

Getting Started with, and Getting the Most out of, SAS® ODS PDF: No Mastery of PROC TEMPLATE Required

LeRoy Bessler, Assurant Health, Milwaukee, USA, bessler@execpc.com

Abstract

Adobe's Portable Document Format (PDF) provides the ability to deliver documents that display content exactly as you intended, and that are not easily modified by a recipient. Though SAS ODS (Output Delivery System) does support creation of PDF output, there are several important ways to make ODS do it much better. Some of the means to do so are not obvious, if documented at all. After presenting the basics, this paper shows you how to most effectively combine small PROC outputs, or a table plus a graph, on one PDF page. For larger documents, to replace the default PDF Table of Contents (a.k.a. "Bookmarks"), it shows you how to create a HyperLinked Table of Contents, and CrossLinked PDF pages. For PDF documents with only one output component, the ODS-provided Table of Contents is best suppressed. While pop-up text may be familiar for web pages, that it is also possible for PDF pages is less well known. Here you can learn how to do it. Also presented are: (a) how to make pop-up text and hyperlinks data-dependent; (b) how to create hyperlinks to destinations or other files (including audio clips and video clips) outside your PDF document; and (c) how to imbed and size images in your PDF documents. Some new SAS 9 features are briefly discussed. No prior experience with ODS is assumed.

Introduction

This paper takes you beyond the default results, to get the most out of using ODS for PDF. Some results don't come naturally. Some methods are not easy to find. Some methods are not documented at all. The solution may require ODS Inline Formatting. This means that formatting controls are inline with text, relying on use of an "escape character" to distinguish controls from text. The working context is almost entirely SAS Release 8.2. For more on new features in SAS 9, especially for use of the ODS LAYOUT and ODS REGION features, see "Using SAS 9 ODS Features to Present Table and Graph Data in an Adobe PDF File", by Lori S. Parsons (lparsons@ovation.org).

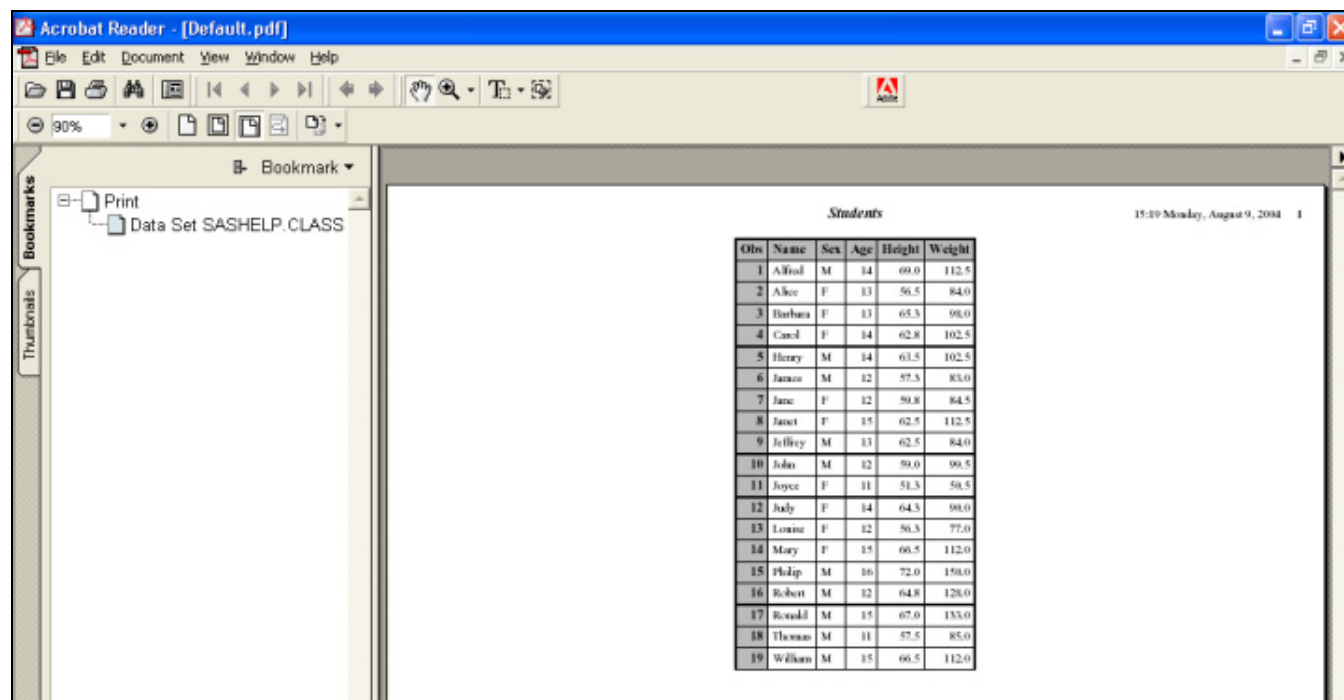
NOTE: To keep the paper compact, screen images have usually been clipped to show only the essential parts of the displayed result.

The ODS Default PDF

We need a target location for the file. First, create a folder, e.g., C:\PDF. All of the examples in this paper will be left generic as to target folder by use of a symbolic parameter. Here, it will be assigned as follows:

```
%let FolderForPDF = %str(C:\PDF);
```

Here is an example of a default format PDF:



By default, you get the index/bookmarks. It should be noted that although the table grid displays in Acrobat Reader with non-uniform thickness, when printed the table grid is uniform.

Multiple Small PROC Outputs on a Single PDF Page

If you run this code:

```
ods pdf file="&FolderForPDF.\MultiProcsPerPageProblem.pdf";
ods pdf StartPage=Never; /* get both PROCs on same page */
title1 'Summary By Age';
title2 color=red 'Summary By Sex title is dropped by SAS';
proc means data=sashelp.class;
class Age; run;
title1 'Summary By Sex';
proc means data=sashelp.class;
class Sex; run;
ods pdf close;
```

you get this surprising result:

Summary By Age
Summary By Sex title is dropped by SAS

The MEANS Procedure

Age	N	Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
11	2	2	Height	2	54.4000000	4.3840620	51.3000000	57.5000000
			Weight	2	67.7500000	24.3951840	56.5000000	85.0000000
12	5	5	Height	5	59.4400000	3.2074232	56.3000000	64.8000000
			Weight	5	94.4000000	20.5286385	77.0000000	128.0000000
13	3	3	Height	3	61.4333333	4.4959241	56.5000000	65.3000000
			Weight	3	88.6666667	8.0829838	84.0000000	98.0000000
14	4	4	Height	4	64.9000000	2.8011902	62.8000000	69.0000000
			Weight	4	101.8750000	9.2138935	98.0000000	112.5000000
15	4	4	Height	4	65.6250000	2.8966243	62.5000000	67.0000000
			Weight	4	117.3750000	10.4193330	112.0000000	133.0000000
16	1	1	Height	1	72.0000000	.	72.0000000	72.0000000
			Weight	1	150.0000000	.	150.0000000	150.0000000

The MEANS Procedure

Sex	N	Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
F	9	9	Age	9	13.2222222	1.3944334	11.0000000	15.0000000
			Height	9	60.5888889	5.0183275	51.3000000	66.5000000
			Weight	9	98.1111111	19.3839137	80.5000000	112.5000000
M	10	10	Age	10	13.4000000	1.6405452	11.0000000	16.0000000
			Height	10	63.9100000	4.9379370	57.5000000	72.0000000
			Weight	10	100.9500000	22.7271864	85.0000000	150.0000000

If you run this code:

```
ods pdf file="&FolderForPDF.\InsertingNotes.pdf";
ods pdf StartPage=Never;
ods noproctitle; /* This option suppresses the useless 'The MEANS Procedure' subtitles.
                  It is NOT code required to restore the missing second title. */

title1 'Summary By Age';
proc means data=sashelp.class;
class Age; run;

ods escapechar='^';
ods pdf text= ^S={font_size=36pt}'; /* spacer */
ods pdf text=^S={just=center font=("Times Roman",13PT,Bold Italic)}Summary By Sex';

proc means data=sashelp.class;
class Sex; run;

ods pdf close;
```

You get acceptable results:

Summary By Age

Age	N	Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
11	2	Height	2	54.4000000	4.3840620	51.3000000	57.5000000	
		Weight	2	67.7500000	24.3951840	58.5000000	85.0000000	
12	5	Height	5	59.4400000	3.2974232	56.3000000	64.8000000	
		Weight	5	94.4000000	20.5286385	77.0000000	128.0000000	
13	5	Height	5	61.4333333	4.4959241	56.5000000	65.5000000	
		Weight	5	88.6666667	8.0829838	84.0000000	98.0000000	
14	4	Height	4	64.9000000	2.8014902	62.8000000	69.0000000	
		Weight	4	101.8750000	9.2138935	98.0000000	112.5000000	
15	4	Height	4	65.6250000	2.8996243	62.5000000	67.0000000	
		Weight	4	117.3750000	10.4193330	112.0000000	133.0000000	
16	1	Height	1	72.0000000	.	72.0000000	72.0000000	
		Weight	1	150.0000000	.	150.0000000	150.0000000	

Summary By Sex

Sex	N	Obs	Variable	N	Mean	Std Dev	Minimum	Maximum
F	9	Age	9	13.2222222	1.7844334	11.0000000	15.0000000	
		Height	9	60.5888889	5.0183275	51.5000000	66.5000000	
		Weight	9	90.1111111	19.3819137	50.5000000	112.5000000	
M	10	Age	10	13.4000000	1.6405452	11.0000000	16.0000000	
		Height	10	63.9100000	4.9370370	57.5000000	72.0000000	
		Weight	10	100.9500000	22.7271864	83.0000000	150.0000000	

Graph of Average Height By Age

Table and Graph on a Single PDF Page

Use this code:

```
options reset=all;

options orientation=landscape;
options printerpath=postscript;

ods pdf file="%FolderForPDF.\TablePlusGraph.pdf";
ods pdf StartPage=Never;

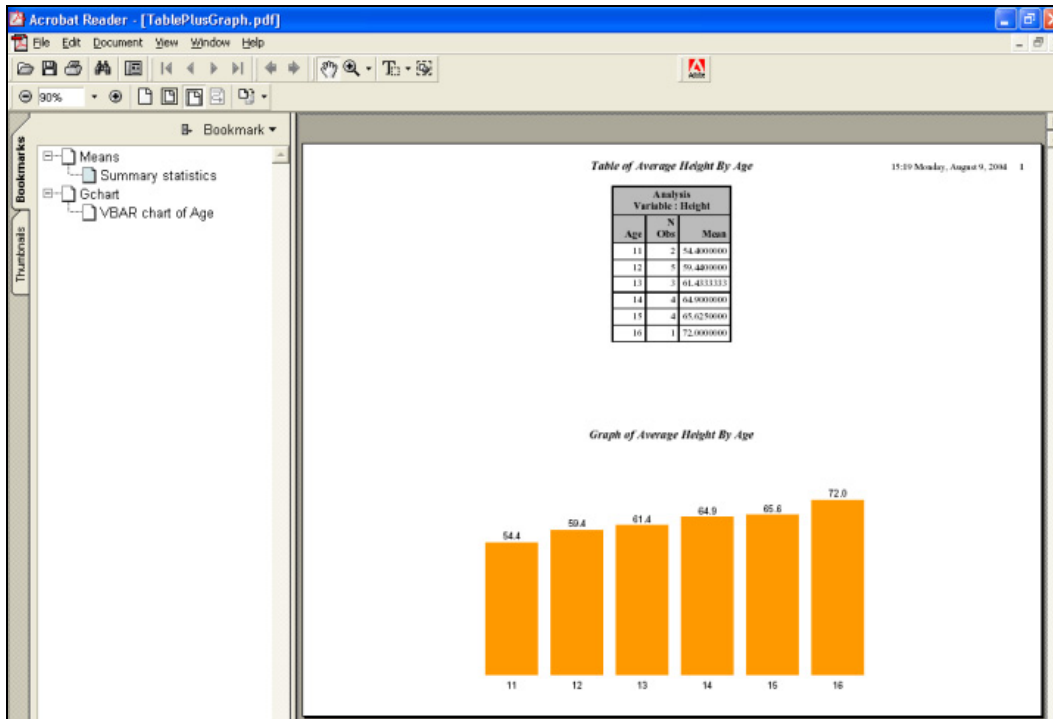
proc means data=sashelp.class mean;
class Age;
var Height;
output out=ToGraph mean=AvgHt;
run; /* PROC MEANS produces a listing by default.
      We also want to use its AvgHt for the graph.*/

options device=pdfc vsize=4 in;
options ftext="Helvetica" htext=10 pt ctext=black;

title1 f="Times/bold/italic" h=13 pt c=black 'Graph of Average Height By Age';
/* font=          and          height= to match PDF default formatting of the table title */
proc gchart data=ToGraph; /* use the data from PROC MEANS */
pattern1 v=solid c=CXFF9900; /* orange */
axis1 style=0 major=none minor=none label=none value=none;
axis2 style=0 major=none minor=none label=none;
vbar Age / sumvar=AvgHt sum discrete raxis=axis1 maxis=axis2;
format AvgHt 4.1;
run; quit;

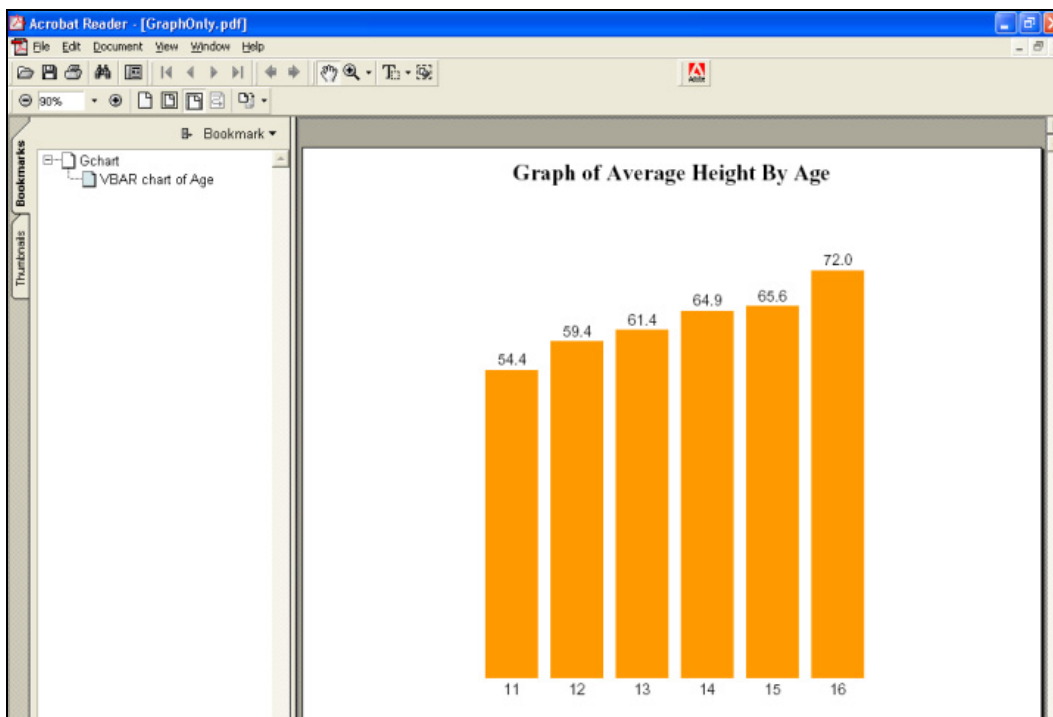
ods pdf close;
```

For this result:



For a Graph Only, use the same code as before EXCEPT:

```
ods pdf file="&FolderForPDF.\GraphOnly.pdf";
/* ods pdf StartPage=Never; Not Needed Here */
proc means data=sashelp.class mean NOPRINT;
. . .
/* goptions . . . vsize=4 in; Not Needed Here */
goptions ftext="Helvetica" htext=16 pt ctext=black;
title1 f="Times/bold" h=24 pt c=black 'Graph of Average Height By Age';
```

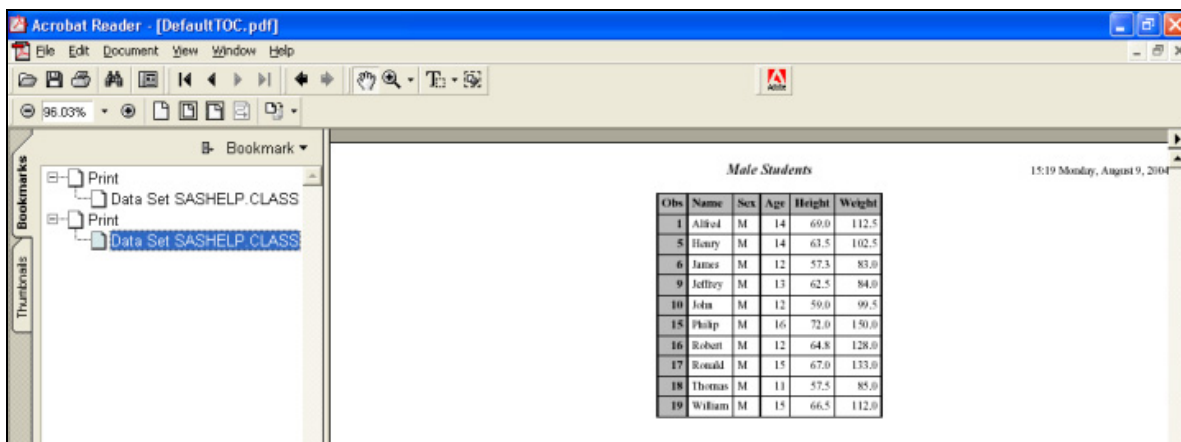
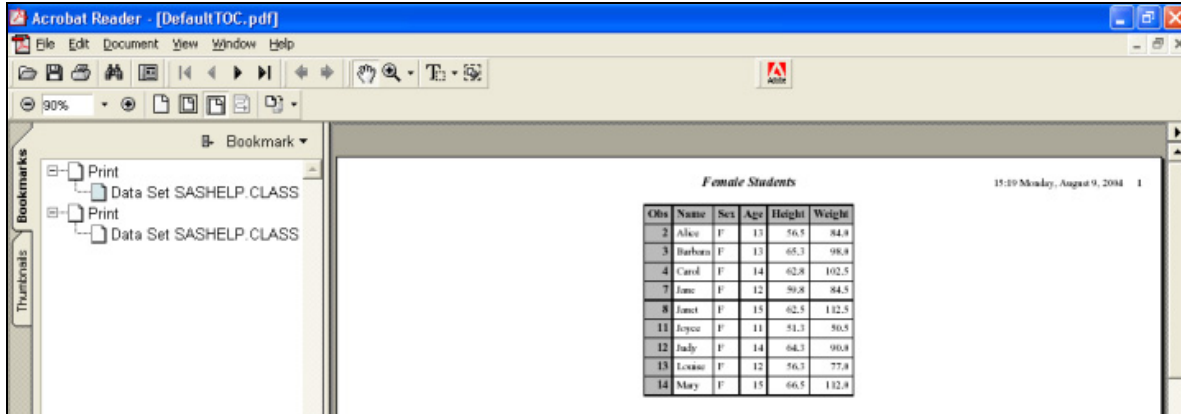


Default PDF Table of Contents

To see the ODS Default Table of Contents, use this code:

```
options reset=all;
ods pdf file="&FolderForPDF.\DefaultTOC.pdf";
title1 'Female Students';
proc print data=sashelp.class(where=(Sex eq 'F')); run;
title1 'Male Students';
proc print data=sashelp.class(where=(Sex eq 'M')); run;
ods pdf close;
```

For these results:



Clipping and magnifying the not very informative Table of Contents:



Customizing ODS PDF “Bookmarks” is possible, but limited. It can be, at least for me, especially frustrating for more complex output such as BY processing results. It may not be worth the trouble. A Table of Contents page, using hyperlinks, may be a better alternative.

Hyperlinked Table of Contents

Minimal code for a hyperlinked table of contents page is as follows:

```
ods escapechar="^";
ods pdf file="&FolderForPDF.\DefaultHyperlinkedTOC.pdf"
    notoc; /* this option turns off the bookmarks window */

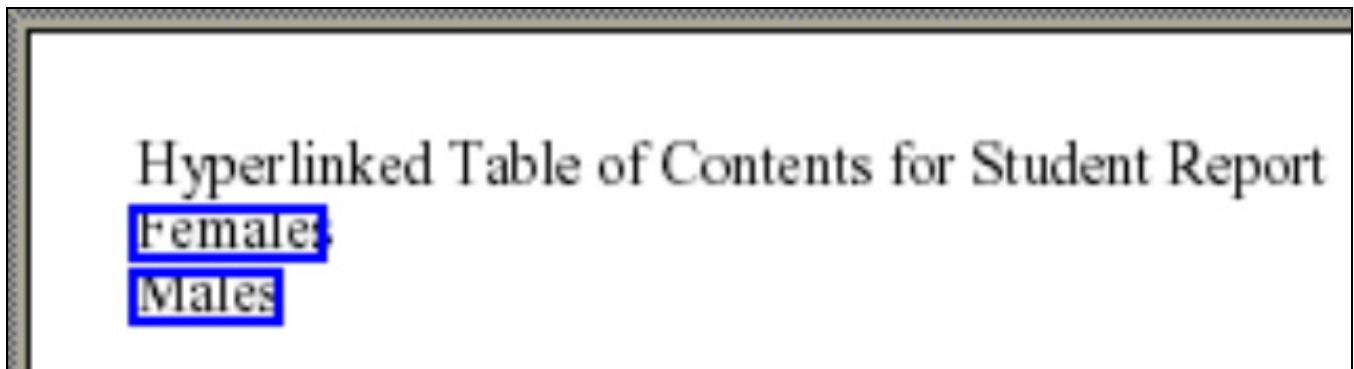
ods pdf text = "Hyperlinked Table of Contents for Student Report";
ods pdf text = "^S={URL='#Girls'}Females";
ods pdf text = "^S={URL='#Boys'}Males";

ods pdf anchor = "Girls";
title1 'Female Students';
proc print data=sashelp.class(where=(Sex eq 'F')); run;

ods pdf anchor = "Boys";
title1 'Male Students';
proc print data=sashelp.class(where=(Sex eq 'M')); run;

ods pdf close;
```

Unfortunately, the result is this (clipping and magnifying the front-page table of contents for the PDF document):



A custom READABLE hyperlinked table of contents page can be made with this code:

```
/* All same as before, EXCEPT add formatting */

/* set it up as a macro variable, to avoid duplicative coding */
%let TOCfmt = %str(JUST=CENTER VJUST=MIDDLE CELLHEIGHT=20PT FONT_SIZE=10PT CELLWIDTH=15CM);

/* Apply the in-line formatting: */
ods pdf text="^S={&TOCfmt}Drillable Table of Contents for Student Report";
ods pdf text="^S={&TOCfmt URL='#Girls'}Females";
ods pdf text="^S={&TOCfmt URL='#Boys'}Males";
```

And this is the result (clipped from the table of contents page):

Drillable Table of Contents for Student Report	
Females	
Males	

Hyperlinked Table of Contents and CrossLinked PDF Pages

A typical user of a multi-page PDF may want to go from page to page without always going back to the Table of Contents page. Though the bookmarks window in the default multi-page PDF has usability or implementability shortcomings, it is conveniently located. But it does come at the expense of shrinking the view of the current page of interest.

Here is the overall code structure:

```
goptions reset=all;
ods escapechar="^";
ods pdf notoc file="%FolderForPDF.\HyperlinkedTOCandCrossLinks.pdf";
%let TOCfmt = %str( same formatting controls as before );

/* insert three code blocks here, with this content:

Table of Contents Page
  Link to Female Students
  Link to Male Students

Female Students Page
  Table of Female Students
  Link to Male Students
  Link to Table of Contents

Male Students Page
  Table of Male Students
  Link to Female Students
  Link to Table of Contents

*/

ods pdf close;
```

Below is the code block for the TOC Page. Though it contains nothing but text and hyperlinks, the table of contents page must be generated with a SAS PROC. The PROC step used produces nothing but an invisible one-line table.

```
ods pdf startpage=never; /* multi-output page */
ods pdf anchor="TOC";

title1; /* no title */
proc print data=sashelp.class(keep=Name where=(Name eq 'Carol'))
  noobs /* turns off the observation count column */
  style      = [rules=none frame=void cellspacing=0] /* turns off the table grid */
  style(header) = [foreground=white background=white] /* invisible column heading */
  style(data)   = [foreground=white background=white]; /* invisible data cell */
run;

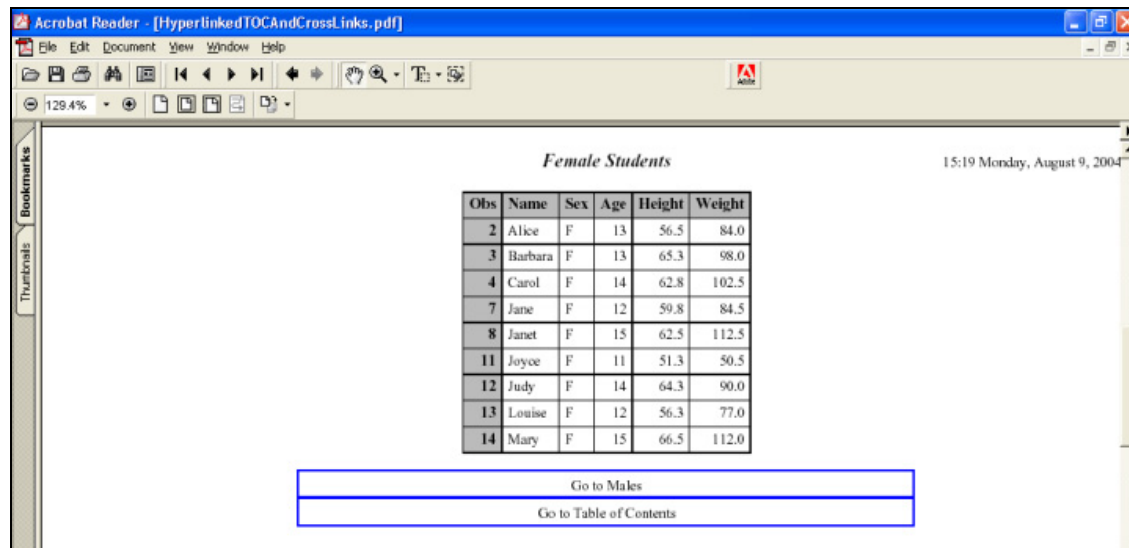
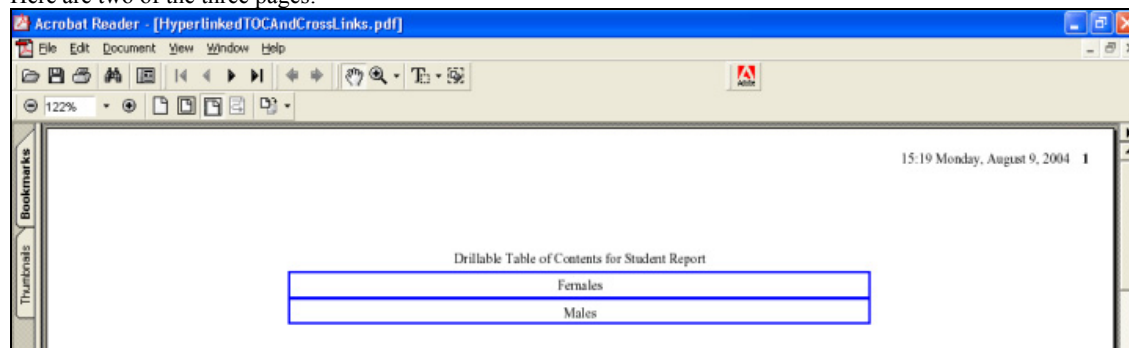
ods pdf text = "^S={&TOCfmt}Hyperlinked Table of Contents for Student Report";
ods pdf text = "^S={&TOCfmt URL='#Girls'}Females";
ods pdf text = "^S={&TOCfmt URL='#Boys'}Males";
/* Above three lines are footnotes below the invisible PROC output on the page. */
```

Here is the code for the Table of Female Students, with links to the rest of the document:

```
ods pdf startpage=never;
ods pdf anchor="Girls";
title1 'Female Students';
proc print data=sashelp.class(where=(Sex eq 'F')); run;
ods pdf text="^S={&TOCfmt URL='#Boys'}Go to Males";
ods pdf text="^S={&TOCfmt URL='#TOC'}Go to Table of Contents";
/* Above lines are footnotes on the Girls page. */
```

Analogous code for the Table of Male Students, with links to the rest of the document, presumably is obvious.

Here are two of the three pages:

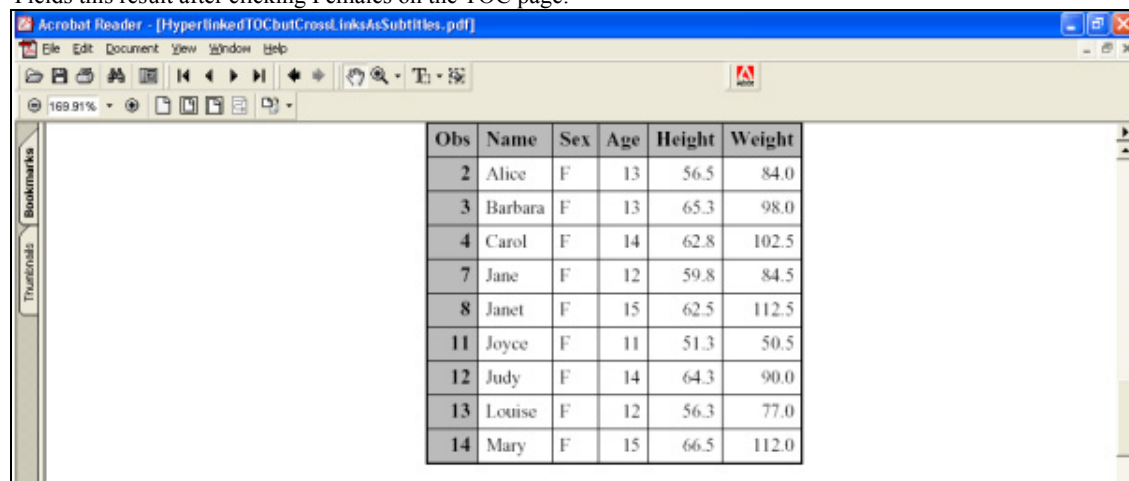


Why Footnotes for the Hyperlinks Rather Than Subtitles?

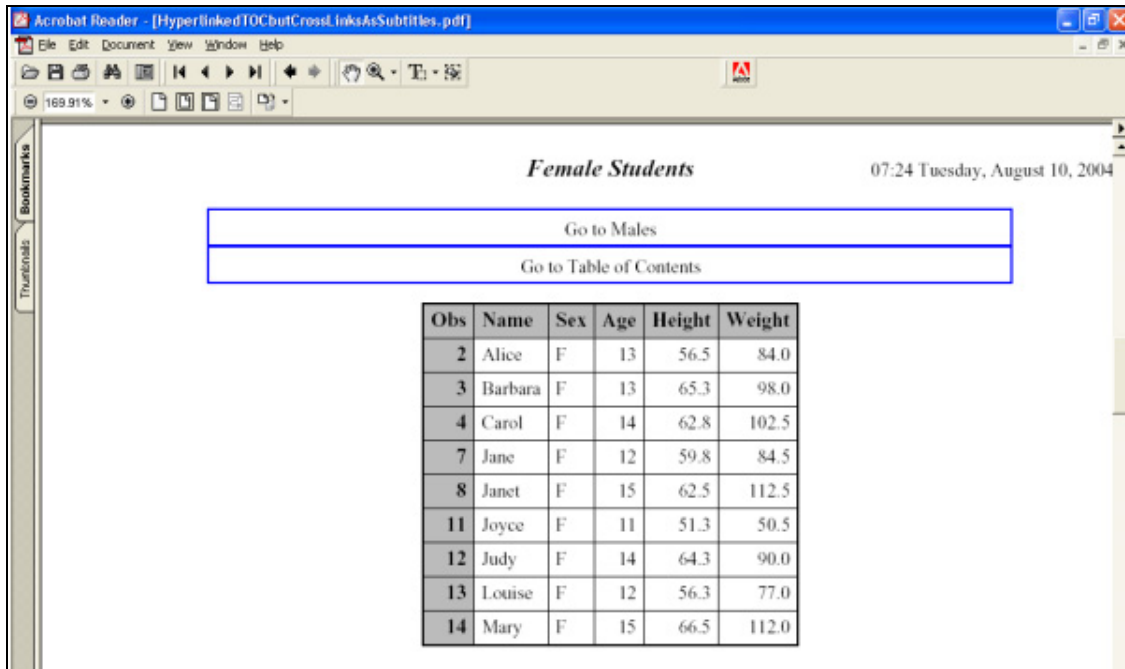
Subtitles are, in fact, equally easy to create. But linking to the non-TOC pages opens those pages awkwardly. For example, using this code:

```
ods pdf startpage=never;
ods pdf anchor="Girls";
title1 'Female Students';
ods pdf text="^S={&TOCfmt URL='#Boys'}Go to Males";
ods pdf text="^S={&TOCfmt URL='#TOC'}Go to Table of Contents";
/* Above lines are subtitles on the Girls page. */
proc print data=sashelp.class(where=(Sex eq 'F')); run;
```

Yields this result after clicking Females on the TOC page:



If you scroll upward, you indeed find the page title and the links:



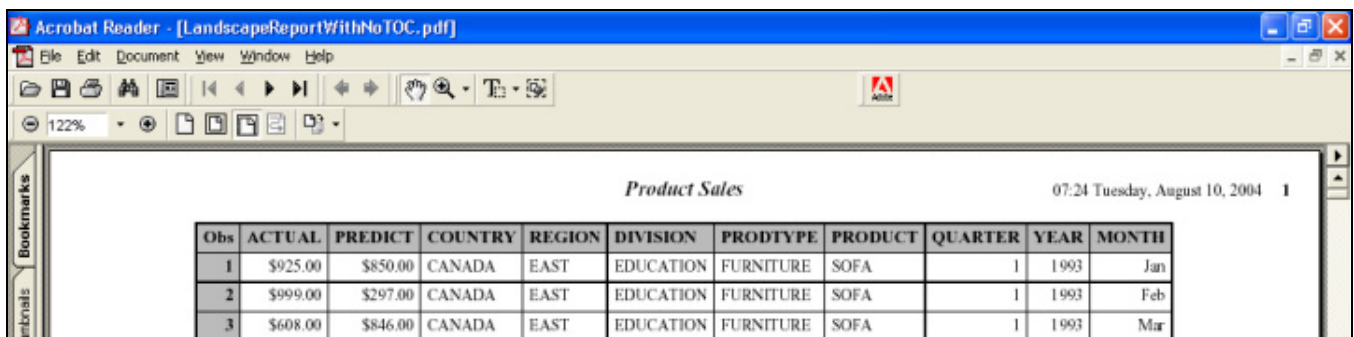
Since scrolling upwards to find the title and to find the hyperlinks as subtitles would be an unacceptable nuisance, titles and subtitles are better used to describe page content, not to list hyperlinks.

No Table of Contents Can Be a Good Thing

NOTOC may be a good option for all/any single PROC output when using PDF for screen display. A data-dense landscape display can use the full screen, and unused clutter is removed from the screen. One simple change, the NOTOC option, makes it go away.

```
options reset=all;
options orientation=landscape;
ods pdf notoc file="%FolderForPDF.\LandscapeReportWithNoTOC.pdf";
title 'Product Sales';
proc print data=sashelp.prdsale; run;
ods pdf close;
```

with this result:

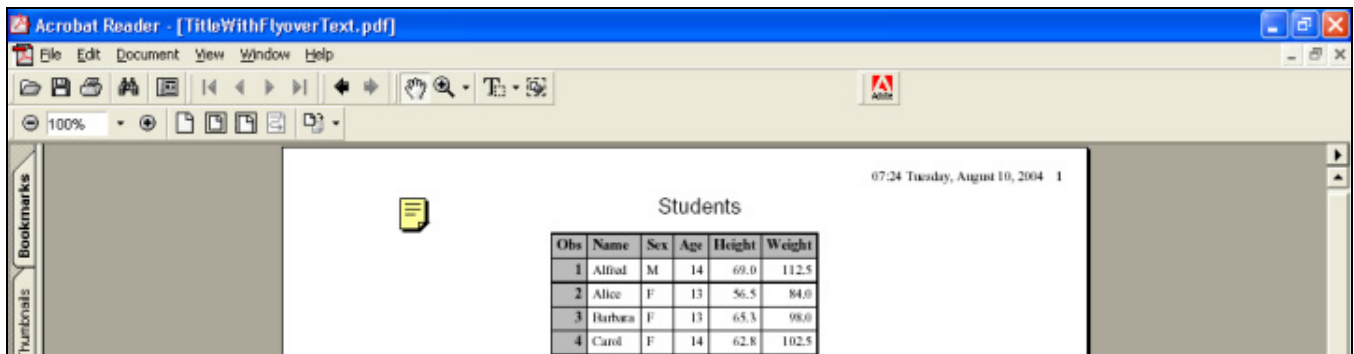


Pop-Up Text

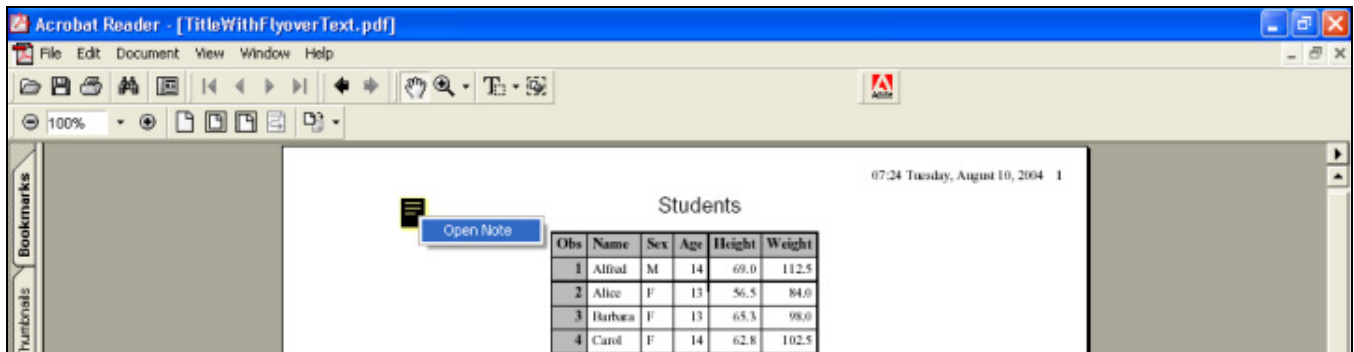
If you would like some information as flyover/pop-up text, not hardcopy, in your PDF document, you can create a (scrollable) note box “behind” an icon. The icon does not print, and the note does not print. You must right-click the icon to open the note box. There is no apparent limit on length of note text. You can define the width of the note box. Increasing its height pushes the table around. If you take the default box width, the icon will overlay the left of the “hard text” on the page. With this code:

```
goptions reset=all;
options orientation=portrait;
ods pdf notoc startpage=never file="&FolderForPDF.\TitleWithFlyoverText.pdf";
ods pdf
text="^S={FLYOVER='This is the CLASS data set from the SASHELP data library in SAS Software'
      CELLWIDTH=6IN
      JUST=CENTER
      FONT_SIZE=16PT
      FONT_FACE=Helvetica}Students";
proc print data=sashelp.class; run;
ods pdf close;
```

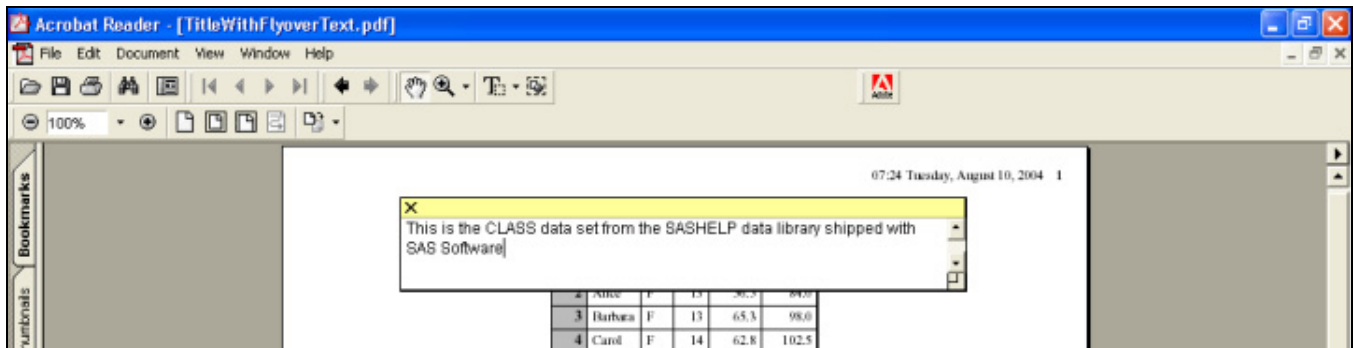
You get this result, starting with display of the icon:



Resting the mouse on the icon:



Opening the note:



Data-Dependent Pop-Up Text in Table Cells

Use a SAS format to selectively define the pop-up text based on data value, with code such as this:

```
proc format lib=work;
value AgeNote
    11 = 'Youngest'
    16 = 'Oldest'
    Other = ' ';
run;

options reset=all;
options orientation=portrait;
ods pdf notoc file="%FolderForPDF.\SomeDataCellsWithFlyoverText.pdf";
proc print data=sashelp.class;
var Name Sex;
var Age / style=[CELLWIDTH=1.2IN /* widen cells to prevent the notes from overlaying data */
               just=right flyover=AgeNote.] ;
var Height Weight;
run;
ods pdf close;
```

Here are clipped versions of the PDF page, to show the data-dependent pop-up text:

10	John	M	12	59.0	99.5
11	Joyce	F	11	51.3	50.5
12	Judy	F	14	64.3	90.0
13	Louise	F	12	56.3	77.0
14	Mary	F	15	66.5	112.0
15	Philip	M	16	72.0	150.0
16	Robert	M	12	64.8	128.0
17	Ronald	M	15	67.0	133.0
18	Thomas	M	11	57.5	85.0

10	John	M	12	59.0	99.5
11	Joyce	F	11	51.3	50.5
12	Judy	F	14	64.3	90.0
13	Louise	F	12	56.3	77.0
14	Mary	F	15	66.5	112.0
15	Philip	M	16	72.0	150.0
16	Robert	M	12	64.8	128.0
17	Ronald	M	15	67.0	133.0
18	Thomas	M	11	57.5	85.0

Data-Dependent Hyperlinks in Table Cells

Use a SAS format to selectively define the hyperlink based on the data value, with code such as this:

```
proc format lib=work;
value $NameLnk
'Alice' = '#Alice'
'Carol' = 'http://www.sas.com'
Other = ' '; /* no link */
run;

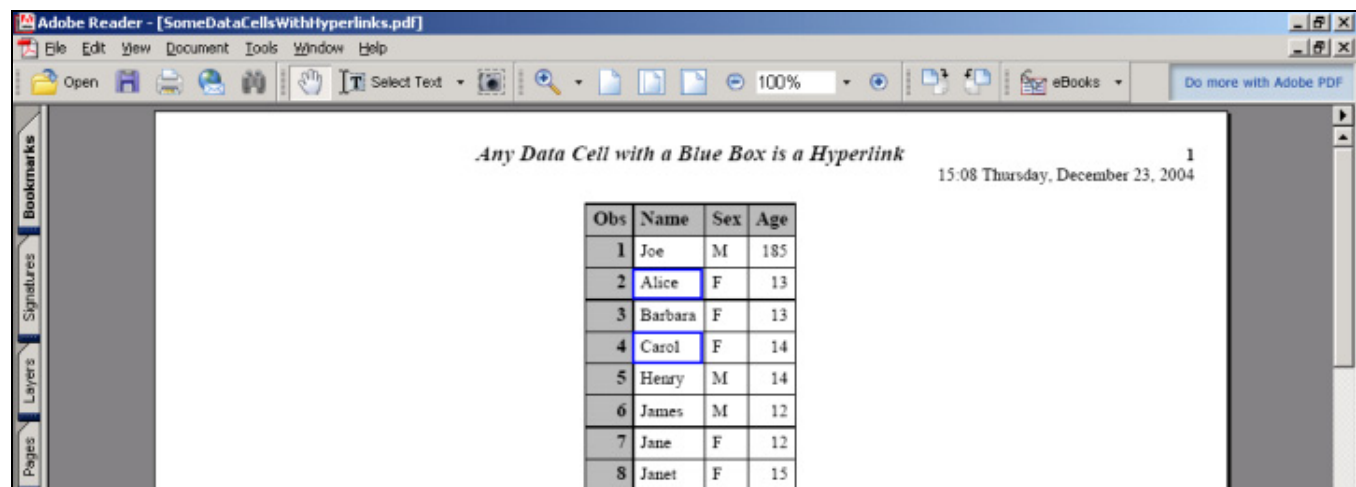
options reset=all;
options orientation=portrait;
ods pdf notoc file="%FolderForPDF.\
    SomeDataCellsWithHyperlinks.pdf";

title 'any data cell with blue box is hyperlink';
proc print data=sashelp.class;
var Name / style=[URL=$NameLnk.] ;
var Sex Age; run;
```

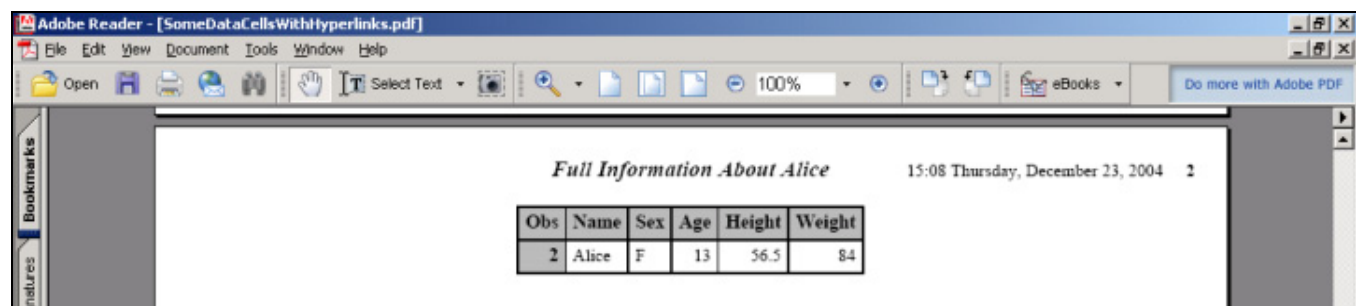
```
ods pdf anchor="Alice";
title1 'Full Information About Alice';
proc print data=sashelp.class
      (where=(Name eq 'Alice'));
run;

ods pdf close;
```

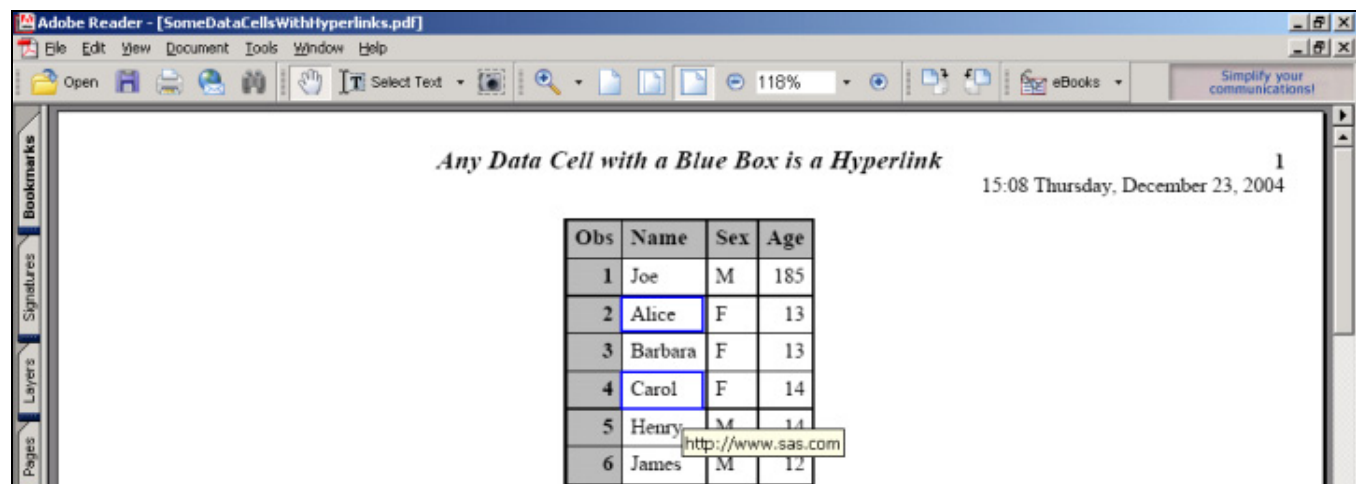
Here is a series of PDF pages to demonstrate the results:



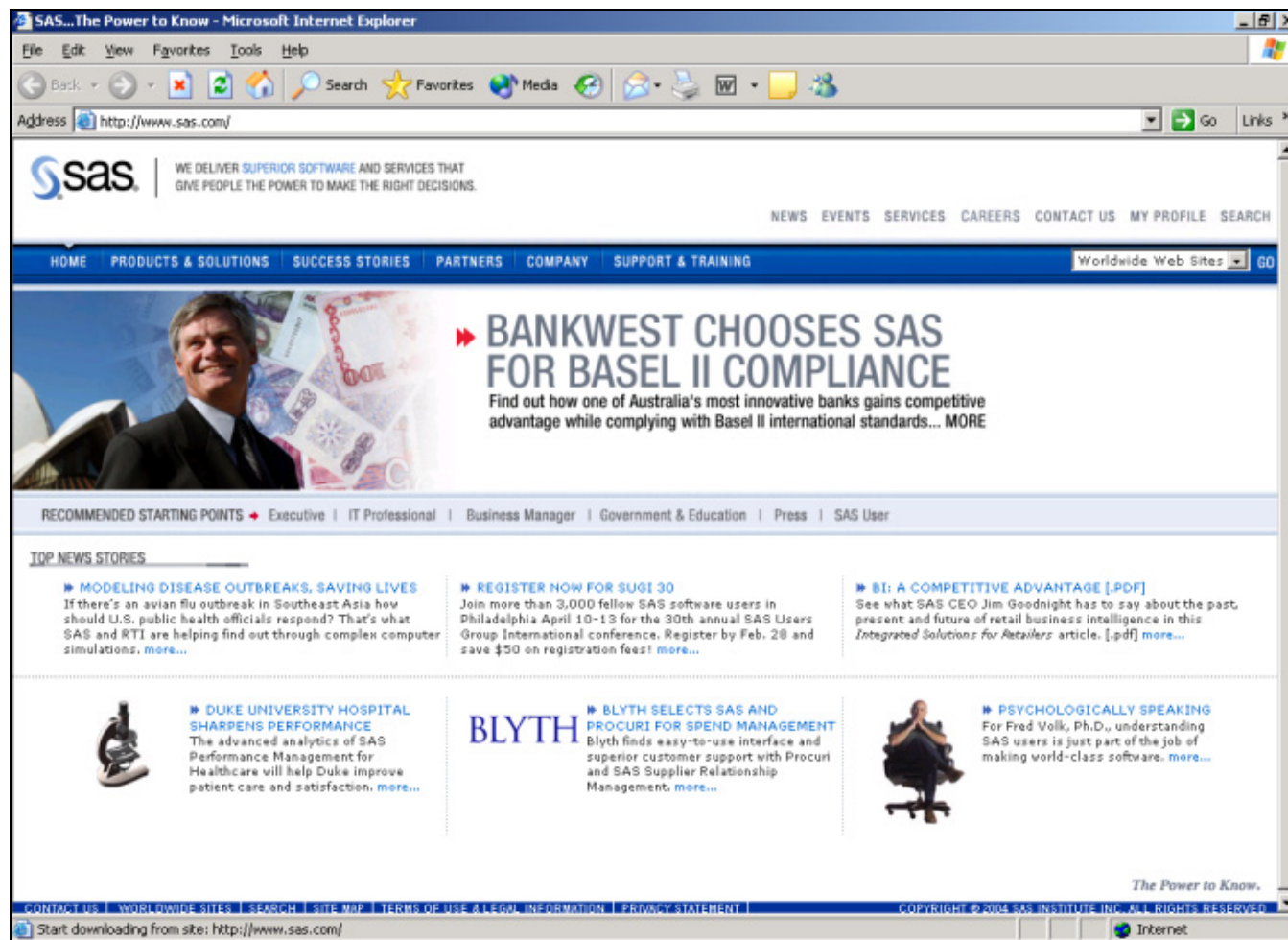
Clicking on the Name cell for Alice:



Resting on the Name cell for Carol:



Clicking on the Name cell for Carol:



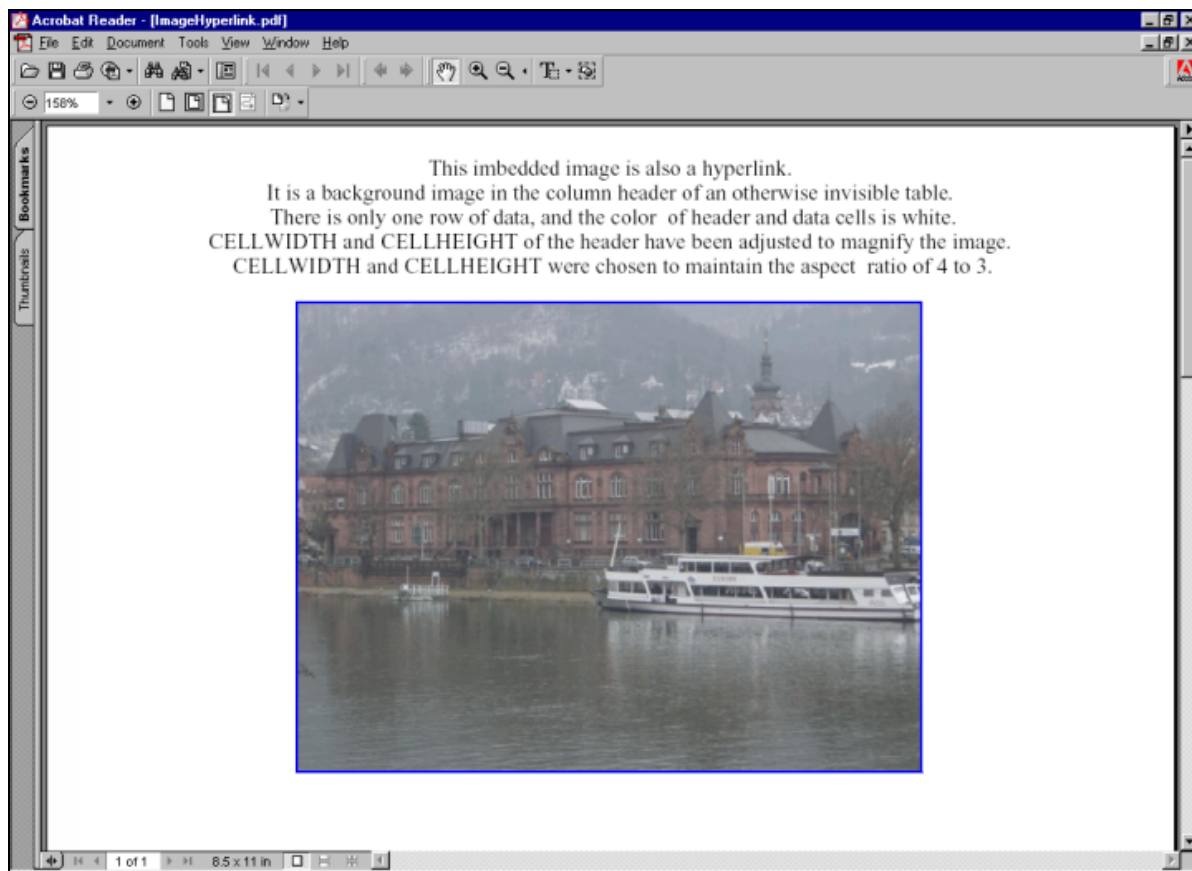
Using an Image As a Hyperlink

PREIMAGE and/or POSTIMAGE and/or BACKGROUNDIMAGE options can be used anywhere in SAS tabular report PROC output to display an image (or multiple images). The URL option can be used in TITLE, FOOTNOTE, or PDF TEXT statements, or in cells for data or for column headers, to create the hyperlink(s). If you create it as a GIF file with SAS/GRAPH, you can use a graph as a hyperlink.

Omitting the ODS code and the TITLE statements, code that can be used to display an image hyperlink is:

```
proc print data=sashelp.class
  (where=(name='Alfred')) noobs label
  style(table) =[frame=void]
  style(header)=[url='http://www.sas.com'
    backgroundimage="&Path.\images\photo1.jpg"
    background=white
    cellwidth=12 cm cellheight=9 cm] /* magnify,
                                     but preserve the 4 X 3 aspect ratio */
  style(data)  =[foreground=white
    font_size=1 pt];
var age;
label age='00'X;
run;
```

The PROC PRINT creates an invisible table as the frame for the image/hyperlink. Here is the result:



Putting It All Together: Image, Audio, Video for a Multi-media Extravaganza

I collected the image for this example PDF document as follows. At <http://www.phuse.info/photos/photo1.html>, I right-clicked on the image, then selected "Save Picture As", and stored the image (without renaming) as photo1.jpg in C:\PDF\images.

I collected the sound clip for this example PDF document as follows. I downloaded the Bavarian Anthem sound file from david.national-anthems.net/bav.htm, and stored it (without renaming) as bav.wav in C:\PDF\audios.

I discovered and collected the video clip for this example PDF document as follows. Using Microsoft Windows Explorer, I searched for files of the form *.avi. From the list of files found, I selected one in Program Files for SAS (in C:\Program Files\SAS Institute\SAS\V8\core\sasmisc) and stored it (without renaming) as tiger.avi in C:\PDF\videos.

```
%let urlPath = %str(C:/PDF);
%let filePath = %str(C:\PDF);
%let TOCfmt = %str(JUST=CENTER VJUST=MIDDLE FONT_SIZE=12PT CELLWIDTH=10CM CELLHEIGHT=24PT);
%let TXTfmt = %str(JUST=CENTER VJUST=MIDDLE FONT_SIZE=12PT CELLWIDTH=20CM CELLHEIGHT=24PT);

proc format;
value $AVlnks
'Inside Stadthalle' = "&urlPath./images/photo6.jpg"
'Anthem of Bavaria' = "&urlPath./audios/bav.wav"
'SAS Video' = "&urlPath./videos/tiger.avi";
run;

data work.ChoicesForImageAudioVideo;
length medium $ 17;
medium = 'Inside Stadthalle'; output;
medium = 'Anthem of Bavaria'; output;
medium = 'SAS Video'; output;
run;
```

```

options nodate nonumber orientation=landscape;
ods escapechar="^";
ods noresults;
ods listing close;
ods pdf notoc file=" . . . ";
ods pdf startpage=never;

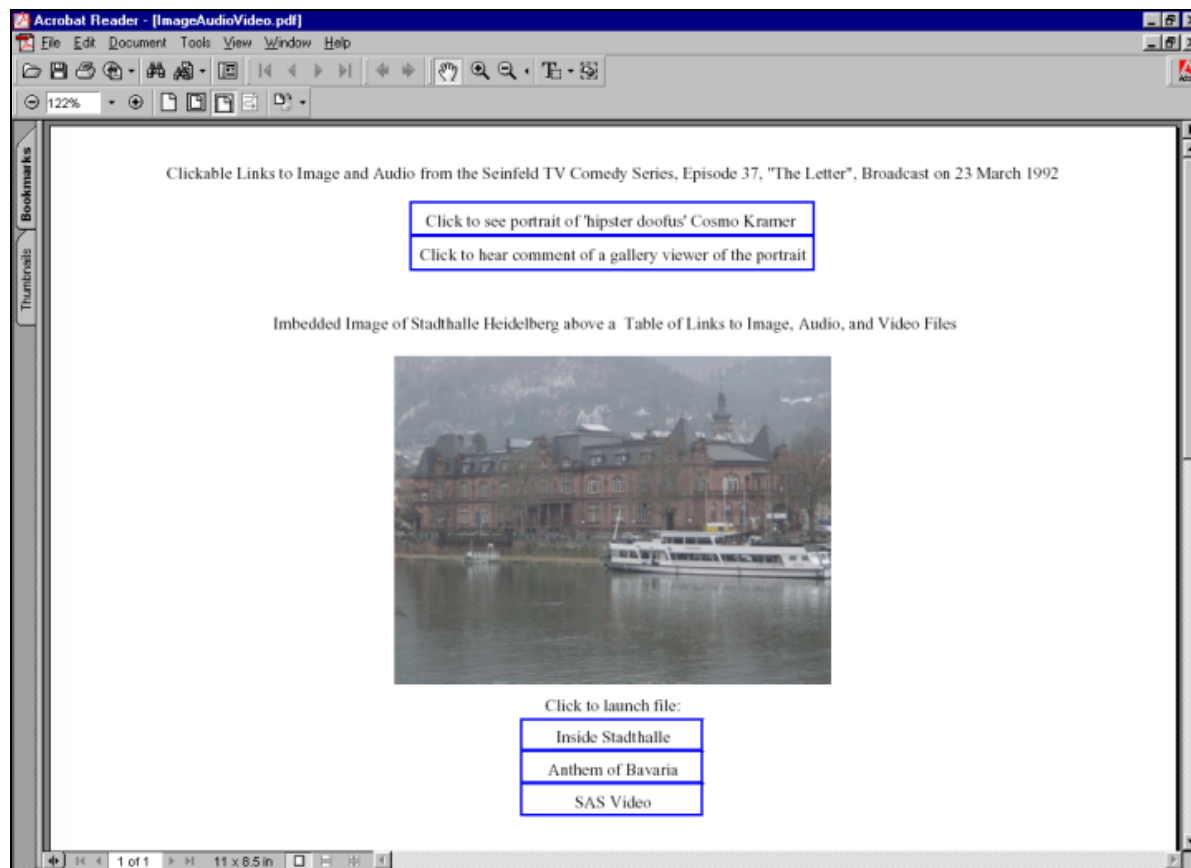
title1 . . . ;
title2 . . . ;
ods pdf text="^S={&TOCfmt URL=" "&urlPath./images/CosmoKramer.jpg"}Click . . . ";
ods pdf text="^S={&TOCfmt URL=" "&urlPath./audios/brute.wav"} . . . ";
ods pdf text="^S={&TXTfmt}";
ods pdf text="^S={&TXTfmt} . . . ";

proc print data=work.ChoicesForImageAudioVideo noobs label
  style=[preimage="&filePath.\images\photo1.jpg" frame=void];
var medium /
  style(data) =[URL=$AVlnks.
    font_size=12pt
    cellwidth=4.5 cm just=center
    cellheight=0.8 cm vjust=middle]
  style(header)=[background=white
    font_size=12pt
    font_weight=medium
    just=center
    cellheight=0.8 cm vjust=middle];
label Medium='Click to launch file: ';
run;

ods pdf close;
ods listing;

```

Here is the result, but no screen images of the linked-to destinations:



Miscellaneous PDF Features: author=, title=, subject=, keywords=

Providing information for the PDF document Properties tab is accomplished with code such as this:

```
ods pdf file      = "&Path.\filename.pdf"
    author      = 'SAS Author'
    title       = 'SAS Paper'
    subject     = 'SAS Subject'
    keywords    = 'SAS ODS PDF output';
. . .
ods pdf close;
```

SAS 9 PDF Compression

Compressing the PDF output simply requires adding a parameter to the ODS PDF statement:

```
ods pdf file="&Path.\filename.pdf" compress=N . . . ;
```

N is integer in range 0 to 9, where 0 is for No Compression, 9 is for Maximum Compression, and the integers in between are for increasing amounts of compression. I found that 1 delivers the maximum incremental effect, and the other added steps deliver smaller increments.

Miscellaneous Tips

Use

```
goptions reset=all;
```

before each ODS code block. Even if you make no use of SAS/GRAPH, this will prevent any surprises. SAS/GRAPH GOPTIONS can have unexpected effects on what is ostensibly non-graphic output.

If you want to voluntarily open your PDF output, rather than have Adobe Acrobat automatically launched inside your SAS session upon completion of your processing, use this statement:

```
ods noresults;
```

Some times I found it necessary to omit http:// from a web-target URL. Other times, I may have found it necessary to include it.

Acknowledgements

Thanks to Chevell Parker for technical advice, and to Deborah Skinner who shared some of her ODS PDF knowledge with me.

Notices

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other product and brand names are trademarks or registered trademarks of their respective owners.

Author Information

Your questions, comments, suggestions, and other solutions are always welcome.

LeRoy Bessler PhD

Email: bessler@execpc.com

Phone: 1 414 351 6748 (evenings and weekends—time is six hours earlier than GMT)

Dr. LeRoy Bessler has special interests in: Software-Intelligent application development, which yields SAS solutions that are reliable, reusable, maintainable, and extendable; and in communication-effective design and construction of reports, tables, graphs, maps, spreadsheets, and web pages.