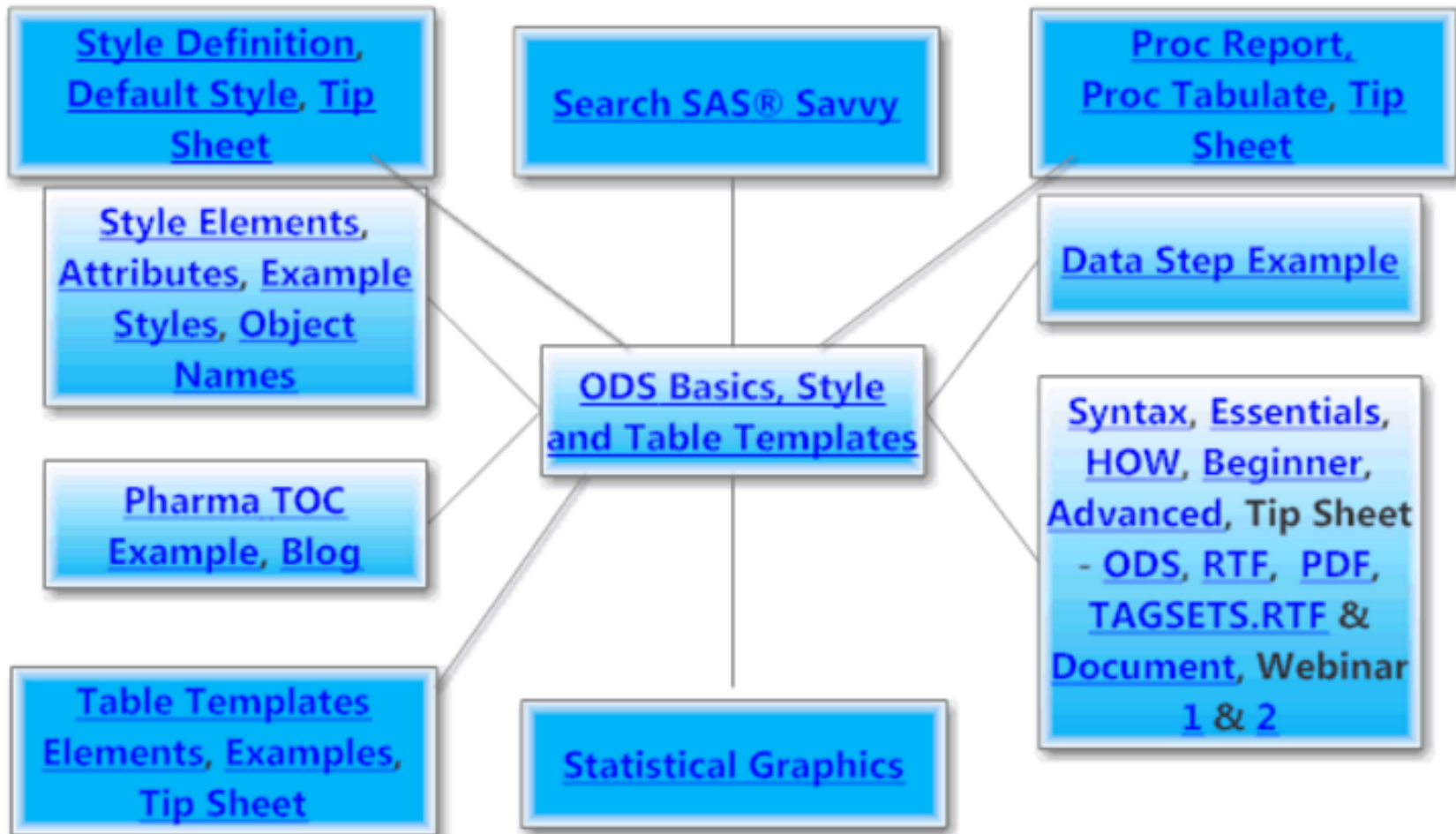
```
%end;
```

```
run;
```

```
%mend rnvar;
```

```
%rnvar;
```

Applying ODS Styles and Templates



A World without SAS Macros (&, %)



What You can Expect *Without* SAS Macros:

- ✓ No order to get common tasks completed
- ✓ No control over the time to deliver
- ✓ No assurance of the quality of work
- ✓ No one in control to correct issues or errors
- ✓ No method to diagnose issues or errors
- ✓ No method to grow or build on knowledge
- ✓ Loss of client confidence
- ✓ Loss of business

Anatomy of SAS Macros: Three Coding Sections

User Input

Define Input Parameters with Defaults

- Use global macro variables and system functions

Macro Purpose

Data Process within Macro Loop

- Use metadata and collect or report findings

Error Messages

Defensive Programming of Input Parameters and Data Processing

- Display user message or abort if invalid

Example of SAS Macro with Four Nested Loops

1. Stratification Group (Dose 1, Dose 2)

2. By Variable (Gender, Age Groups)

3. Subset Population (SAF, ITT)

4. Analysis Variable (height, weight)

Core SAS Processing Steps

Alternative is to hard code and maintain 16 combinations of exploratory analysis

SAS Programmers can be clever with Macros

FOXTROT By Bill Amend

```
#include <stdio.h>
int main(void)
{
    int count;
    for (count = 1; count <= 500; count++)
        printf("I will not throw paper airplanes in class.");
    return 0;
}
```

AMEND 10-3

NICE TRY.



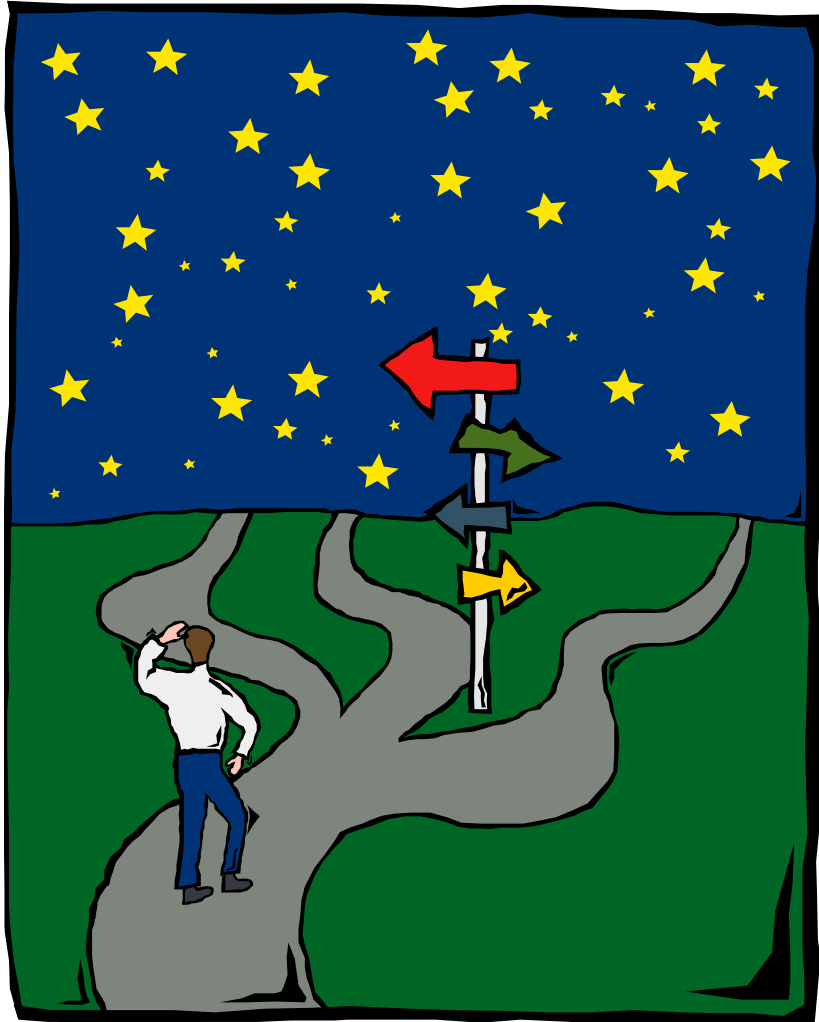
© 2001 Bill Amend. Distributed by Universal Uclick.

2) Communication / Organization / Documentation: Better Project Management

1. Technical training with hands-on activities for on-boarding and monthly, engage team to present code reviews
2. Mentor program to guide and answer questions
3. IT support to address any system related issues
4. Resource Planning and Management
5. Communicate weekly conference calls
6. Central project tracking sheet and issues log for better project management
7. Follow and maintain process flow charts and task steps including - SDTM and ADaM specifications, create/qc SDTM, ADaM and TFLs
8. CC on all emails with client



3) Master the Basics / Continuous Learning: Certification Career Paths



Find the Certification that fits Your Needs and Goals

- ✓ Base Certification
- ✓ Advanced Certification
- ✓ Clinical Trials Programmer
Certification

- ✓ Enroll in an online class

sas®

**Certified Base
Programmer**

DATA Step vs. PROC SQL Camps



DATA Step Die-Hards

Factors: First learned in SAS, First encounter of flexibility, First frustration of complexity



PROC SQL Die-Hards

Happy Camper – Use the best of both worlds



SAS Display Manager World	SAS Enterprise Guide World
Technical	Non-Technical, Technical
DATA Step Programming, Requires More Time for Programming	Same DATA Step Programming , Requires Less Time using up to 90 Tasks
One Dataset/Program/Output File per window	Cross Reference/ Query Multiple Intermediate Datasets /Programs/Output Files
Open <u>Programmer's</u> working environment (Interactive – DATA Step Debugger)	Controlled <u>Analyst's</u> working environment (Defaults, Batch Processing)
Non-Visual Programming	Visual Process Flow

What If SAS EG Could?



- Help with querying and validating?
exploration and management

- Help with documentation?
portable package format (Notes, SAS Program, List,
Log and Output file)

More

- Enforce standards and automation?

Fully compatible with SAS macros and menus

- Help with prototyping table layouts and performing
complex statistical analysis?

Yes

with feature to create custom tasks

SAS Programming: Leaving a Trail to Follow



From Raw to Final Dataset

- ✓ Keep intermediate datasets to leave a trail of progress instead of updating the same temporary dataset. This makes it easier to diagnose issues and errors.
- ✓ Use a system with letters and numbers to name intermediate datasets to better maintain SAS program.
- ✓ SAS Enterprise Guide makes it easy to track and document progress.

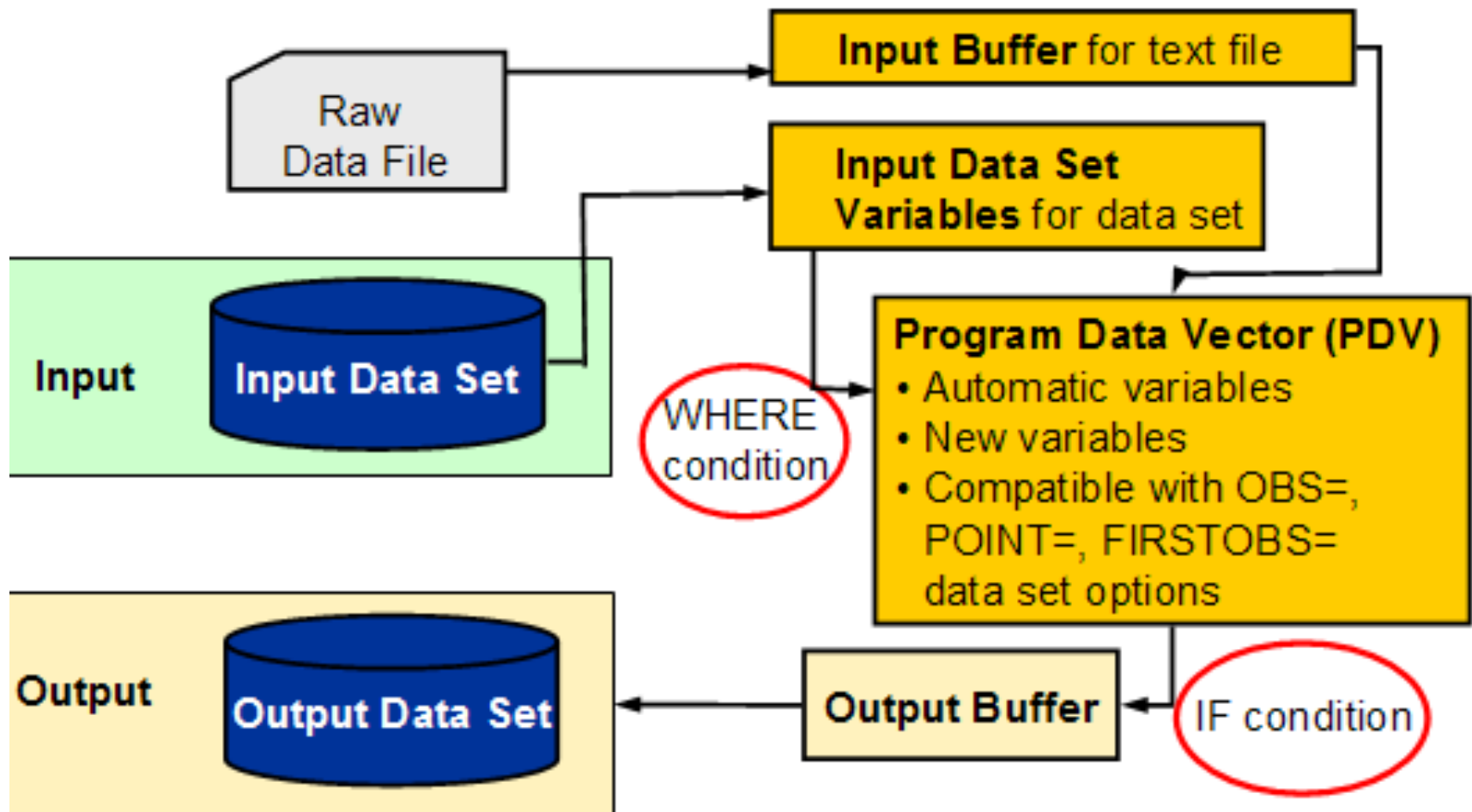
Understanding Differences between Where vs. If

Usage Note 24286: When do I use a WHERE statement instead of an IF statement to subset a data set?

Details

About

Rate It



Understanding Differences between PUT/INPUT

Need for PUT(X, Y) and INPUT(X, Y) Functions to process numbers

To: Output Variable **Type Expression**

From: Input Data Type	New: Character Variable	New: Numeric Variable
Character Variables w/ chars or <u>nums</u>	❶ PUT(X, Y)	❸ INPUT(X , Y)
Numeric Variables w/ <u>nums</u>	❷ PUT(X , Y)	PROC FORMAT;

Numeric variables used in character
variable expression or format.

Issue Ex.

213 || ' Goldenwood Circle';

Numbers stored in
character variables
used in numeric
calculations.

Issue Ex. '2' + '2' = 4;

Do you believe in magic??



- ✓ Using GPS to travel the shortest distance
- ✓ Cooking with recipes
- ✓ Using To-Do Lists
- ✓ Using calculators
- ✓ Getting organized with Excel files
- ✓ Extreme Multi-tasking with Proc SQL to save steps
- ✓ Add group descriptive statistics without changing the original dataset structure (record sort, variable order) or content
- ✓ Ideal for calculations such as change from group mean baseline

What is Group Descriptive Statistics?

	Name	Sex	Age	Height	Weight	sexn
1	Judy	F	14	64.3	90	Count of females
2	Jane	F	12	59.8	84.5	
3	Joyce	F	11	51.3	50.5	
4	Barbara	F	13	65.3	98	
5	Carol	F	14	62.8	102.5	
6	Mary	F	15	66.5	112	
7	Louise	F	12	56.3	77	
8	Alice	F	13	56.5	84	
9	Janet	F	15	62.5	112.5	
10	Philip	M	16	72	150	Count of males
11	James	M	12	57.3	83	
12	Henry	M	14	63.5	102.5	
13	John	M	12	59	99.5	
14	William	M	15	66.5	112	
15	Alfred	M	14	69	112.5	
16	Jeffrey	M	13	62.5	84	
17	Thomas	M	11	57.5	85	
18	Ronald	M	15	67	133	
19	Robert	M	12	64.8	128	

How would you create sexn variable? How many steps would it take?

What if you wanted to add min and max of Weight by Sex?

At least three steps – 1) Output dataset from Proc Freq, 2) Proc Sort both datasets and 3) Data step merge by sex.

Proc SQL Subquery Linked by Group Variable

```
proc sql; create table class2 as
```

```
select a.*,
```

```
b.sexn
```

```
from sashelp.class as a
```

```
left join
```

```
(select
```

```
sex,
```

```
count(sex) as sexn
```

```
from sashelp.class where sex > ' '
```

```
group by sex
```

```
) as b on
```

```
a.sex=b.sex;
```

```
quit;
```

1. Select all master variables

2. Include the new group descriptive statistics variable

3. Apply LEFT JOIN to keep all master records

4. Create internal dataset using subquery

5. Select or create grouping variable

6. Apply one or more summary functions

7. Link internal dataset by grouping variable

Save two steps!

4) Passion for Excellence: Stay Informed of SAS Tips

Something for nothing? Adding group descriptive statistics



Sunil Gupta | JUNE 20, 2012

8533



6

Tweet

0

G+1

0



Like

3



Share

24

Can you actually get something for nothing? With PROC SQL's subquery and remerging features, yes, you can.

Converting variable types—use PUT() or INPUT()?



Sunil Gupta | MAY 1, 2015

4786



5

Tweet

14

G+1

0



Like

3

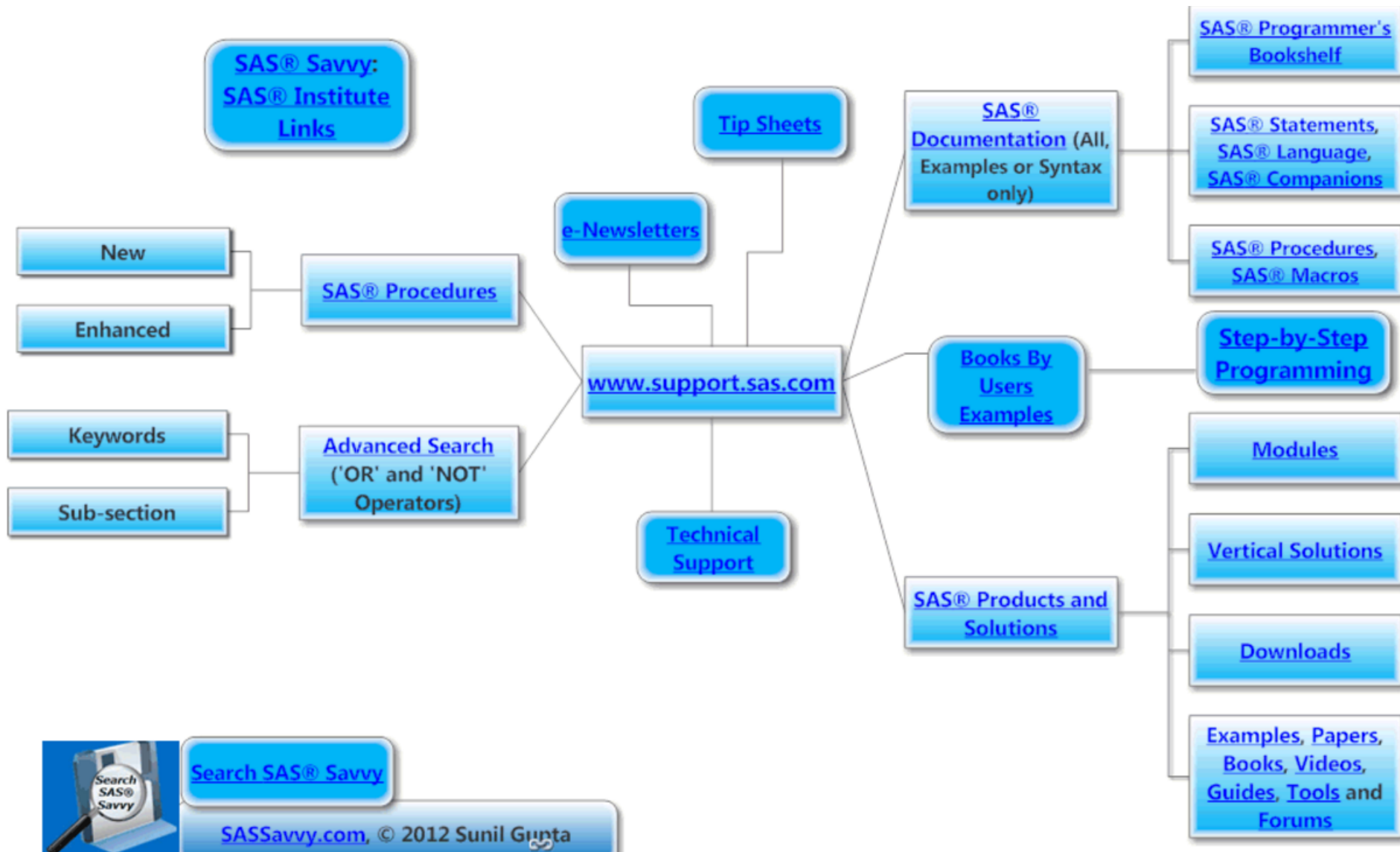


Share

2

How many times have you had a need to convert between variable types such as converting character to numeric or numeric to character? For example, what if you have a character variable with numeric values but you need to perform some calculations? Or, if you have a numeric variable but you need to concatenate it to a character variable? If you are like most SAS programmers, you need to use PUT() and INPUT() at least once to complete these tasks.

Take Advantage of SAS Institute Resources



Three Ways to Leverage Online Mentoring

Job aid guides, lists of best practices and collaborating with peers are key to using technology to meet daily workplace challenges.

Sunil Gupta | February 7, 2014 | 5 Comments

SASSavvy.com's recent publications

Related Topics: Mentoring, Coaching, Technology, Learning and Development

KEYWORDS Coaching / Collaboration / Communication Technologies / Development / HR Technology / Mentoring



Reprints

Share:



1



Tweet

0



Share

1



G+1

1



SAVVY ONLINE SEARCHES SAVE TIME

Savvy online searches enable us to, for example, save links to favorite sites, make journals of summaries and tips, and document answers to frequently asked questions.

Article Author: By Sunil Gupta, Founder, SASSavvy.com

These days, just about everyone conducts online searches to find quick answers to most questions. But the real question is: How to do smarter online searches that can save time and learn from previous searches?

Savvy online searches enable us to, for example, save links to favorite sites, make journals of summaries and tips, and document answers to frequently asked questions. For the last four years, I have used my online platform, SASSavvy.com, to mentor university and remote students as they learn to make savvy online searches. I utilized technology to help standardize and automate the process of building an online knowledge base.

5) Best Practices: Return of the SAS Jedi Master

**SAS Procedures
(Proc SQL, etc.)**

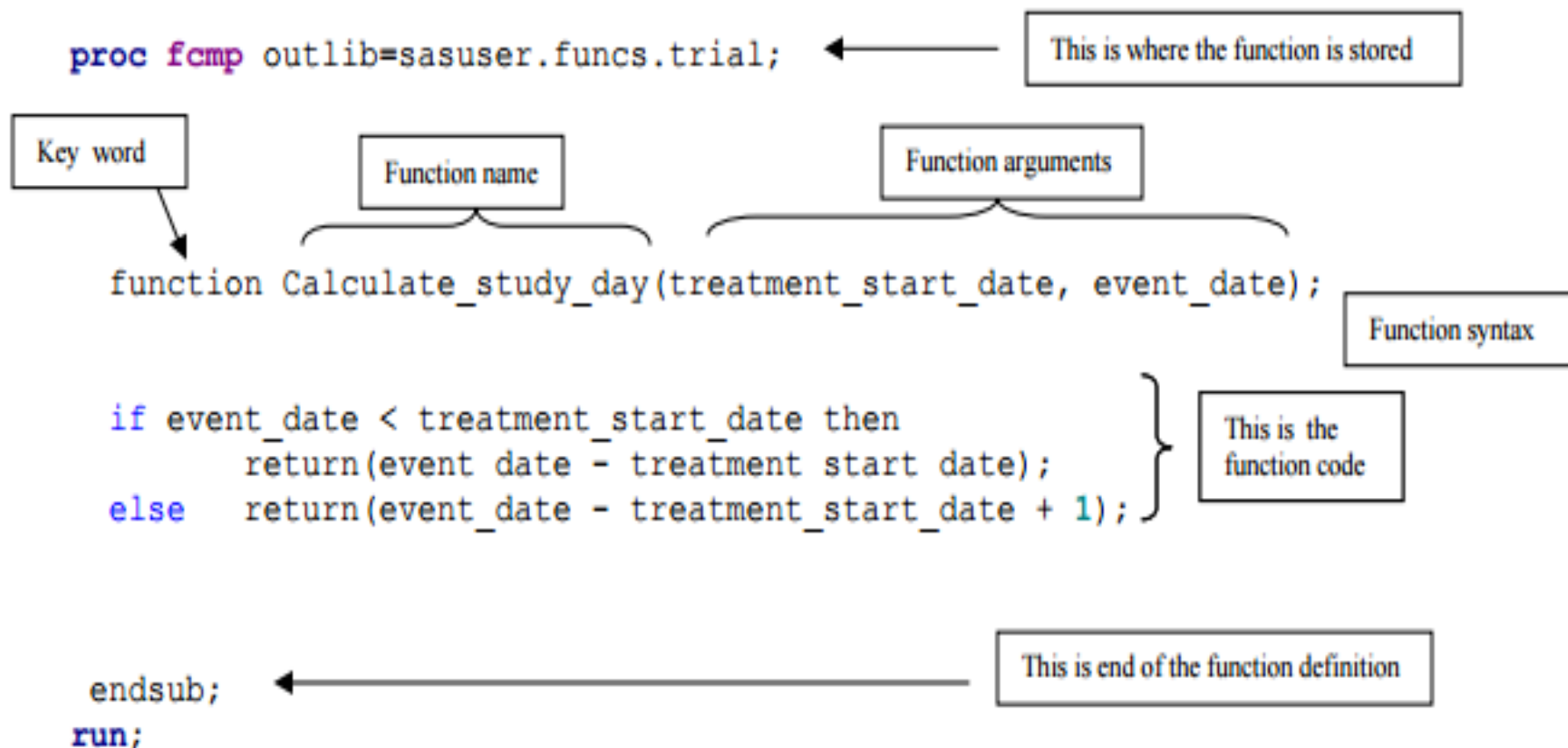
**Data Step and SAS
Macro Programming**

Advanced SAS Skills

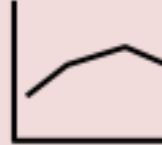
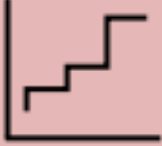
**Advanced Tools
(SAS Index, System
Options, etc.)**

**Advanced Technology
(SAS 9.3)**

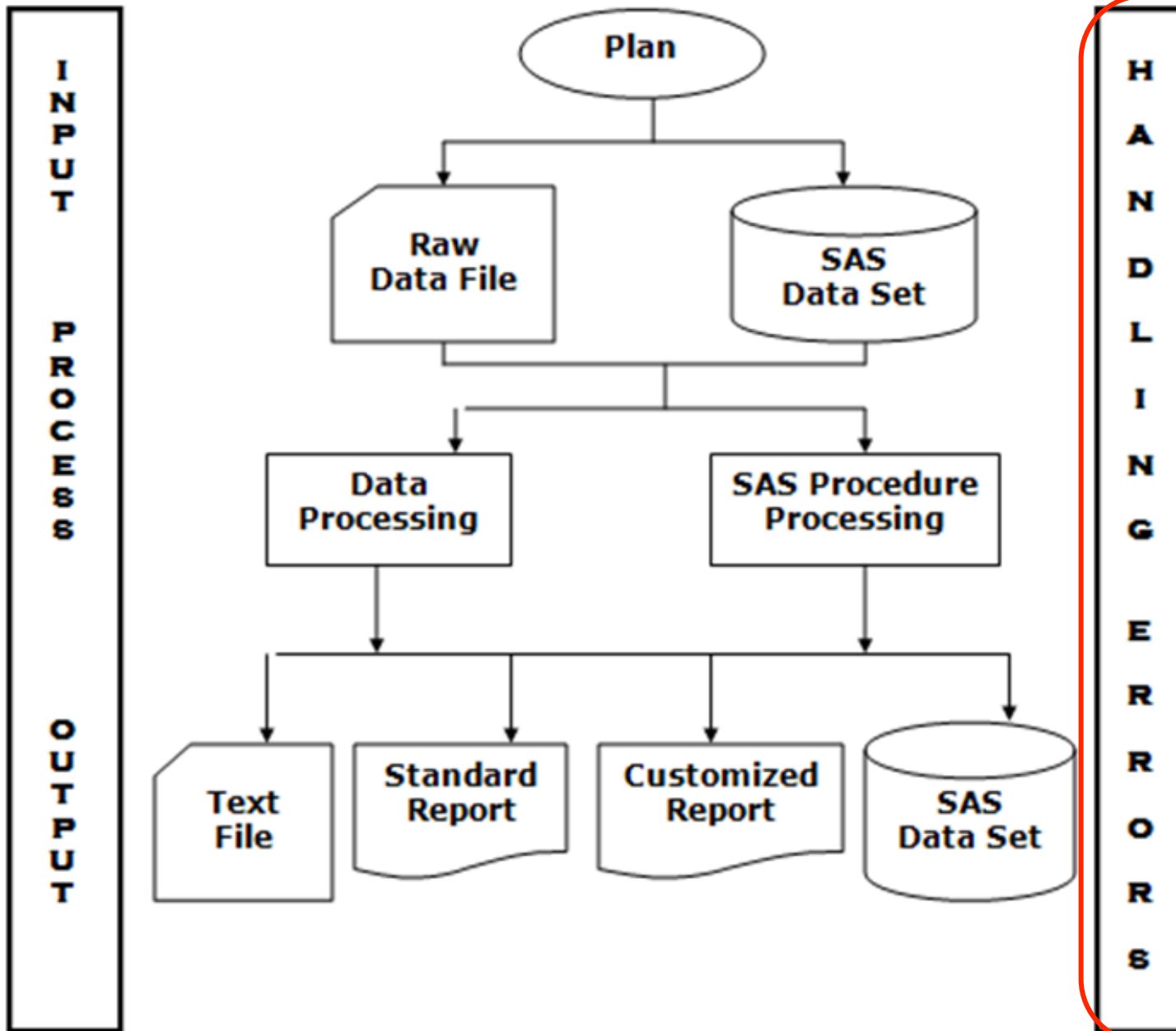
Create My SAS Function – Function Compiler (FCMP) Procedure



ODS Statistical Graphics - PROC SGPLOT

	SYNTAX	SELECTED OPTIONS	
SCATTER 	SCATTER X=var Y=var/options;	DATALABEL=var	Displays a label for each DATA point
SERIES 	SERIES X=var Y=var/options;	BREAK CURVELABEL	Creates a break in the line for each missing value Labels the series curve using the Y variable label
STEP 	STEP X=var Y=var/options;	BREAK CURVELABEL	Creates a break in the line for each missing value Labels the step curve using the Y variable label
NEEDLE 	NEEDLE X=var Y=var/options;	BASELINE=val	Specifies a numeric value on the Y axis for the baseline
VECTOR¹ 	VECTOR X=var Y=var/options;	XORIGIN=val YORIGIN=val	Specifies X coordinate for origin, either numeric value or numeric variable. Specifies Y coordinate for origin, either numeric value or numeric variable.

Anatomy of a SAS Program



SAS
Programmers
are often less
productive
because of
their limited
expertise in
SAS
debugging
skills.

Just like cats,
SAS programs
do have nine
lives.

Anatomy of a Table Shell: Demographics

Summary Table – Four Construction Factors

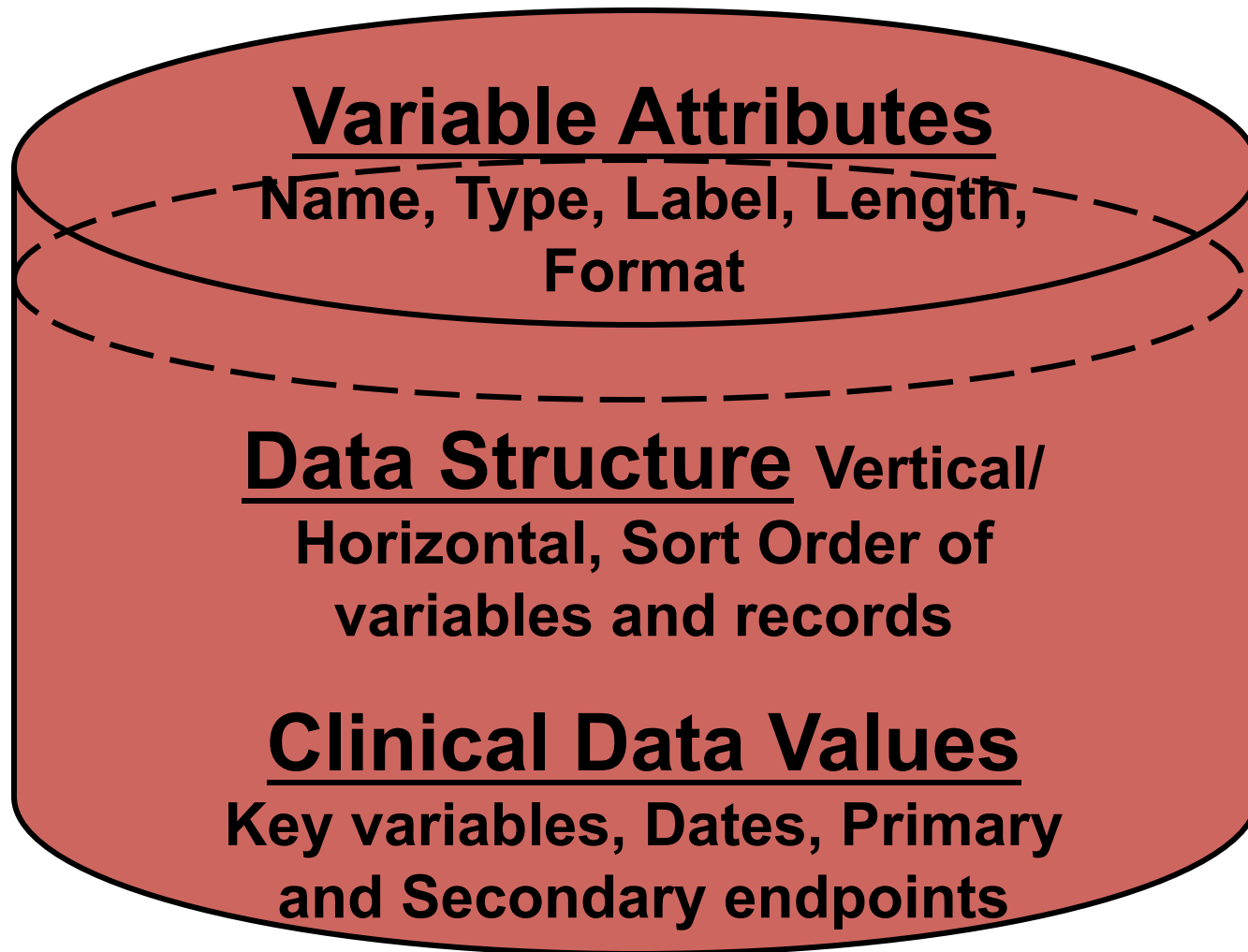
Pharma R&D	DDMMYY:HH:MM:SS
Protocol 001	Page X of Y
Table 1.1 Demographics	
Population: Efficacy Set	

Variable		Active (N=XX)	Placebo (N=XX)
Sex	Male	X (XX.X %)	X (XX.X %)
	Female	X (XX.X %)	X (XX.X %)
Race	Caucasian	X (XX.X %)	X (XX.X %)
	Other	X (XX.X %)	X (XX.X %)
Age	N	XX	XX
	Mean	XX.X	XX.X
	Std	XX.X	XX.X
	Min	XX.X	XX.X
	Max	XX.X	XX.X

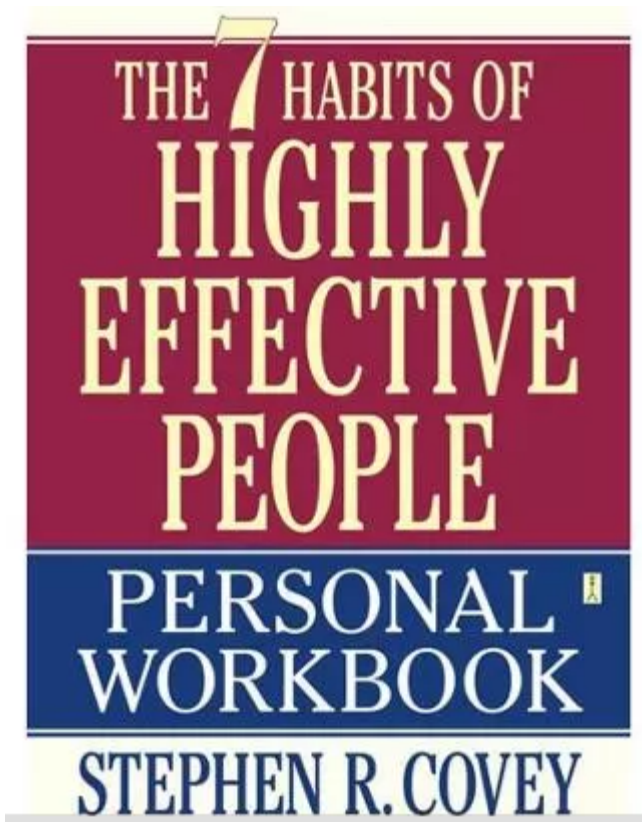
Anatomy of a Table Shell: Suggestions for Macro Program – Four Construction Factors

- ❶ **Which population or subset?** Ex. Efficacy, Safety. Create a macro parameter to specify population. (required)
- ❷ **Which additional subset or By variable if any?** Ex. Age Group: 10 to 40 years. Create a macro parameter to specify additional subset. (optional)
- ❸ **Which treatment groups or total column and in what column order?** Ex. Active, Placebo. Create a macro parameter to specify treatment groups and in what column order. (required)
- ❹ **Which analysis variables and in what order?** Ex. Sex, Race, Age. Create a system to build the results and p-values for each analysis variables and in the correct order. (required)

Three Components when validating Data Sets



Seven Habits of Highly Effective People



SAS Programmers should remember to also apply these Seven Habits:

- ✓ **HABIT 1: BE PROACTIVE**
 - ✓ Plan and schedule SAS programming and resources
- ✓ **HABIT 2: BEGIN WITH THE END IN MIND**
 - ✓ Create complete dataset/table specifications
- ✓ **HABIT 3: PUT FIRST THINGS FIRST**
 - ✓ Apply onboarding training and high expectations
- ✓ **HABIT 4: THINK WIN-WIN**
 - ✓ Create an environment of trust and teamwork
- ✓ **HABIT 5: SEEK FIRST TO UNDERSTAND, THEN TO BE UNDERSTOOD**
 - ✓ Listen, listen and listen to your clients
- ✓ **HABIT 6: SYNERGIZE**
 - ✓ Discuss with client and team ways to improve service
- ✓ **HABIT 7: SHARPEN THE SAW**
 - ✓ Allocate time to review issues or lessons learned

Effectiveness is doing the right things. Follow and update SOPs with standard processes.

Efficiency is concerned with doing things right. Follow all steps and don't take shortcuts.

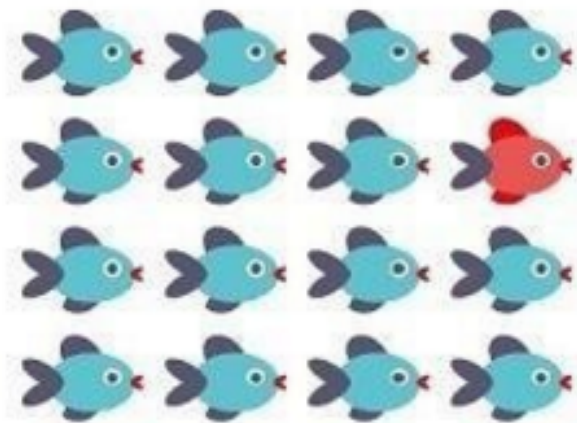


**SAS Programmers need to be *both*
Effective and Efficient!**

SAS Career: Management Track

Differences between

BOSS ← Vs. → **LEADER**



- **Drives** employees
- Depends on **authority**
- Inspires **fear**
- Says **"I"**
- Places **blame** for the breakdown
- **Knows** how it's done
- **Uses** people
- **Takes** credit
- **Commands**
- Says **"Go!"**

- **Coaches** employees
- Depends on **goodwill**
- Generates **enthusiasm**
- Says **"we"**
- **Fixes** the breakdown
- **Shows** how it's done
- **Develops** people
- **Gives** credit
- **Asks**
- Says **"Let's go!"**

Five Habits of Highly Effective SAS Programmers

Sunil Gupta

Associate Director,
Statistical Programming
Cytel

SASSavvy.com

