HTML for the SAS® Programmer

Lauren Haworth
Kaiser Permanente Center for Health Research
Portland, Oregon

➢ ABSTRACT

With more and more output being delivered via the Internet, a little knowledge of HTML can go a long way toward improving the appearance of your output. This paper introduces some simple HTML coding techniques that are useful for SAS programmers. The paper will show how your SAS output is converted into HTML, and demonstrate HTML tricks that you can use in your SAS code to dress up your SAS HTML output.

All of the examples can be produced using either SAS version 6 or version 7.

➢ INTRODUCTION: HOW TO CREATE HTML OUTPUT FROM SAS

There are several ways you can move your SAS output to the web. If you are going to be doing a lot of web publishing, you may want to investigate the SAS/InterNet® product. However, if you just want to publish the occasional table, report, or printout on the web, you can do this with SAS\BASE®.

To put your output on the web, you need to convert standard SAS output into HTML files. HTML stands for HyperText Markup Language, and it is the common language understood by web browsers like Netscape and Internet Explorer.

To create HTML-formatted output using SAS version 6, you can download HTML formatting tools from the SAS web site. These include a macro that will format output from any procedure to HTML and a specialized macro that will convert a PROC TABULATE table to an HTML table. There is also a macro that lets you display a SAS dataset on the web. These are all free products.

To create HTML-formatted output using SAS version 7, all you have to do is use the new Output Delivery System and specify HTML as the output format.

Before getting into the details of syntax for creating HTML output under either version 6 or version 7, let's look at an actual HTML file to see how it works.

➢ WHAT YOUR OUTPUT LOOKS LIKE ON THE WEB

When you create an HTML file from your SAS output, the usual SAS output format of text and numbers separated by spaces is "marked up" with HTML "tags." These tags tell Internet browsers how to display your output. They identify headers and footers, specify fonts and type sizes, reference images, and point to other locations on the web (hyperlinks).

For example, the following PROC PRINT output …

```
Cuppa Coffee Co.
Daily Sales Totals


drink        price       volume
Latte        $2.20       42,655
Cappucino    $2.30       31,985
Mocha        $2.80       24,677


Note: decaffeinated beverages not included.
```

… would look like this on the web …
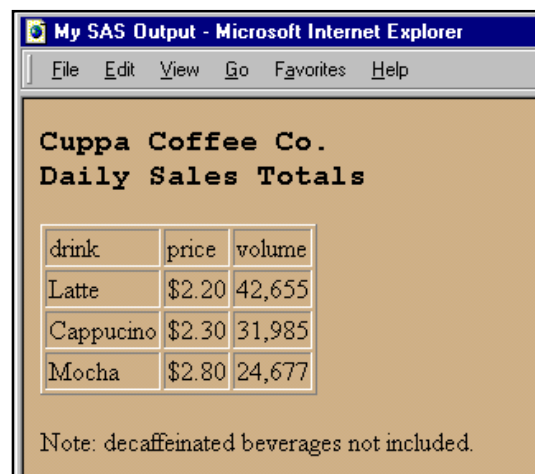
… and would contain the HTML code shown in Figure 1.

Figure 1: HTML code

`<HTML>` ———————————— <HTML> </HTML> surrounds all code, and identifies it to the browser as HTML code.

```
<HEAD>
<TITLE>My SAS Output</TITLE>
</HEAD>
```

<HEAD> </HEAD> surrounds your header. In this section, you will find general information that applies to the entire file. The <TITLE> </TITLE> tags identify the text to display in the browser's title bar when the page is viewed.

`<BODY BGCOLOR=TAN>`

<BODY> </BODY> tags enclose the "guts" of your HTML output. All of the content that will be displayed on the page is contained between these two tags. The BGCOLOR parameter controls the background color for the page

```
<PRE><H3>Cuppa Coffee Co.
Daily Sales Totals
</H3></PRE>
```

<PRE> </PRE> enclose a lines of text that you want to have displayed exactly as is, including any spaces and line breaks. <H3> </H3> enclose lines that you want formatted as a level 3 header. This gives them a larger and bolder font than the rest of the body text.

`<P>` ———————————— <P> marks a break between paragraphs. It puts a large break between the lines above and below. If you don't want such a large break, use the <BR> tag instead.

`<TABLE BORDER=1>`

<TABLE> </TABLE> enclose the code for a table. BORDER=1 controls the width of the borders around and within the table.

The <STRONG> </STRONG> tags call for a bolder typeface for the first row of the table.

`<STRONG>`

The <TR> </TR> tags mark the beginning and end of each table row. The <TD> </TD> tags mark the beginning and end of each table cell in the row.

```
<TR>   <TD>drink</TD>      <TD>price</TD>   <TD>volume</TD>  </TR>
</STRONG>

<TR>    <TD>Latte</TD       <TD>$2.20</TD>   <TD>42,655</TD>  </TR>
<TR>   <TD>Cappucino</TD>  <TD>$2.30</TD>   <TD>31,985</TD>  </TR>
<TR>   <TD>Mocha</TD>       <TD>$2.80</TD>   <TD>24,677</TD>  </TR>

</TABLE>
```
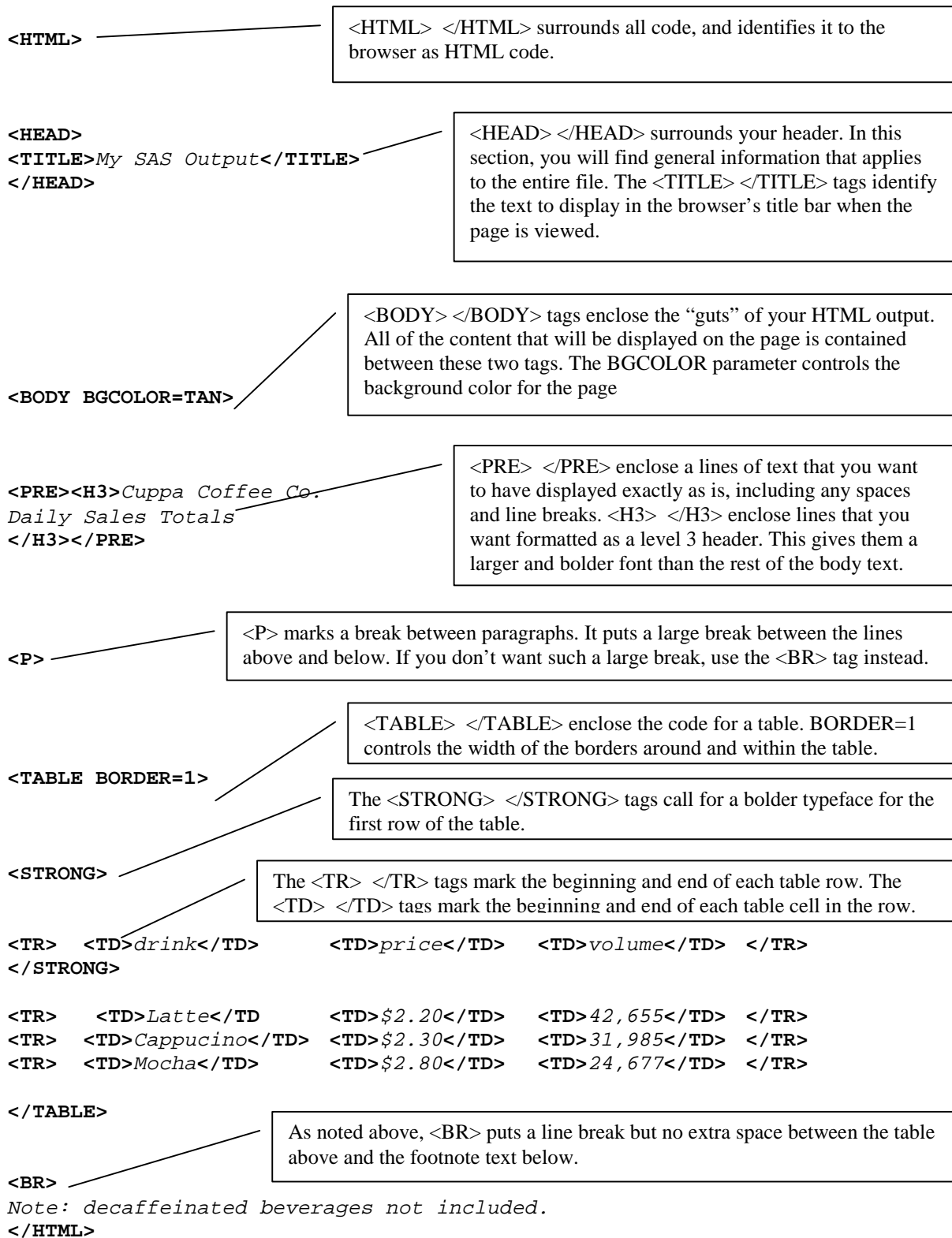
As noted above, <BR> puts a line break but no extra space between the table above and the footnote text below.

```
<BR>
Note: decaffeinated beverages not included.
</HTML>
```

As you review the information in Figure 1, you will see that the HTML file is composed of the basic text of the SAS output (shown in italics), surrounded by numerous HTML tags enclosed in angle brackets (shown in bold face).

If you look closer, you will also see that HTML tags come in pairs. If you look at the very first tag on the page: <HTML>, you will see that there is a matching </HTML> tag at the very bottom of the file.

Figure 1 explains the meaning of each of the tags in the file. This is just a small sampling of HTML tags; there are dozens more. However, if you can understand the tags used here, you will understand 80-90% of the tags you will find in more complicated files.

➢ CREATING HTML OUTPUT USING SAS VERSION 6

To create HTML output from your SAS output, all you have to do is download and install the SAS HTML formatting tools. Once you've done that, creating the output is easy.

What you do is write your usual SAS code to produce the output, and then add two macro calls. One goes before the procedure that produces the output and the other goes after.

For example, to output a PROC PRINT to HTML, the code would look like this:

```
%OUT2HTM(CAPTURE=ON, RUNMODE=B);
    TITLE 'Cuppa Coffee Co.';
    TITLE2 'Daily Sales Totals';
    PROC PRINT DATA=TEMP NOOBS;
    RUN;
%OUT2HTM(CAPTURE=OFF, RUNMODE=B,
        OPENMODE=REPLACE,
        HTMLFILE=myfile.html);
```

OUT2HTM is the macro that does the conversion from SAS output to HTML. It has dozens of parameters, but you only need to use a few to get the job done. The CAPTURE=ON and CAPTURE=OFF tell the macro to start and stop capturing output. RUNMODE=B indicates that the job will be run in batch mode. OPENMODE=REPLACE causes the output HTML file to be replaced each time the code is rerun. Finally, HTMLFILE=myfile.html tells SAS where to put your HTML output.

The file created by OUT2HTM will be very similar to the code presented in Figure 1. The macro will add a few extra informational tags at the top of the file, but they will not affect the way the page appears when viewed in a browser.

In addition, HTML files created by SAS and other programs will not have their contents as neatly lined up as the code in Figure 1. Text and tags will sprawl from line to line in the output. There is no requirement in HTML that the code follows any standard format. You could put the entire contents of your HTML file on a single line if you wanted to.

➢ CREATING HTML OUTPUT USING SAS VERSION 7

The only difference in how you produce HTML files in version 7 is that you put in a different call before and after the SAS code. Instead of calling one of the HTML formatting macros, you call the Output Delivery System.

The following code shows how it works:

```
ODS HTML BODY='myfile.html';
    TITLE 'Cuppa Coffee Co.';
    TITLE2 'Daily Sales Totals';
    PROC PRINT DATA=TEMP NOOBS;
    RUN;
ODS HTML CLOSE;
```

The output produced by this code will be virtually the same as the output produced by the previous code. The only differences that you will see is that version 6 output is black and white by default, and version 7 output uses more color.

However, with both OUT2HTM and ODS, there are dozens of parameters that you can use to control the look of your output. Version 7 offers more flexibility and control, but you can produce very attractive output in either version. The following examples will show a few tricks you can use to customize your output.

➢ TRICK #1: ADDING A HYPERLINK TO YOUR FOOTNOTE.

Now that you know a little about HTML tags, it's time to turn this knowledge to our advantage. We can use a couple of HTML tags in our FOOTNOTE statement to add a hyperlink to our output. This is a useful trick because it allows you to annotate your output. You can use the footnote to list the source of your data, and then attach a hyperlink so when the viewer clicks on the footnote, it automatically links them to the web site for that source.

To do this, you need to learn a new HTML tag. The tag for a hyperlink is:

```
<A HREF="xxx">text</A>
```

The "xxx" is where you put the information on where you want the hyperlink to go. This needs to be a valid URL (Universal Resource Locator) like "http://www.sas.com" or "http://www.mycompany.org/sales."

The "text" is where you put the text for the hyperlink. This is the text that will be displayed on the page (with an underline to indicate that this is a hyperlink). For example, you could use "SAS Institute" or "1998 Sales Figures."

To see this code in action, we can add the following footnote to the code from our previous examples:[1]

```
FOOTNOTE "Source: <A
  HREF="http//www.cuppacoffee.org">
  Cuppa Coffee Co.</A>";
```

What happens when you run this code is that SAS passes along the HTML tags along with the rest of the text for the footnote. When the file is viewed with a browser, these tags are interpreted, and the footnote is displayed with a hyperlink. This is how the file would look in a browser:

```
Cuppa Coffee Co.
Daily Sales Totals

drink              price        volume

Latte              $2.20        42,655
Cappucino          $2.30        31,985
Mocha              $2.80        24,677

Source: Cuppa Coffee Co.
```

➤ TRICK #2: CHANGING THE TITLES

Another HTML tag that we can use to our advantage is the header tag. This tag identifies the level of the header. Lower levels (<H1> or <H2>) create big, bold titles. Higher levels create smaller titles. By default, OUT2HTM uses the <H3> header for your titles.

---

[1] If you are using version 6 and OUT2HTM, you need to make one other change: add ENCODE=N to the parameters in the second %OUT2HTM call.

However, if we want a bigger title, all we have to do is add header tags to the text in our titles. The following code shows how this is done:

```
TITLE "<H1>Cuppa Coffee Co.</H1>";
TITLE "<H2>Daily Sales Totals</H2>";
```

This code uses the <H1> </H1> tags on the first title for a really big title, and the <H2> </H2> tags for the second title for a slightly smaller title. The results are shown below:

```
Cuppa Coffee Co.

Daily Sales Totals

drink              price        volume

Latte              $2.20        42,655
Cappucino          $2.30        31,985
Mocha              $2.80        24,677
```

➤ TRICK #3: CHANGING THE FONTS

The titles in the previous example are certainly bigger, but they're not very attractive. This is because each web browser has a default font and point size for each heading level. Unfortunately, the fonts are pretty basic.

This example will show how to specify the font and point size for each title, instead of just using the standard heading definitions.

To do this, you need to learn a new pair of HTML tags: <FONT> </FONT>. What these tags do is control the appearance of the text between the two tags. You can specify the typeface, color, and size, among other attributes.

For this example, we're going to put some tags into our titles to select two new typefaces and point size settings. The syntax is:

```
TITLE "<FONT FACE='Comic Sans MS'
    SIZE=5>Cuppa Coffee Co.</FONT>";
TITLE2 "<FONT FACE='Arial'
    SIZE=3>Daily Sales Totals</FONT>";
```

Notice how the two parameters for the FONT tag are used. The FACE= tag is used to specify the typeface. The SIZE= tag is used to specify the font size. A SIZE setting of 1 is the standard font size used for the rest of the text on the screen. By

using two larger settings, we end up with bigger titles. The results are shown below:

➤ TRICK #4: ADDING A LOGO TO THE PAGE

One of the most powerful aspects of the HTML is its ability to display images along with the text on a page. Images are added to the page using the <IMG> tag.

This is a very simple tag to use. All you have to do is specify the image source by listing the file name for the image. Ideally, you should use a web-friendly image format like GIF or JPG for your images.

The following code shows how to display an image in the top left corner of the page by adding an <IMG> tag to the first TITLE statement.

```
TITLE "<IMG SRC='ccclogo.gif'
    HEIGHT=40>";
```

This IMG tag uses parameters to specify the file name to be used and the size to which the image will be expanded or compressed. It is important to specify a size, or you may end up with an image being displayed either too small to see or too large to fit on the screen. Picking the right size usually involves some trial and error.

The following output shows what happens when you add this title to the code from the previous examples:

➤ TRICK #5: ADDING AN IMAGE TO THE BACKGROUND

Each of the previous examples has shown some basic black text on a white background. But if you've spent much time surfing the Web, you probably know that it's possible to use all sorts of colors for the text, and you can use a colorful image for the page background.

This final example will show how you can add a background image and appropriately colored foreground text to your output.

Unlike the previous examples, this time you have to use the OUT2HTM macro to create you output. However, you can still use this technique if you are running version 7. The OUT2HTM macro works for both versions, it's just that most of the time it's easier to use ODS with version 7.

Instead of adding HTML tags to our titles or footnotes, this time we will take advantage of a few of the parameters available for the OUT2HTM macro. We add the following tags to the second OUT2HTM call.

```
%OUT2HTM(CAPTURE=OFF, RUNMODE=B,
    OPENMODE=REPLACE, CENTER=Y,
    BGTYPE=IMAGE, BG='cccback.jpg'
    CTEXT='#DEDACB', SEPTYPE=NONE,
    HTMLFILE=myfile.html);
```

Two of the new parameters (BGTYPE and BG) are used to specify that we will use an image for the page background. Two other parameters (CTEXT and CENTER) set up an appropriate text color to go with the image and center the text so it lines up with the image. Finally, SEPTYPE is used to remove the horizontal divider that OUT2HTM adds by default.

The new output is below. Though it's hard to tell in this black and white format, the background image is a coffee brown shade, and the text is a pale tan.

If you looked at the HTML code that was generated for this output, you would see that OUT2HTM has added two new parameters to the <BODY> tag in the file. The new body tag that controls the background and text colors is as follows.

```
<BODY BACKGROUND='cccback.jpg'
    TEXT='#DEDACB'>
```

Colors can be specified by their names (there's a limited list of legal names), or by their hex codes, as was done here.

Although you didn't have to specify this tag in order to use it, it's still helpful to review the code to see what tags OUT2HTM has used to build your output.

➢ CONCLUSIONS

These examples are just a brief introduction to the world of HTML. You can use these tricks to spice up your HTML output.

However, if you're going to be creating a lot of HTML output from SAS, I strongly recommend that you take the time to learn more about HTML. Although SAS gives you great tools to create HTML output, you will find it a lot easier to use these tools if you can understand the resulting output.

Start with reading through the OUT2HTM and TAB2HTM instructions in more detail. There are dozens of parameters that you can use to shape your HTML output. Alternatively, if you're using ODS, you'll want to read up on ODS styles and templates. These tools will give you a lot of control over your output.

The references section lists several good tutorials and books to help you learn more about HTML.

Finally, the next time you're out surfing the web, take a look at the HTML code behind the pages you're viewing. You can learn a lot by viewing other people's web sites.

Netscape and Internet Explorer both have menu options to view the source code for the page on the screen. You won't recognize all of the tags you see, but the basic tags presented here will show up again and again.

➢ REFERENCES

Free downloadable HTML tutorials:

junior.apk.net/~jbarta/

Free downloadable HTML references:

hot.virtual-pc.com/htmlib/index.html

www.htmlhelp.com/distribution/

Good books for HTML beginners:

Castro, Elizabeth, *HTML 4 For The World Wide Web*, Peachpit Press (www.peachpit.com)

Musciano, Chuck, Bill Kennedy, and Mike Loukides, *HTML: the Definitive Guide*, O'Reilly & Associates (web.ora.com)

Lemay, Laura, *Sams Teach Yourself Web Publishing with HTML 4 in 21 Days*, SAMS (mcp.com)

➢ ACKNOWLEDGEMENTS

SAS is a registered trademark of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

➢ CONTACTING THE AUTHOR

Please direct any questions or feedback to the author at:

Kaiser Permanente Center for Health Research
3800 N. Interstate Ave.
Portland, OR 97227-1098

E-mail: haworthla@chr.mts.kpnw.org