

Using Proc SQL select into for List Processing

Ronald J. Fehd, Centers for Disease Control and
Prevention, Atlanta, GA, USA

SAS User Group Conferences, fall, 2006

Concept: List Processing

- ▶ *Dynamic Programming*
- ▶ Less work
- ▶ More output
- ▶ Programming with what is there

Concept: List Processing

- ▶ *Dynamic Programming*
- ▶ Less work
- ▶ More output
- ▶ Programming with what is there

Concept: List Processing

- ▶ *Dynamic Programming*
- ▶ Less work
- ▶ More output
- ▶ Programming with what is there

List Processing: How do I process **every** ... ?

1. column in a dataset
2. file in a folder
3. member in a libref
4. *item in a list*

List Processing: How do I process **every** ... ?

1. column in a dataset
2. file in a folder
3. member in a libref
4. *item in a list*

Audience Prerequisites

- ▶ programming: 3 or more years
- ▶ data structures
- ▶ macro language
- ▶ procedures: Contents, Print

Outline

Theory

SQL Basics

- Syntax: SQL

- Syntax: select

- Processes

List Processing

- Making Macro Variables

- Writing Text

- Using SQL Dictionaries

Making Macro Arrays

A Program consists of:

Data Structure

- ▶ attributes
- ▶ organization

Algorithm

- ▶ input
- ▶ process
- ▶ output

see pg. 3

SQL Syntax

```
proc sql;
    create          Libref.Data;
    describe       Libref.Data;
    select
                  from Libref.Data
                  ;
quit             ;
```

see pg. 3

SQL Syntax

```
proc sql;  
    create           Libref.Data;  
    describe       Libref.Data;  
    select  
                from   Libref.Data  
                ;  
quit           ;
```

see pg. 3

SQL Syntax

<code>select</code>		<code>variable(s)</code>
<code>into</code>		<code>:macro variable</code>
<code>from</code>		<code>Libref.Data</code>
	<code>where</code>	<code>condition</code>
<code>group by</code>		<code>variable(s)</code>
	<code>having</code>	<code>condition</code>
<code>order by</code>		<code>variable(s)</code>

see pg. 3

SQL Syntax

<code>select</code>		<code>variable(s)</code>
<code>into</code>		<code>:macro variable</code>
<code>from</code>		<code>Libref.Data</code>
	<code>where</code>	<code>condition</code>
<code>group by</code>		<code>variable(s)</code>
	<code>having</code>	<code>condition</code>
<code>order by</code>		<code>variable(s)</code>

see pg. 3

Basic Processes

- ▶ Listing Data Structure
- ▶ Listing Data
- ▶ Subsetting Data
- ▶ Creating Data
- ▶ Making Unique

see pg. 4

Batch Processing

```
Notepad MyProgram.sas  
SAS      MyProgram  
Notepad MyProgram.log  
Notepad MyProgram.lst
```

see pg. 5

Basic Processing Exercises: 1, 2

- ▶ ExBase1-1: review data structure: SAShelp.Class
- ▶ ExBase1-2: use where clause
- ▶ ExBase2-1: review data structure: Dictionary.Columns
- ▶ ExBase2-2: view any data structure

see pg. 6, 7

Summary

select: use comma as delimiter

wrong: `select Column1 Column2`

right: `select Column1, Column2`

where : use UPPER CASE

wrong: `where Libname eq 'sashelp'`

right: `where Libname eq 'SASHELP'`

see pg. 8

Summary

select: use comma as delimiter

wrong: select Column1 Column2

right: select Column1, Column2

where : use UPPER CASE

wrong: where Libname eq 'sashelp'

right: where Libname eq 'SASHELP'

see pg. 8

Summary

select: use comma as delimiter

wrong: select Column1 Column2

right: select Column1, Column2

where : use UPPER CASE

wrong: where Libname eq 'sashelp'

right: where Libname eq 'SASHELP'

see pg. 8

Creating a Macro Variable

```
proc sql; select Variable-Value(s)
              into   :MacroVariable
              from   Libref.Data
              where  (Condition);
quit;
```

see pg. 8

Basic Processing Exercises: 3

- ▶ ExBase3-1: review data structure: Dictionary.Tables
- ▶ ExBase3-2: making 2 macro variables

see pg. 8, 9

Writing Text into a Macro Variable

```
proc sql; select  '* prefix text'  
                Variable-Value(s)  
                '* suffix text'  
                into  :MacroVariable  separated by ' '  
                from  Libref.Data  
                where  (Condition);  
quit;
```

see pg. 10

Basic Processing Exercises: 4

- ▶ ExBase4-1: review data structure: Dictionary.Tables
- ▶ ExBase4-2: select text into macro variable

see pg. 11, 12

Summary

- ▶ input: examine data structure
- ▶ process: check text
- ▶ output: log, list, ODS

see pg. 13

List of Dictionaries

- ▶ Columns
- ▶ Dictionaries, v9
- ▶ Macros
- ▶ Options
- ▶ Tables

see pg. 13

Registered Trademark

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.

Contact Information

Author: Ronald Fehd

SAS-L's macro maven

Affiliation: Centers for Disease Control
Help Desk, level 3: SAS

E-mail: **RJF2@cdc.gov**

Presentation pdf: \LaTeX Beamer class