

ODS (Output Delivery System)

Prepared by



International SAS® Training and Consulting
Destiny Corporation

100 Great Meadow Rd, Suite 601

Wethersfield, CT 06109-2379

Phone: (860) 721-1684 1-800-7TRAINING

Fax: (860) 721-9784

Web: www.destinycorp.com Email: info@destinycorp.com
Copyright 2002

This presentation is designed to demonstrate the different possibilities we use when we build web based applications for clients and teach them the web publishing techniques we use with SAS Software. Actual examples are available on our web site.

Basic HTML Publishing

The most basic form is brute force HTML generation using the traditional data step. This tends to be the most flexible, but the most time consuming and requires complete knowledge of how HTML works. FILE and PUT statements are used extensively.

```

Command ***
00001 libname saved 'c:\sas';
00002 data _null_;
00003 set saved.demograp;
00004 file 'c:\datenull.html';
00005 if _n_ = 1 then do;
00006   put '<h1>SAS Data Set Demograp</h1>';
00007   / '<h2>from the Saved library</h2>';
00008   / '<ol>';
00009   and;
00010   put '1. F 0 0 $0';
00011   if x then put '';
00012 run;
00013 x 'c:\progra\intern\iexplora.exe c:\datenull.html';
00014

```

SAS Data Set Demograp

from the Saved library

1. F 0 0 \$0
2. F 0 0 \$0
3. F 0 0 \$0
4. F 0 1 \$13,000
5. F 1 1 \$5,600
6. F 2 1 \$8,000

Data Set Publishing with DS2HTM

SAS offers a macro that allows one to take a SAS data file and present it in an HTML form.

```

Command ***
00001 libname saved 'c:\sas';
00002 data work.demograp;
00003 set saved.demograp;
00004 format salary dollar10.;
00005 run;
00006
00007 title1 'Data Set Table';
00008
00009 odshtml(data =work.demograp,
00010        where =status ^= 'SEP',
00011        var =gender status children salary,
00012        htmlfile=c:\test1.html,
00013        runmode =b);
00014
00015 x 'c:\progra\intern\iexplora.exe c:\test1.html';
00016

```

Data Set Table

GENDER	STATUS	CHILDREN	SALARY
F	S	0	\$0
F	S	0	\$0
F	S	0	\$0
F	S	0	\$13,000
F	M	1	\$5,600

SAS Output Publishing with OUT2HTM

Any procedure's output can be captures and published with the OUT2HTM macro.

```

Command ***
00001 libname saved 'c:\sas';
00002 odshtml(capture=off);
00003
00004 title1 'Proc Report Output';
00005 options nlsdate ymdmmdd;
00006 proc report data=saved.demograp (base ps=4 split=) center nowindows;
00007 column gender status age salary children cars;
00008 define gender / group format= $2. width=8 spacing=2 left "GENDER";
00009 define status / group format= $2. width=8 spacing=2 left "STATUS";
00010 define age / sum format= 3. width=8 spacing=2 right "AGE";
00011 define salary / sum format= best. width=8 spacing=2 right "SALARY";
00012 define children / sum format= best. width=8 spacing=2 right "CHILDREN";
00013 define cars / sum format= best. width=8 spacing=2 right "CARS";
00014 break after gender / ol skip summarize;
00015 run;
00016
00017 title1 'Proc Freq Output';
00018 proc freq data=saved.demograp;
00019 table status*gender;
00020 run;
00021
00022 title1 'Proc Tabulate Output';
00023 proc tabulate data=saved.demograp;
00024 class status gender;
00025 var salary;
00026 table status,gender*salary;
00027 run;
00028
00029 odshtml(capture=off,
00030        runmode=b,
00031        htmlfile=c:\test1.html,
00032        ctest=lib,
00033        test=report);
00034 x 'c:\progra\intern\iexplora.exe c:\test1.html';
00035

```

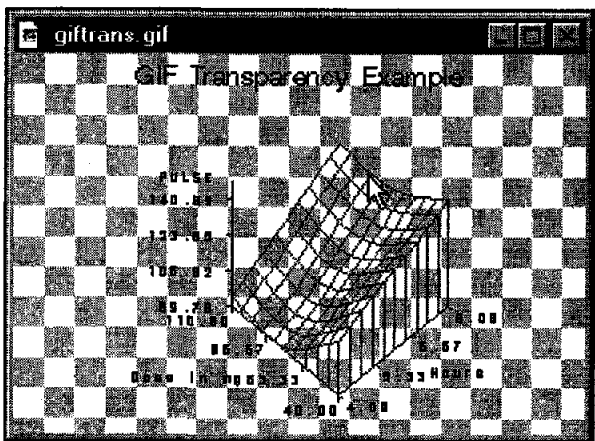
Proc Report Output

GENDER	STATUS	AGE	SALARY	CHILDREN	CARS
F	S	29	8000	2	1
F	S	45	131000	91	20
F	S	37	15000	0	2
F	SEP	23	16000	1	1
F	S	46	10000	3	1
F		35	230600	97	25
F	S	33	20000	5	2
F	S	37	114300	18	9
F	S	37	15000	0	2
F	SEP	25	11000	2	1
F		31	150100	25	14

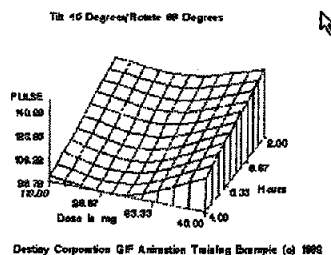

```

00001  * The GTF device driver is available in Releases 6.000 and 6.12 of A&S software.;
00002  * Transparency Example.;
00003
00004  @include gtfdevice "gtfdev";
00005  proc gtfdev gtfdevice=device name;
00006  copy gtf from=device;device name=gtfname;
00007  end gtfname use="0";
00008  quit;
00009
00010 %name ngelz=1280,helz=640,op1=5,rom=3,color=3;
00011  xif dname then op1 must be greater than zero.;
00012  %name %op1
00013  %option helz=keyval%{helz/op1} in valz=keyval%{helz/op1} in
00014  %option%{helz/op1} %op1;
00015  %end;
00016  %end ngelz;
00017
00018 filename out "c:\gtfname.gtf";
00019 %option gtfname1 rom%{helz};
00020 %option gtfname2 gtfname out gtfname%{helz};
00021 %option%{helz} %helz, %helz, %op1, rom%{helz}, %helz;
00022
00023 proc gtfdev saved,contour artwork,weath;
00024  %option gtfdev gtfname%{helz} %helz;
00025  run;
00026  %option "gtf transparency Example";
00027  %option gtfdev weath;
00028  %option label does "Done in ng %gtfname: %name";
00029  %option gtfdev gtfname%{helz} %helz;
00030  run;
00031
00032  x "C:\Program\Comman\N\Color\Photo4\PHOTO4.IMG c:\gtfname.gtf";
00033

```

[illegible]

(Look Mom, see what I can do!)



When ODS was introduced in Version 8 of SAS software, a whole new way of publishing was created.

First, notice the typical style of using ODS.

1. The ODS 'capture style' utility is turned on.
2. The process executes.
3. The ODS utility is turned off.

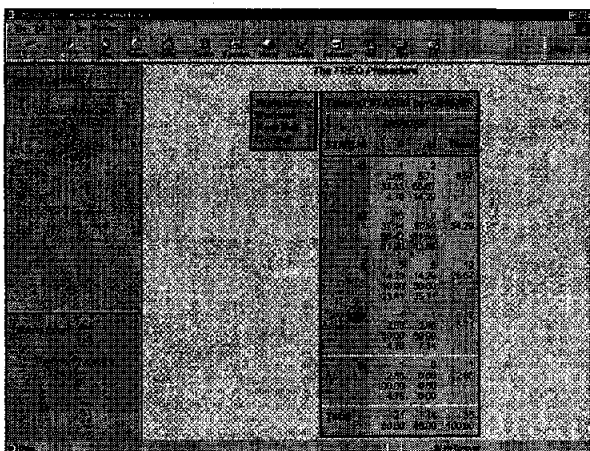
The general ODS syntax can look like the following.

```

Command ==>
00001 ods html body='c:\body.html'
00002           contents='c:\contents.html'
00003           page='c:\page.html'
00004           frame='c:\frame.html';
00005
00006 proc freq data=saved.demogrf;
00007     table status*gender;
00008 run;
00009 ods html close;
00010
00011

```

This syntax routes HTML output anywhere you desire. It creates four files. The FRAME.HTML file pulls them all together.

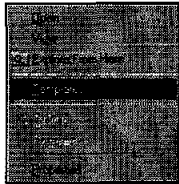


Styles in HTML

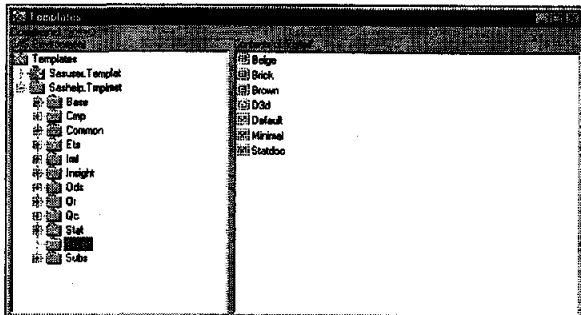
There are several HTML styles that come with SAS. They are designed to allow the user to create an HTML standard of colors, fonts and more across all HTML output. Any of the default ones can be used, or they can be modified to suit your needs. To see the existing styles, go to the Results window and select.



Right click and select Templates.



Select SASHELP.TMPLMST, Styles to see the available styles.



To examine a style, execute Proc Template with the Style name. Default is the default style used.



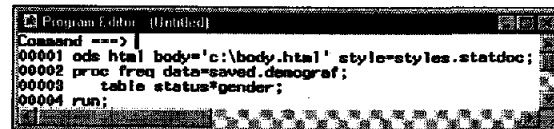
```

Command ===>
254 proc template;
255   source style.default;
define style Styles.Default;
  style fonts
  'TitleFont' = ('Arial, Helvetica, Helv',4,Bold Italic)
  'TitleFont' = ('Arial, Helvetica, Helv',5,Bold Italic)
  'StrongFont' = ('Arial, Helvetica, Helv',4,Bold)
  'EmphasFont' = ('Arial, Helvetica, Helv',3,Italic)
  'FixedEmphasFont' = ('Courier',3,Italic)
  'FixedStrongFont' = ('Courier',4,Bold)
  'FixedLeadInFont' = ('Courier',4)
  'FixedFont' = ('Courier',4)
  'headingEmphasFont' = ('Arial, Helvetica, Helv',4,Bold Italic)
  'headingFont' = ('Arial, Helvetica, Helv',4,Bold)
  'docFont' = ('Arial, Helvetica, Helv',3);
  style color_list /
  'fgB2' = cxx008000
  'fgB1' = cxx004000
  'fgH1' = cxx00FF00
  'bgH1' = cxx808000
  'fgH2' = cxx003000
  'bgH2' = cxx000000
  'fgH3' = cxx000000
  'bgH3' = cxx000000
  'fgH' = cxx000000
  'bgH' = cxx000000
  style colors /
  'headerEmphas' = color_list['fgH2']
  'headerStrong' = color_list['bgH2']
  'headerFgStrong' = color_list['fgH2']
  'headerBgStrong' = color_list['bgH2']

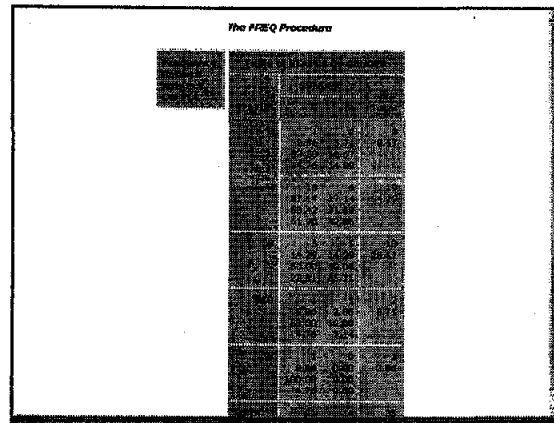
```

Cut and paste the code for modification. Rerun it with a new name and your own style can be created.

To select a style, use the ODS statement



It produces the following HTML. (Looks different than styles.default)



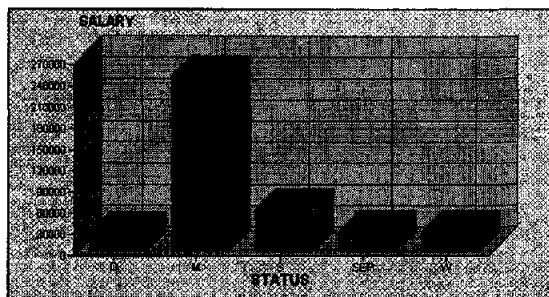
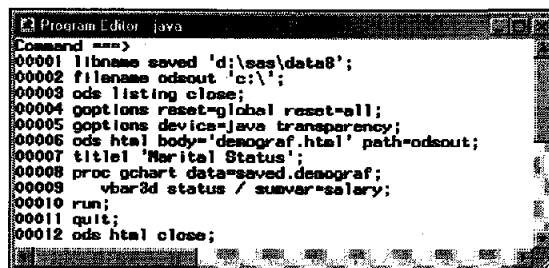
See the online help to learn more about changing styles and what the values mean.

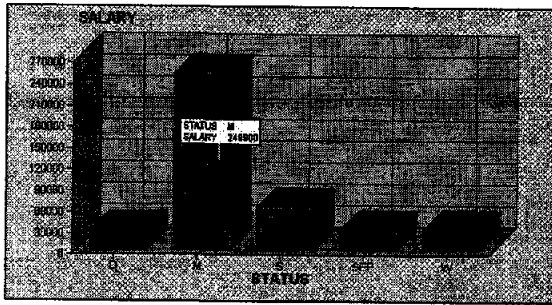
Graphical Support

There are several other drivers available for outputting graphical objects. Consider the following.

Java Support

There is Java support. Consider the following driver.





There are several menus available from the right click of a mouse.



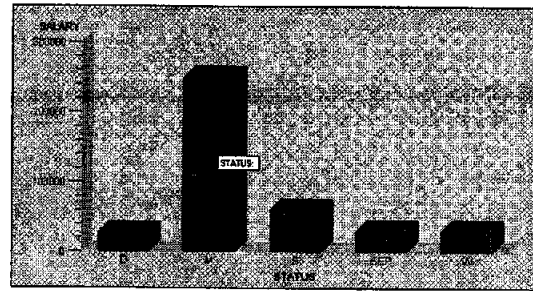
Experiment with each to understand the capabilities.

Active X Support

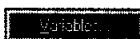
SAS now supports Active X controls. Consider the following code with the Active X Driver.

```

Program Editor - active2
Command ==>
00001 libname saved 'd:\sas\data8';
00002 filename odsout 'c:\';
00003 ods listing close;
00004 goptions reset-global reset-all;
00005 goptions device=activex transparency;
00006 ods html body='demograf.html' path=odsout;
00007 title 'Marital Status';
00008 proc gchart data=saved.demograf;
00009 vbar3d status / sumvar=salary;
00010 run;
00011 quit;
00012 ods html close;
  
```



A right click yields many possibilities.



Consider testing out the different options.

Using Styles in Procedures

Style options are available in SAS. Their syntax is added to procedures when formatting them for output. This is available on the procedure level in the following way. Use the syntax

`S={foreground=blue}`

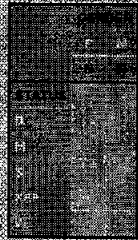
At various locations to specify colors.

Current supported Procedures are Tabulate and Report.

Styles in Proc Tabulate

```
Program Editor - tabcolor
Command ==>
00001 libname saved 'd:\sas\data8';
00002
00003 ods html body="c:\tabcolor.html";
00004 ods listing close;
00005
00006 title 'Tabulate Color Example using styles';
00007 footnote;
00008 proc tabulate data=saved.desograp;
00009 class status gender / s={foreground=blue};
00010 classlev status gender / s={foreground=yellow};
00011 var salary age / s={foreground=white};
00012 keyword all sum / s={foreground=white};
00013 keylabel all="Total";
00014 table status,gender*{s={foreground=orange}};
00015 run;
00016
00017 ods html close;
00018 ods listing;
00019
```

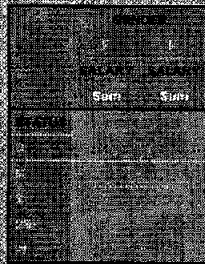
Tabulate Color Example using styles



Additional examples use background and font_style.

```
Program Editor - tabcolor
Command ==>
00001 libname saved 'd:\sas\data8';
00002
00003 ods html body="c:\tabcolor.html";
00004 ods listing close;
00005
00006 title 'Tabulate Color Example using styles';
00007 footnote;
00008 proc tabulate data=saved.desograp;
00009 class status gender / s={background=yellow foreground=blue};
00010 classlev status gender / s={foreground=yellow};
00011 var salary age / s={foreground=white};
00012 keyword all sum / s={foreground=white};
00013 keylabel all="Total";
00014 table status,gender*{s={foreground=orange font_style=italic}}*salary;
00015 run;
00016
00017 ods html close;
00018 ods listing;
00019
```

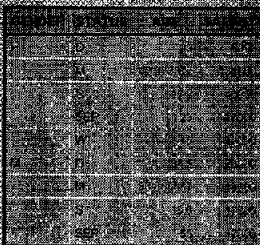
Tabulate Color Example using styles



Styles in Proc Report

```
Program Editor - tabcolor
Command ==>
00001 libname saved 'd:\sas\data8';
00002
00003 ods html body="c:\report.html";
00004 ods listing close;
00005
00006 title 'Report Color Example using styles';
00007 footnote;
00008 proc report data=saved.desograp nowindows;
00009 column gender status age salary;
00010 define gender / group style={foreground=green};
00011 define status / group style={foreground=blue};
00012 define age / mean style={foreground=red};
00013 define salary / sum style={foreground=brown};
00014 run;
00015
00016 ods html close;
00017 ods listing;
00018
```

Report Color Example using styles



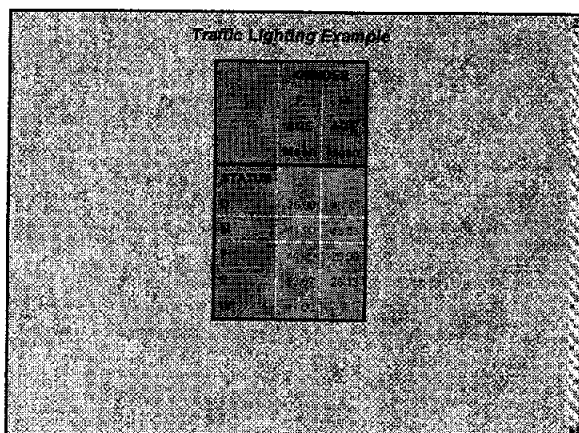
Traffic Lighting

This is a technique that employs styles with formatted values to allow changing the color and presentation of cells as the value changes. Examine the following code.

```

Program Editor traffic
Command ==>
00001 ods html body="c:\traffic";
00002 ods listing close;
00003
00004 title 'Traffic Lighting Example';
00005 proc format;
00006   value agefat
00007     1-18 = 'C-ADNESE'
00008     19-30 = 'Blue'
00009     31-45 = 'Green'
00010     46-65 = 'Red'
00011     other = 'Gray';
00012 run;
00013
00014 proc tabulate data=sasdemo1.us style={foreground=agefat.};
00015   class status gender / s={foreground=black};
00016   classlev status gender / s={foreground=red};
00017   var age;
00018   table status, gender*age*scan;
00019 run;
00020
00021 ods html close;
00022 ods listing;

```



Hyperlinks in Graphs and Reports

Hyperlinks in Graphs

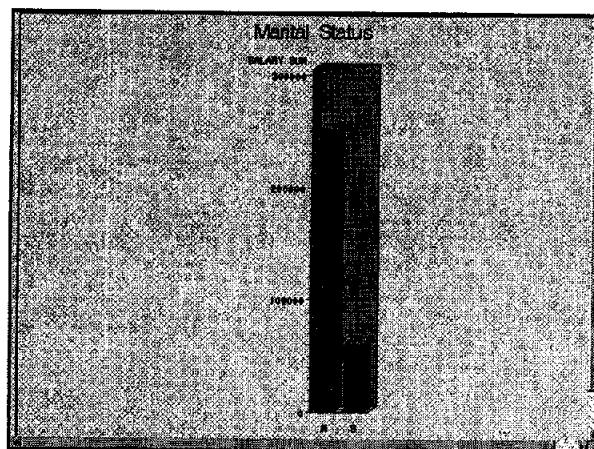
In Version 8 of SAS Software, HTML examples that link to each other are possible. Wouldn't it be great to create a graph and then be able to click on the items in the graph to branch to appropriate detail about those items?

The following code could create that result.

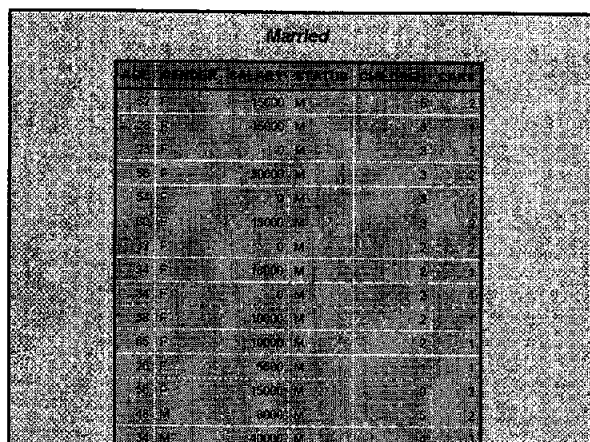
```

Program Editor graph.html
Command ==>
00001 libname sasdata 'd:\sas\data8';
00002 filename odsout 'c:\';
00003 ods listing close;
00004 options reset-global;
00005 data work.demograp;
00006   length linkas $ 40;
00007   set sasdata.demograp;
00008   if status='S' then linkas = 'href="single.html"';
00009   else if status='M' then linkas = 'href="married.html"';
00010 run;
00011 options device=gif transparency;
00012 ods html body="demograp.html" path=odsout;
00013 title 'Marital Status';
00014 proc gchart data=work.demograp;
00015   vbar3d status / suvbar=salary;
00016   html-linkas;
00017   where status in ('S','M');
00018 run;
00019 quit;
00020
00021 ods html body="single.html" path=odsout;
00022 title 'Singles';
00023 proc print data=work.demograp(drop=linkas) noobs;
00024   where status='S';
00025 run;
00026
00027 ods html body="married.html" path=odsout;
00028 title 'Married';
00029 proc print data=work.demograp(drop=linkas) noobs;
00030   where status='M';
00031 run;
00032 ods html close;
00033 ods listing;
00034

```



Click on the appropriate bars to see the detail.



Singles

AGE	GENDER	SALARY	STATUS	PROCESSED	DATE
18	F	10000	Single	1	0
19	F	10000	Single	1	0
20	F	10000	Single	1	0
21	F	10000	Single	1	0
22	F	10000	Single	1	0
23	F	10000	Single	1	0
24	F	10000	Single	1	0
25	F	10000	Single	1	0
26	F	10000	Single	1	0
27	F	10000	Single	1	0
28	F	10000	Single	1	0
29	F	10000	Single	1	0
30	F	10000	Single	1	0
31	F	10000	Single	1	0
32	F	10000	Single	1	0
33	F	10000	Single	1	0
34	F	10000	Single	1	0
35	F	10000	Single	1	0
36	F	10000	Single	1	0
37	F	10000	Single	1	0
38	F	10000	Single	1	0
39	F	10000	Single	1	0
40	F	10000	Single	1	0
41	F	10000	Single	1	0
42	F	10000	Single	1	0
43	F	10000	Single	1	0
44	F	10000	Single	1	0
45	F	10000	Single	1	0
46	F	10000	Single	1	0
47	F	10000	Single	1	0
48	F	10000	Single	1	0
49	F	10000	Single	1	0
50	F	10000	Single	1	0
51	F	10000	Single	1	0
52	F	10000	Single	1	0
53	F	10000	Single	1	0
54	F	10000	Single	1	0
55	F	10000	Single	1	0
56	F	10000	Single	1	0
57	F	10000	Single	1	0
58	F	10000	Single	1	0
59	F	10000	Single	1	0
60	F	10000	Single	1	0
61	F	10000	Single	1	0
62	F	10000	Single	1	0
63	F	10000	Single	1	0
64	F	10000	Single	1	0
65	F	10000	Single	1	0
66	F	10000	Single	1	0
67	F	10000	Single	1	0
68	F	10000	Single	1	0
69	F	10000	Single	1	0
70	F	10000	Single	1	0
71	F	10000	Single	1	0
72	F	10000	Single	1	0
73	F	10000	Single	1	0
74	F	10000	Single	1	0
75	F	10000	Single	1	0
76	F	10000	Single	1	0
77	F	10000	Single	1	0
78	F	10000	Single	1	0
79	F	10000	Single	1	0
80	F	10000	Single	1	0
81	F	10000	Single	1	0
82	F	10000	Single	1	0
83	F	10000	Single	1	0
84	F	10000	Single	1	0
85	F	10000	Single	1	0
86	F	10000	Single	1	0
87	F	10000	Single	1	0
88	F	10000	Single	1	0
89	F	10000	Single	1	0
90	F	10000	Single	1	0
91	F	10000	Single	1	0
92	F	10000	Single	1	0
93	F	10000	Single	1	0
94	F	10000	Single	1	0
95	F	10000	Single	1	0
96	F	10000	Single	1	0
97	F	10000	Single	1	0
98	F	10000	Single	1	0
99	F	10000	Single	1	0
100	F	10000	Single	1	0

The key items to get this to work are the following:

1. Use the GIF driver for graph output.
2. Create a variable attached to the grouped data item that includes a valid hyperlink. Here we use LINKME.
3. Use the HTML= option on the charting statement.

Hyperlinks in Reports

This is an easy task by formatting the values with predefined hyper links. Examine the following code:

```

Program Editor tab.html
Command ==>
00001 ods html body="c:\traffic";
00002 ods listing close;
00003 title 'Hyperlinks in Procedures';
00004 proc format;
00005 value agefat
00006 1-18 = 'cxDAE6E'
00007 19-30 = 'Blue'
00008 31-45 = 'Green'
00009 46-55 = 'Red'
00010 other = 'Gray';
00011 value $status
00012 'D' = '<A href=divorced.html>Divorced</A>'
00013 'M' = '<A href=married.html>Married</A>'
00014 'W' = '<A href=widowed.html>Widowed</A>'
00015 'S' = '<A href=single.html>Single</A>'
00016 'P' = '<A href=separated.html>Separated</A>'
00017 ;
00018 run;
00019 proc tabulate data=saved.demogius style={foreground=agefat.};
00020 class status gender / s={foreground=black};
00021 classlev status gender / s={foreground=red};
00022 var age;
00023 table status,gender*age*mean;
00024 format status $status.;
00025 run;
00026 ods html close;
00027
00028 **** Not all status prints are shown;
00029 ods html body="c:\divorced.html";
00030 proc print data=saved.demogius;
00031 where status='D';
00032 run;
00033 ods html close;
00034
00035 ods listing;
00036

```

Hyperlinks in Procedures

AGE	GENDER	SALARY	STATUS	PROCESSED	DATE
18	F	10000	Single	1	0
19	F	10000	Single	1	0
20	F	10000	Single	1	0
21	F	10000	Single	1	0
22	F	10000	Single	1	0
23	F	10000	Single	1	0
24	F	10000	Single	1	0
25	F	10000	Single	1	0
26	F	10000	Single	1	0
27	F	10000	Single	1	0
28	F	10000	Single	1	0
29	F	10000	Single	1	0
30	F	10000	Single	1	0
31	F	10000	Single	1	0
32	F	10000	Single	1	0
33	F	10000	Single	1	0
34	F	10000	Single	1	0
35	F	10000	Single	1	0
36	F	10000	Single	1	0
37	F	10000	Single	1	0
38	F	10000	Single	1	0
39	F	10000	Single	1	0
40	F	10000	Single	1	0
41	F	10000	Single	1	0
42	F	10000	Single	1	0
43	F	10000	Single	1	0
44	F	10000	Single	1	0
45	F	10000	Single	1	0
46	F	10000	Single	1	0
47	F	10000	Single	1	0
48	F	10000	Single	1	0
49	F	10000	Single	1	0
50	F	10000	Single	1	0
51	F	10000	Single	1	0
52	F	10000	Single	1	0
53	F	10000	Single	1	0
54	F	10000	Single	1	0
55	F	10000	Single	1	0
56	F	10000	Single	1	0
57	F	10000	Single	1	0
58	F	10000	Single	1	0
59	F	10000	Single	1	0
60	F	10000	Single	1	0
61	F	10000	Single	1	0
62	F	10000	Single	1	0
63	F	10000	Single	1	0
64	F	10000	Single	1	0
65	F	10000	Single	1	0
66	F	10000	Single	1	0
67	F	10000	Single	1	0
68	F	10000	Single	1	0
69	F	10000	Single	1	0
70	F	10000	Single	1	0
71	F	10000	Single	1	0
72	F	10000	Single	1	0
73	F	10000	Single	1	0
74	F	10000	Single	1	0
75	F	10000	Single	1	0
76	F	10000	Single	1	0
77	F	10000	Single	1	0
78	F	10000	Single	1	0
79	F	10000	Single	1	0
80	F	10000	Single	1	0
81	F	10000	Single	1	0
82	F	10000	Single	1	0
83	F	10000	Single	1	0
84	F	10000	Single	1	0
85	F	10000	Single	1	0
86	F	10000	Single	1	0
87	F	10000	Single	1	0
88	F	10000	Single	1	0
89	F	10000	Single	1	0
90	F	10000	Single	1	0
91	F	10000	Single	1	0
92	F	10000	Single	1	0
93	F	10000	Single	1	0
94	F	10000	Single	1	0
95	F	10000	Single	1	0
96	F	10000	Single	1	0
97	F	10000	Single	1	0
98	F	10000	Single	1	0
99	F	10000	Single	1	0
100	F	10000	Single	1	0

This paper consists of excerpts from Destiny Corporation's course materials. Copyright © 2000. This material may not be duplicated in any way. Please contact Destiny Corporation for more information.