REPORT WRITING WITH PROC PRINT AND PROC QPRINT

James A. Le Goullon, Unisys Corporation
Debra Martinson, Unisys Corporation

ABSTRACT

The PRINT and QPRINT procedures are used to print data from a SAS dataset. The reports produced by these procedures can be as simple or complex as needed, depending on the options used.

INTRODUCTION

PROC PRINT can produce anything from a very simple line-by-line listing of observations to a "customized" report format with subgroups, subtotals, labels and more. QPRINT is an alternative to the PRINT procedure, and although it does not offer all of the features of PROC PRINT, it may provide the programmer with greater control over the report layout.

Both the PRINT and QPRINT procedures can only print data stored in a SAS dataset. For information on creating a SAS dataset read Chapter 3 of the SAS User's Guide: Basics. Introduction to the Data Step. This paper features PROC PRINT and concludes with an outline of PROC QPRINT and a simple comparison of the two.

SPECIFICATIONS

PROC PRINT can be used alone, with PROC PRINT options, or with other SAS statements to produce the desired results:

We will see PROC PRINT produce a default report in its simplest form. We will then add options and statements to enhance the report. Remember that a SAS procedure can only be executed on SAS datasets.

Let's begin by writing a data step to create a SAS dataset and then use PROC PRINT to view the results.

A SIMPLE PROGRAM TO CREATE A SAS DATASET:

```
DATA SALESPH;
  INPUT @01 DATEPAID MMDDY16.
  @08 NAME $20.
  @09 SOCSEC $8.
  @07 COMEQUIP $2.
  @07 CONSERV $2.
  @08 SALEZONE $4.;
  TOTALPAY = COMEQUIP + CONSERV;
  CARDS;
  010486 MARTINSON, DEBRA 889678544 1250.00 0510.00 A
  010886 LEGOULLON, JIM 557599466 0489.00 0063.00 C
  112086 D'NEAL, GARY 892792712 0850.00 0110.00 C
  112086 HENRY, STEPHANIE 755454354 2050.00 0650.00 C
  112086 HOWARD, CASSIE 918167169 0975.00 0430.00 B
  212086 JAMES, BRANDON 786956495 2050.00 0600.00 C
  212086 GDALANO, CRAIG 891673478 0975.00 0430.00 B
  212086 COLEMAN, TAMIA 981281080 1150.00 0635.90 B
  100386 CAMELLE, DALE 000000000 0150.00 0030.00 C
  100286 MORALES, MARY 07259265 3050.00 0210.00 A
  100286 L'TALIEU, JEFF 827827626 0750.00 0210.00 A
  100286 TANNER, GARY 827827626 0750.00 0210.00 A
;```

**FIGURE #1**

The program in Figure #1 creates a SAS dataset containing the date the salespeople were paid, their name, their Social Security number, their commission on equipment sold, their commission on service contracts and their sales zone, then both commissions figures are added to compute total gross pay.

This was followed by the simplest form of PROC PRINT...

```
PROC PRINT;
```

... producing a default report as shown in Figure #2:

```
SAS 8:03 THURSDAY, DECEMBER 4, 1986 1

<table>
<thead>
<tr>
<th>OBS</th>
<th>DATEPAID</th>
<th>NAME</th>
<th>SOCSEC</th>
<th>COMEQUIP</th>
<th>CONSERV</th>
<th>SALEZONE</th>
<th>TOTALPAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9500</td>
<td>MARTINSON, DEBRA</td>
<td>899678544</td>
<td>1250</td>
<td>510.00</td>
<td>A</td>
<td>1780.00</td>
</tr>
<tr>
<td>2</td>
<td>9504</td>
<td>LEGOULLON, JIM</td>
<td>557599466</td>
<td>489</td>
<td>83.00</td>
<td>C</td>
<td>552.00</td>
</tr>
<tr>
<td>3</td>
<td>9765</td>
<td>D'NEAL, GARY</td>
<td>892792712</td>
<td>850</td>
<td>110.00</td>
<td>C</td>
<td>960.00</td>
</tr>
<tr>
<td>4</td>
<td>9035</td>
<td>HENRY, STEPHANIE</td>
<td>755454354</td>
<td>2050</td>
<td>600.00</td>
<td>C</td>
<td>2650.00</td>
</tr>
<tr>
<td>5</td>
<td>9035</td>
<td>HOWARD, CASSIE</td>
<td>918167169</td>
<td>975</td>
<td>430.00</td>
<td>B</td>
<td>1405.00</td>
</tr>
<tr>
<td>6</td>
<td>9635</td>
<td>JAMES, BRANDON</td>
<td>786956495</td>
<td>450</td>
<td>647.44</td>
<td>B</td>
<td>277.53</td>
</tr>
<tr>
<td>7</td>
<td>9635</td>
<td>GDALANO, CRAIG</td>
<td>891673478</td>
<td>250</td>
<td>27.53</td>
<td>B</td>
<td>277.53</td>
</tr>
<tr>
<td>8</td>
<td>9636</td>
<td>COLEMAN, TAMIA</td>
<td>981281080</td>
<td>1150</td>
<td>630.90</td>
<td>B</td>
<td>1785.90</td>
</tr>
<tr>
<td>9</td>
<td>9772</td>
<td>CAMELLE, DALE</td>
<td>000000000</td>
<td>0</td>
<td>30.00</td>
<td>C</td>
<td>180.00</td>
</tr>
<tr>
<td>10</td>
<td>9791</td>
<td>MORALES, MARY</td>
<td>72505285</td>
<td>2050</td>
<td>315.16</td>
<td>A</td>
<td>3365.16</td>
</tr>
<tr>
<td>11</td>
<td>9792</td>
<td>L'TALIEU, JEFF</td>
<td>827827626</td>
<td>750</td>
<td>210.00</td>
<td>A</td>
<td>960.00</td>
</tr>
<tr>
<td>12</td>
<td>9790</td>
<td>TANNER, GARY</td>
<td>834378982</td>
<td>175</td>
<td>18.23</td>
<td>A</td>
<td>190.23</td>
</tr>
</tbody>
</table>
```

**FIGURE #2**

110
THE PROC PRINT STATEMENT USED ALONE

Used alone, without options and other statements, the PRINT procedure defaults are as follows:

1. Centers the title of "SAS" on the second line of the report.
2. Prints an "OBS" column as the first column of the report.
3. Variables are printed in the order in which they appear in the SAS data set.
4. Variable names are used as column headings.
5. Page breaks occur when line space is exhausted on the page.
6. Data from the last SAS data set created in the program is used.
7. Adjusts the column spacing from page to page depending on varying variable data lengths (pages are not uniform).

OPTIONS OF THE PROC PRINT STATEMENT

Now let's add some options ...

PROC PRINT options;

- DATA = sasdataset
  - The SAS dataset to be printed.
- NOOBS
  - Suppress printing "OBS" column.
- N
  - Print the number of observations at the bottom of the page, or the bottom of each subgroup.
- UNIFORM (or U)
  - Make the output "uniform" throughout the report as opposed to centered, page by page.
- DOUBLE (or D)
  - Create double spaced output.
- ROUND
  - Round numeric values to two decimal places (unless the number of places is specified in a FORMAT statement).
- LABEL
  - Specify that labels are to be used as column headings. The program must contain a LABEL statement.
- SPLIT = 'splitcharacter'
  - Specify the split character used in your LABEL statement. NOTE: the SPLIT = option implies that labels will be used. The LABEL option is not required in the PROC PRINT statement.

PROC PRINT STATEMENT WITH SOME OPTIONS:

PROC PRINT DOUBLE ROUND N NOOBS;

This produced the report in figure #3:

```
   DATEPAID   NAME         SOCSEC  COMEQUIP  CONSERV  SALEZONE  TOTALPAY
   9500      MARTINSON, DEBRA   880978544              510.00      A        1760.00
   9504      LEOULLON, JIM     5537998466             489       65.00      C        552.00
   9795      O'NEAL, GARY      892792712               850      110.00      C         960.00
   9835      BERRY, STEPHANIE  755484354               2050     600.00      C        2650.00
   9835      HOWARD, CASSIE    918187169               975      430.00      B        1405.00
   9835      JAMES, BRANDON    768956495               450      647.44      B        1097.44
   9838      SOALANO, CRAIG    891673478               250      27.53      B        277.53
   9836      COLEMAN, TANYA    981281080               1150     635.90      B        1785.90
   9772      CAMPBELL, DAVE    0                      150      30.00      C        180.00
   9791      MORALES, MARY     725982685               3050     315.16      A        3365.16
   9792      L'TALIEN, JEFF     827827626               750      210.00      A        960.00
   9790      TANNER, GARY       837437892               175      15.23      A        190.23

   N = 12

FIGURE #3
```
### Statements Used with PROC PRINT

PROC PRINT options;
- VAR variables;
- ID variables;
- BY variables;
- PAGEBY byvariable;
- SUM variables;
- SUMBY byvariable;

- VAR Statement
  - Specifies which variables will be printed and their order across the page.

- ID Statement
  - Overlays the "OBS" column with a specified variable.

- BY Statement
  - Puts the output into subgroups. The SAS dataset must be sorted (with PROC SORT) by the same BY variable(s).

- PAGEBY Statement
  - Creates a page break each time the value of the BY variable changes. PROC PRINT must include a BY statement.

- SUM Statement
  - Specifies that certain numeric columns be summed, and the sum values printed at the bottom of the column.

- SUMBY Statement
  - Sums the columns by the BY variable (the sum value is printed each time the value of the BY variable changes). Again, SUMBY will only work if accompanied by a BY statement in the PROC PRINT.

### Using the SAS Statements

Start with the program example from Figure #1. Followed by a PROC SORT statement sorting the SAS dataset by SALEZONE. (This prepares the PROC PRINT for the BY statement that will create subgroups):

```sas
PROC SORT;
  BY SALEZONE;
```

Followed by a PROC PRINT statement using the ID, VAR, BY, and SUMBY statements:

```sas
PROC PRINT DOUBLE ROUND N;
  ID NAME;
  VAR SOCSEC DATEPAID TOTALPAY;
  BY SALEZONE;
  SUMBY SALEZONE;
```

To produce the report in Figure #4.

### Figure #4

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCSEC</th>
<th>DATEPAID</th>
<th>TOTALPAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>MARTINSON, DEBRA</td>
<td>889478544</td>
<td>0500</td>
<td>1760.00</td>
</tr>
<tr>
<td>MORALES, MARY</td>
<td>73692385</td>
<td>0781</td>
<td>3355.16</td>
</tr>
<tr>
<td>L'ALLIEN, JEFF</td>
<td>821073829</td>
<td>0702</td>
<td>960.00</td>
</tr>
<tr>
<td>TANNER, GARY</td>
<td>837437892</td>
<td>0780</td>
<td>192.22</td>
</tr>
<tr>
<td>SALEZONE</td>
<td></td>
<td></td>
<td>6275.29</td>
</tr>
</tbody>
</table>

N= 4

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCSEC</th>
<th>DATEPAID</th>
<th>TOTALPAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOWARD, CASSIE</td>
<td>918107160</td>
<td>0835</td>
<td>1405.00</td>
</tr>
<tr>
<td>JAMES, BRANDON</td>
<td>789059895</td>
<td>0835</td>
<td>1097.44</td>
</tr>
<tr>
<td>SMALL, CRASH</td>
<td>901679870</td>
<td>0839</td>
<td>377.00</td>
</tr>
<tr>
<td>CRISTIAN, TANYA</td>
<td>981231080</td>
<td>0836</td>
<td>1795.50</td>
</tr>
<tr>
<td>SALEZONE</td>
<td></td>
<td></td>
<td>4590.77</td>
</tr>
</tbody>
</table>

N= 4

<table>
<thead>
<tr>
<th>NAME</th>
<th>SOCSEC</th>
<th>DATEPAID</th>
<th>TOTALPAY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEGGILLON, JIM</td>
<td>857509488</td>
<td>0904</td>
<td>352.00</td>
</tr>
<tr>
<td>O'NEAL, GARY</td>
<td>857795712</td>
<td>0979</td>
<td>680.00</td>
</tr>
<tr>
<td>BEYER, STEPHANIE</td>
<td>796454354</td>
<td>0835</td>
<td>2650.00</td>
</tr>
<tr>
<td>CAMPBELL, DALE</td>
<td>0</td>
<td>0772</td>
<td>160.00</td>
</tr>
<tr>
<td>SALEZONE</td>
<td></td>
<td></td>
<td>4342.00</td>
</tr>
</tbody>
</table>

N= 4

### Figure #4

SAS
ENHANCING THE "LOOK" OF YOUR OUTPUT

THE TITLE STATEMENT

The TITLE statement is used to specify up to 10 lines of titles for most SAS output, including PROC PRINT and QPRINT.

The TITLE statement, to be used with your PROC PRINT, must appear somewhere after a previous RUN statement, if one exists, and before the next RUN, DATA, or PROC statement. Text following the TITLE statement must be enclosed in either single or double quotes, double quotes if macro references are used.

PROGRAM EXAMPLE OF THE TITLE STATEMENT:

```
TITLE1 "A B C COMPANY";
TITLE3 "COMMISSIONS PAID" "BY SALES ZONE";
```

An example of the new report with titles is shown in Figure #5.

THE LABEL STATEMENT

The LABEL statement is used to create column labels. Without the LABEL statement, the PRINT procedure uses the variable names. Text used as labels can be anything, but must be enclosed in quotes.

You may use split characters to cause the label to split to the next line with a limit of two splits (or three line labels). You must then use the SPLIT= option of the PROC PRINT statement and specify what is to be used as the split character.

EXAMPLE OF A LABEL STATEMENT:

```
LABEL
   PRICE = "SALES*PRICE* " ;
```

TO PRODUCE:

```
SALES
PRICE
```

Notice that the words SALES and PRICE are lined up over the " ".

Also take note, every character before and after the SPLIT character is printed, including blanks. If you need to line labels up then you would use blanks.

PROGRAM EXAMPLE OF THE LABEL STATEMENT:

```
LABEL
   DATEPAID = "PAY*DAY*" ;
   NAME = "SALERlFFERSON*" ;
   SOCSEC = "SOCIAL*SECURITY*" ;
   COMMISSION = "COMMISSION*SOLD*" ;
   TOTALPAY = "TOTAL PAY*" ;
   SALESZOE = "SALES ZONE*" ;
```

Notice the two spaces after the " and before PAY and after the * and before DAY. These are used to center PAY and DAY over the eight dashes (-).

An example of the new report with these labels is shown in Figure #6.

3. THE FORMAT STATEMENT

With a FORMAT statement, you format your data for output using SAS formats already available in the SAS library or by creating your own. For example, formats statements are used to print numeric figures with commas or dollar signs. It is also necessary to use a format to print SAS dates in a readable form.

In our example, we will be using the FORMAT statement to enhance the output of the DATEPAID, SOCSEC, and TOTALPAY variables.

AN EXAMPLE OF FORMAT STATEMENTS:

```
FORMAT TOTALPAY DOLLAR10.2;
FORMAT SOCSEC SSN ;
FORMAT DATEPAID MMDDYY8. ;
```

AND OF COMBINING THE THREE INTO ONE FORMAT STATEMENT:

```
FORMAT TOTALPAY DOLLAR10.2
   SOCSEC SSN ;
   DATEPAID MMDDYY8. ;
```

Either way is fine.

The above FORMAT statement was used to format the TOTALPAY, SOCSEC and DATEPAID variables shown on the final report in Figure #7.

Figure #8 shows the finished program starting after the DATA step (Figure #1).
### Figure #5

**ABC Company**

**Monday, December 8, 1986**

#### Commissions Paid

<table>
<thead>
<tr>
<th>Name</th>
<th>Social Security #1</th>
<th>Date Paid</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>MARTINSON, DEBRA</td>
<td>880078644</td>
<td>5/500</td>
<td>1,160.00</td>
</tr>
<tr>
<td>MORALES, MARY</td>
<td>77693986</td>
<td>5/793</td>
<td>339.16</td>
</tr>
<tr>
<td>L'ALIEN, JEFF</td>
<td>875778266</td>
<td>5/793</td>
<td>360.00</td>
</tr>
<tr>
<td>TANNER, GARY</td>
<td>837437902</td>
<td>5/793</td>
<td>190.23</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td></td>
<td>8276.39</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

#### Salezone B

<table>
<thead>
<tr>
<th>Name</th>
<th>Social Security #1</th>
<th>Date Paid</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOWARD, CASSIE</td>
<td>918187106</td>
<td>5/835</td>
<td>1405.90</td>
</tr>
<tr>
<td>JAMES, BRANON</td>
<td>780856495</td>
<td>5/835</td>
<td>1097.44</td>
</tr>
<tr>
<td>GOLLAND, CRAIG</td>
<td>861874769</td>
<td>5/835</td>
<td>277.83</td>
</tr>
<tr>
<td>COLEMAN, TANYA</td>
<td>901281060</td>
<td>5/835</td>
<td>1788.90</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td></td>
<td>4965.87</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

#### Salezone C

<table>
<thead>
<tr>
<th>Name</th>
<th>Social Security #1</th>
<th>Date Paid</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEDDULLON, JIM</td>
<td>870078544</td>
<td>5/835</td>
<td>583.00</td>
</tr>
<tr>
<td>O'NEAL, GARY</td>
<td>918283719</td>
<td>5/835</td>
<td>660.00</td>
</tr>
<tr>
<td>BERRY, STEPHANIE</td>
<td>780456296</td>
<td>5/835</td>
<td>2600.00</td>
</tr>
<tr>
<td>CAMPBELL, DALE</td>
<td>0081281060</td>
<td>5/872</td>
<td>180.00</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td></td>
<td>4342.00</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

**Figure #6**

**ABC Company**

**Monday, December 8, 1986**

#### Commission Paid

<table>
<thead>
<tr>
<th>Salesperson's Name</th>
<th>Social Security #</th>
<th>Sale Zone #</th>
<th>Pay</th>
<th>Day</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>MARTINSON, DEBRA</td>
<td>880078644</td>
<td>A</td>
<td>5500</td>
<td>5/500</td>
<td>1,160.00</td>
</tr>
<tr>
<td>MORALES, MARY</td>
<td>77693986</td>
<td>A</td>
<td>5793</td>
<td>5/793</td>
<td>339.16</td>
</tr>
<tr>
<td>L'ALIEN, JEFF</td>
<td>875778266</td>
<td>A</td>
<td>5793</td>
<td>5/793</td>
<td>360.00</td>
</tr>
<tr>
<td>TANNER, GARY</td>
<td>837437902</td>
<td>A</td>
<td>5793</td>
<td>5/793</td>
<td>190.23</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td>A</td>
<td></td>
<td></td>
<td>8276.39</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td>A</td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

#### Salezone B

<table>
<thead>
<tr>
<th>Salesperson's Name</th>
<th>Social Security #</th>
<th>Sale Zone #</th>
<th>Pay</th>
<th>Day</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>HOWARD, CASSIE</td>
<td>918187106</td>
<td>B</td>
<td>835</td>
<td>5/835</td>
<td>1405.90</td>
</tr>
<tr>
<td>JAMES, BRANON</td>
<td>780856495</td>
<td>B</td>
<td>835</td>
<td>5/835</td>
<td>1097.44</td>
</tr>
<tr>
<td>GOLLAND, CRAIG</td>
<td>861874769</td>
<td>B</td>
<td>835</td>
<td>5/835</td>
<td>277.83</td>
</tr>
<tr>
<td>COLEMAN, TANYA</td>
<td>901281060</td>
<td>B</td>
<td>835</td>
<td>5/835</td>
<td>1788.90</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td>B</td>
<td></td>
<td></td>
<td>4965.87</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td>B</td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

#### Salezone C

<table>
<thead>
<tr>
<th>Salesperson's Name</th>
<th>Social Security #</th>
<th>Sale Zone #</th>
<th>Pay</th>
<th>Day</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>LEDDULLON, JIM</td>
<td>870078544</td>
<td>C</td>
<td>835</td>
<td>5/835</td>
<td>583.00</td>
</tr>
<tr>
<td>O'NEAL, GARY</td>
<td>918283719</td>
<td>C</td>
<td>835</td>
<td>5/835</td>
<td>660.00</td>
</tr>
<tr>
<td>BERRY, STEPHANIE</td>
<td>780456296</td>
<td>C</td>
<td>835</td>
<td>5/835</td>
<td>2600.00</td>
</tr>
<tr>
<td>CAMPBELL, DALE</td>
<td>0081281060</td>
<td>C</td>
<td>872</td>
<td>5/872</td>
<td>180.00</td>
</tr>
<tr>
<td><strong>SALEZONE</strong></td>
<td></td>
<td>C</td>
<td></td>
<td></td>
<td>4342.00</td>
</tr>
<tr>
<td><strong>N</strong></td>
<td></td>
<td>C</td>
<td></td>
<td></td>
<td>4</td>
</tr>
</tbody>
</table>

**Figure #6**

Number: 114
<table>
<thead>
<tr>
<th>Salesperson's Name</th>
<th>Social Security #</th>
<th>Day</th>
<th>Pay</th>
<th>Total Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Martinson, Debby</td>
<td>890-87-8644</td>
<td>01/04/88</td>
<td>1,780</td>
<td></td>
</tr>
<tr>
<td>Morales, Mary</td>
<td>007-25-9725</td>
<td>10/22/88</td>
<td>2,205</td>
<td></td>
</tr>
<tr>
<td>L'Italian, Jeff</td>
<td>837-87-4392</td>
<td>10/23/88</td>
<td>980</td>
<td></td>
</tr>
<tr>
<td>Tanner, Gary</td>
<td></td>
<td></td>
<td></td>
<td>1100</td>
</tr>
</tbody>
</table>

- **Sales Zone A**
- **Salesman's Name**
  - Martison, Debby
  - Morales, Mary
  - L'Italian, Jeff
  - Tanner, Gary
- **Social Security**
  - 890-87-8644
  - 007-25-9725
  - 837-87-4392
- **Total Pay**
  - 1,780
  - 2,205
  - 980
  - 1,100

- **Sales Zone B**
- **Salesman's Name**
  - Howard, Cassie
  - James, Brandon
  - Soiland, Craig
  - Coleman, Tanya
- **Social Security**
  - 816-16-7108
  - 769-95-0495
  - 891-57-3478
  - 881-28-1080
- **Total Pay**
  - 1,405
  - 1,097
  - 276
  - 1,788

- **Sales Zone C**
- **Salesman's Name**
  - Legoillon, Jim
  - O'Neal, Gary
  - Berry, Stephanie
  - Cambell, Dale
- **Social Security**
  - 837-88-4686
  - 892-70-7712
  - 795-86-9354
  - 000-00-0000
- **Total Pay**
  - 1,952
  - 380
  - 850
  - 180

**Figure #7**

PROC SORT;
BY SALEZONE;
PROC PRINT N SPLIT="*";
ID NAME;
VAR SOCSEC DATEPAID TOTALPAY;
BY SALEZONE;
SUM TOTALPAY;
SUMBY SALEZONE;
TITLE1 "A B C COMPANY";
TITLE3 "COMMISSIONS PAID" "BY SALES ZONE";
LABEL DATEPAID = "Pay* Day*"
NAME = "Salesperson's Name*"
SOCSEC = "Social Security #*"
COMQQUIP = "Commission*Equip Sold*"
COMSERV = "Commission*Serv Contracts*"
TOTALPAY = "Total Pay*"
SALEZONE = "Sales Zone*"
;
FORMAT TOTALPAY DOLLAR10.2
SOCSEC SSN.
DATEPAID MMDDYY6.;

**Figure #8**

115
THE QPRINT PROCEDURE

The QPRINT procedure is an alternative to PROC PRINT that prints values of observations in a table. QPRINT may offer the SAS programmer greater execution efficiency and more control over the report layout. QPRINT does not, however, offer PAGEBY, SUM, or SUMBY statements as available with PROC PRINT.

Just one feature of QPRINT is the ability to print your data in panels or "newspaper" style, print constant text between columns, and more.

PROC QPRINT SPECIFICATIONS:

The following statements can be used with QPRINT:

```
PROC QPRINT options;
RESET options;
OBS options;
ID variables {/options};
VAR variables {/options};
BY variables;
CONSTANT quoted text {/options};
HEAD options;
FOOT options;
TAIL {name};
```

STATEMENTS USED WITH QPRINT

Below are statements of the QPRINT procedure. This is a very brief "highlight" at most! More information can be obtained through the Institute.

- **RESET**
  - The RESET statement allows OBS, ID, VAR, and HEAD options to be reset. This allows two or more different looking reports on one page.

- **OBS**
  - The OBS statement causes the printing of an OBS column in each panel of the report. This is the default in PROC PRINT.

- **ID Statement**
  - Like the ID statement in PROC PRINT but with extra options. For instance, a header can be specified to span all ID variables.

- **VAR Statement**
  - Again, similar to the ID statement in PROC PRINT but with extra options.

- **CONSTANT Statement**
  - The CONSTANT statement specifies the printing of character constant to be printed in every row of a table.

- **HEAD and TAIL Statements**
  - Can be used together to specify and mark the beginning and ending of a heading to be printed over one or more (or all) columns.

- **FOOT and Tail Statements**
  - Same with the HEAD and TAIL statements only the text is printed as footings to the specified columns.

- **BY Statement**
  - Used to created separate tables for each group as defined by the BY variables. Each new table for each BY variable begins on a new page.

OPTIONS USED WITH QPRINT

Again, this list is not all inclusive, but only highlights some feature options special to QPRINT:

- **ALIGN**
  - With ALIGN, SAS centers the widest panel, and all other panels are indented identically - otherwise each panel is centered individually according to its data values.

- **FOOT**
  - Where you may specify the number of lines to be left between the data and any footings.

- **HEAD**
  - Indicate the number of lines to be left between the data and any column headings.

- **PANEL**
  - Specify the number of blank lines to leave between panels.

- **CHARS**
  - Specifies the number of characters of a variable name or heading to be used. The default is to use all of the characters.

- **HEAD=NAME, LABEL, or BLANK**
  - Indicate whether variable names, labels or nothing is to be printed as column headings.
• INDENT=
  - Can indent data values from the column heading.

• JUSTIFY=LEFT, CENTER, RIGHT or DEFAULT
  - Specifies justification of column headings.

• LINE=
  - Specifies which line a column heading or footing is printed.

• MULTIPLE=
  - Prints the table in a multi-column format - or "newspaper" style.

• NUMBER{=quoted string}
  - Prints the number of observations but differs from the N option of PROC PRINT where an optional quoted string can be used as a label.

• OVERLINE{='c'}
  - Prints a line of hyphens (or another specified character) between the data and the footings.

• ROWS=n, MAX, PAGE, OBS, or WRAP
  - Specify the number of rows to be printed in each block, print all rows in the group or dataset, all that will fit on a page, print as many panels as can be printed on a few pages as possible, or in wraparound fashion.

• VERTICAL/NOVERTICAL
  - Default is NOVERTICAL - specifying the VERTICAL option causes the variable names to print vertically.

• WIDEN/NOWIDEN
  - Default is NOWIDEN - the WIDEN option widens columns as appropriate for enough space to print all of a heading or footing.

• WIDTH=
  - Specify the range of column widths.

• ZONE=
  - Specifies the maximum width of the table.

COMPARISON OF THE PRINT AND QPRINT PROCEDURES

Briefly comparing the two:

1. Only with the PRINT procedure can the PAGEBY, SUM, or SUMBY statements be used.

2. Only with QPRINT can you span headings and footings over groups or columns.

3. QPRINT offers greater control of the overall report layout and narrow tables can be printed in multicolumn (or panel) layout.

4. For a large number of variables, QPRINT requires less memory than PRINT.

5. PRINT may use less computer time if printing less than 100 variables but QPRINT becomes more and more efficient than PRINT as the number of observations increases.

6. QPRINT requires less pages to print although if the data values vary greatly, PRINT may use substantially less pages without the UNIFORM option.

CONCLUSION

In conclusion, both the PRINT and QPRINT procedures offer the SAS programmer a fast and easy way to produce printed output of a SAS dataset. It is very easy to build on the basic default report, enhancing it until you have exactly what you want.

REFERENCES:

The PROC PRINT information covered in this talk, unless otherwise specified, is documented in SAS User's Guide: Basics, Chapter 43, pages 1007 through 1017.


The authors may be reached at Unisys Corporation, San Tomas @ Central Expressway, Santa Clara, CA, 95052.

SAS is a registered trademark of SAS Institute Inc., Cary, NC, USA.