

Chapter 5. How to Make a Web Report

Chapter Table of Contents

Chapter 5. How to Make a Web Report 293

How to Create a Web Report 294

Writing Your Own HTML Tags 296

Experimenting with HTML Tags 297

Customizing the Web Report's Titles 298

Customizing the Web Report's Data Columns 301

Customizing Control Breaks and Grand Totals 303

Putting Graphics in Your Web Report 305

Putting Graphics in Your Report Title 305

Putting Graphics in the Body of Your Report 306

Putting Graphics at Control Breaks 308

Putting Hot Links in your Web Report 308

Using HTML Tables in your Web Report 312

Using Dynamic HTML Tags 315

Using the PRESCRIPT and POSTSCRIPT Options 318

Summary of Options for Web Reports 320

Common HTML Tags 321

Chapter 5. How to Make a Web Report

In earlier chapters we've seen how to create custom reports with Spectrum Writer. In this chapter you will learn how Spectrum Writer can format your reports especially for viewing on Web browsers (such as Microsoft's Internet Explorer and Netscape's Navigator). You can put such reports on your company's Intranet or Internet site for easy company-wide (or public) viewing. Or, send the report to your colleagues as an e-mail attachment that they can automatically view from their e-mail reader. Spectrum Writer is a powerful tool in the move toward paperless "enterprise reporting."

As you will see, you can simply add one statement to any existing Spectrum Writer report to create a Web-viewable version of that report. But that's just the starting point in making attractive Web reports with Spectrum Writer. Advanced users — those familiar with the HTML language — can insert their own HTML commands into a Spectrum Writer report. These commands can take advantage of such Web features as:

- Custom fonts, font sizes, and colors, as well as bold, italic and underlined text.
- Special effects like animation, blinking text or text that scrolls marquee-like across the screen.
- Logos, graphics, charts, and photographs. For example, you could include employee photographs in a personnel directory. (See [Figure 43](#) on page 312 for an example of this.)
- "Hot links" that help viewers navigate within your report — or which let them jump to external Web pages.
- Playing audio or video clips. For example, a viewer could click on a product number in an inventory report and automatically see a video clip demonstrating the product in use!

Note: Before reading the rest of this chapter, you should already be familiar with creating regular Spectrum Writer reports. [Chapter 2, "How to Request a Report"](#) explains this.

How to Create a Web Report

Web pages can be formatted using a language called HTML. In order to properly view your Spectrum Writer report on the Web, certain HTML commands (also called "tags") must be added to the report. Without these HTML tags, the Web browser would "wrap" all of your report lines together into one big, jumbled "paragraph."

The HTML option tells Spectrum Writer to automatically add basic HTML information to your report:

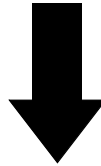
OPTION: HTML

These Control Statements:

```

OPTION: HTML('ABC COMPANY SALES REPORT')
TITLE: 'SALES REPORT'
INPUT: SALES-FILE
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER AMOUNT TAX

```

**Result in this Web Report:**

ABC COMPANY SALES REPORT - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss Real.com Links

SALES REPORT						
REGION	EMPL NAME	SALES DATE	SALES TIME	CUSTOMER	AMOUNT	TAX
SOUTH	JOHNSON	03/12/95	10:25:00	ACE ELECTRICAL	101.38	6.09
WEST	BAKER	03/26/95	12:09:09	JACKS CAFE	137.00	8.22
EAST	MORRISON	03/29/95	15:30:22	STAR MARKET	44.35	2.66
EAST	MORRISON	03/30/95	19:05:41	AI PHOTOGRAPHY	29.65	1.78
EAST	SIMPSON	04/01/95	08:17:57	EUROPEAN DELI	14.99	0.90
NORTH	JOHNSON	04/01/95	17:02:47	VILLA HOTEL	234.45	14.07
NORTH	JOHNSON	04/05/95	14:33:10	MARYS ANTIQUES	9.98	0.60
WEST	BAKER	04/12/95	14:31:12	JACKS CAFE	135.75	8.15
WEST	THOMAS	04/14/95	15:41:38	YOGURT CITY	9.98	0.60
NORTH	JONES	04/15/95	07:58:32	EE GROCERY	10.25	0.62
NORTH	JONES	04/15/95	08:01:59	TOY TOWN	121.76	7.31
NORTH	JONES	04/15/95	13:52:41	TOY TOWN	10.25	0.62
SOUTH	JOHNSON	04/16/95	11:48:33	ACME BUILDING	500.00	30.00
EAST	SIMPSON	04/30/95	15:30:21	J & S LUMBER	23.87	1.43
*** GRAND TOTAL (14 ITEMS)					1,383.66	83.05

Done Local intranet

Figure 35. A basic Web report (viewed on Microsoft's Internet Explorer)

How to Create a Web Report

The above statement tells Spectrum Writer to add HTML tags to your report so that it will display properly on a Web browser. After that, all you do is route Spectrum Writer's output to a file which you can then upload to your Web site or other destination.

Tip: Name your uploaded file using an extension of ".htm" or ".html". That tells Web browsers that your file is an HTML file.

The HTML option also lets you specify a title for your Web page, if you like:

```
OPTION: HTML('ABC COMPANY SALES REPORT')
```

The HTML title is different from the regular report titles. This special HTML title appears at the top of the Web browser's window as the name of your Web page.

Figure 35 shows a Web page created using the above statement.

Writing Your Own HTML Tags

For users who know the HTML language, Spectrum Writer lets you specify your own HTML tags directly within your Web report. By specifying your own HTML tags, you can create very impressive Web reports like the examples shown in the remainder of this chapter.

Note: The rest of this chapter is intended for readers who are familiar with the HTML language.

"HTML tags" are formatting instructions that tell Web browsers how to format a portion of a Web page's text. HTML tags always begin with a "less than" sign and end with a "greater than" sign, like this <...>.

Many HTML tags are used in pairs. Their formatting information applies to all of the text between the opening HTML tag (for example,) and the closing HTML tag (for example,). Closing HTML tags consist of a slash and the first word (or abbreviation) of the opening tag. For example:

```
ABC COMPANY <B>SALES REPORT</B>
```

When the above text is displayed by a Web browser, it will look like this:

```
ABC COMPANY SALES REPORT
```

The words "**SALES REPORT**" are in bold letters. That is because those words are within the HTML "bold" tags. Notice that the HTML tags themselves are not displayed by Web browsers.

Other HTML tags are used by themselves. For example,
 specifies that a line "break" is wanted. It does not require a closing tag.

A list of common HTML tags that you might want to use begins on [page 321](#).

Spectrum Writer's HTML option causes your report to be surrounded with just enough HTML information to make it viewable on the Web. To further customize your report for the Web,

Spectrum Writer lets you insert your own HTML tags (as character literals) directly within your report. Here are some of the places you might want to put your own HTML tags:

- in TITLE statements, to customize the report titles. For example:

```
TITLE: 'ABC COMPANY <B>SALES REPORT</B>'
```

- in COLUMNS statements, to customize one or more columns of data. For example:

```
COLUMNS: REGION EMPL-NAME '<B>' SALES-DATE '</B>' SALES-TIME CUSTOMER AMOUNT TAX
```

- in BREAK statements, to customize the lines printed at control breaks (or the Grand Total lines). For example:

```
BREAK: REGION NOTOTALS  
FOOTING('<B>TOTALS FOR' REGION 36 AMOUNT(TOTAL) TAX(TOTAL) '</B>')
```

- in COMPUTE statements, to create fields that contain dynamically built HTML tags. These fields can then be used in other control statements. For example:

```
COMPUTE: COLOR-TAG = WHEN(AMOUNT > 100) ASSIGN('<FONT COLOR=GREEN>')  
ELSE ASSIGN('<FONT COLOR=RED >')
```

- in PRESCRIPT and POSTSCRIPT options, for formatting information that applies to the entire report. For example:

```
OPTION: PRESCRIPT('<FONT COLOR=BLUE>')  
OPTION: POSTSCRIPT('</FONT>')
```

Experimenting with HTML Tags

Before we continue, a word of caution is in order. Formatting reports with HTML tags is more of an art than a science. For one thing, there are a number of different versions of HTML and it is constantly evolving as new versions of Web browsers are released. In addition, different browsers sometimes process the same HTML tag in slightly different ways. Furthermore, different preference settings in viewers' browsers can cause the same Web report to look different to different viewers. Finally, remember that the actual report that you see on the Web is the result of a two-step process. First Spectrum Writer formats a text-only report that contains your HTML tags mixed in with your report data. Then the Web browser strips the HTML tags out of the formatted report (sometimes throwing off Spectrum Writer's own carefully calculated alignment) and then reformats the report data that remains. All of this is to say that sometimes you will get quite unexpected results the first time you try out an HTML tag! But with enough experimentation, you can usually achieve the desired result.

Tip: While refining a new Web report over and over again, it is easy for HTML syntax errors to creep into the results. There are resources on the Web that perform free syntax-checking of HTML files. This can be very helpful if you are not getting the results you expect from your HTML file. At the time of this writing, one such "HTML validator" service is available at www.w3.org.

The following sections show how to successfully use HTML tags in several specific areas of your reports.

Customizing the Web Report's Titles

You can customize a report title by including HTML tags within the title text in your TITLE statement. For example:

```
TITLE: ' <H1>ABC COMPANY</H1>'
TITLE: ' <H2>RECENT SALES</H2>'
```

Spectrum Writer will simply write out your HTML tags along with the rest of your title text. The Web browser will then recognize the HTML tags and format the text within the tags accordingly.

The <H1> and </H1> HTML tags in the first TITLE statement above tell the Web browser to format the title text within those tags as a "level 1 header." Thus, the person viewing your report on the Web will see "ABC COMPANY" in big, bold text at the top of each report page. They will not see the HTML tags themselves because the Web browser strips them away. Similarly, the second title will be formatted as a "level 2 header" (somewhat smaller than a level 1 header).

Here is another example. In this case, we make the first title red and the second title blue:

```
TITLE: ' <H1><FONT COLOR=RED>ABC COMPANY</FONT></H1>'
TITLE: ' <H2><FONT COLOR=BLUE>RECENT SALES</FONT></H2>'
```

The <Hn> tags cause the titles to be left-justified in some browsers. If you want the titles centered, just enclose them in <CENTER> tags:

```
TITLE: ' <CENTER><H1><FONT COLOR=RED>ABC COMPANY</FONT></H1></CENTER>'
TITLE: ' <CENTER><H2><FONT COLOR=BLUE>RECENT SALES</FONT></H2></CENTER>'
```

Figure 36 shows a report that uses the above TITLE statements.

Notice that the <CENTER> tags cause the titles to be centered within the Web browser window (not necessarily centered just over the report columns). You can always force your titles into any desired position by including padding blanks in the title text:

```
TITLE: ' <H1><FONT COLOR=RED>          ABC COMPANY</FONT></H1>'
TITLE: ' <H2><FONT COLOR=BLUE>      RECENT SALES BY REGION</FONT></H2>'
```

You can use HTML tags in the TITLE statement to change a title's font, size and color and to specify bold, underlined and italicized text. There are also HTML tags to make titles blink or scroll marquee-like. The appropriate HTML tags are listed in the table that begins on [page 321](#).

Following are some additional suggestions for certain situations that may come up when customizing your Web report's titles.

To Avoid Report Titles in the Middle of Screens

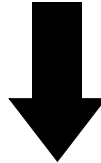
Since a PC screen does not usually display as many report lines as a sheet of paper, it may take several PC screens to show a single "page" of a Spectrum Writer report. This means that the report titles and column headings may seem to appear randomly as viewers "page" through the report online.

These Control Statements:

```

OPTION: HTML('ABC COMPANY -- RECENT SALES')
TITLE:  '<CENTER><H1><FONT COLOR=RED>ABC COMPANY</FONT></H1></CENTER>'
TITLE:  '<CENTER><H2><FONT COLOR=BLUE>RECENT SALES</FONT></H2></CENTER>'
INPUT:  SALES-FILE
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER AMOUNT TAX

```

**Result in this Web Report:**

ABC COMPANY -- RECENT SALES - Netscape

File Edit View Go Communicate Help

ABC COMPANY

RECENT SALES

REGION	EMPL-NAME	SALES-DATE	SALES-TIME	CUSTOMER	AMOUNT	TAX
SOUTH	JOHNSON	03/12/95	10:25:00	ACE ELECTRICAL	101.38	6.09
WEST	BAKER	03/26/95	12:09:09	JACKS CAFE	137.00	8.22
EAST	MORRISON	03/29/95	15:30:22	STAR MARKET	44.35	2.66
EAST	MORRISON	03/30/95	19:05:41	AL PHOTOGRAPHY	29.65	1.78
EAST	SIMPSON	04/01/95	08:17:57	EUROPEAN DELI	14.99	0.90
NORTH	JOHNSON	04/01/95	17:02:47	VILLA HOTEL	234.45	14.07
NORTH	JOHNSON	04/05/95	14:33:10	MARYS ANTIQUES	9.98	0.60
WEST	BAKER	04/12/95	14:31:12	JACKS CAFE	135.75	8.15
WEST	THOMAS	04/14/95	15:41:38	YOGURT CITY	9.98	0.60
NORTH	JONES	04/15/95	07:58:32	EE GROCERY	10.25	0.62
NORTH	JONES	04/15/95	08:01:59	TOY TOMY	121.76	7.31
NORTH	JONES	04/15/95	13:52:41	TOY TOMY	10.25	0.62
SOUTH	JOHNSON	04/16/95	11:48:33	ACME BUILDING	500.00	30.00
EAST	SIMPSON	04/30/95	15:30:21	J & S LUMBER	23.87	1.43
*** GRAND TOTAL (14 ITEMS)					1,383.66	83.05

Document Done

Figure 36. A Web report with customized titles

Customizing the Web Report's Titles

Use the TITLEONCE option to just print the report titles and column headings once at the beginning of your report. This solves the problem of random titles and saves valuable screen space for actual data. For example:

OPTION: TITLEONCE

Syncing Report Titles with the Web Browser Screen

You may also be able to sync the Web browser window with the report pages by specifying "short" report pages. That would let viewers use their PC's Page Up and Page Down keys to page through your report, one screen per report page. Use Spectrum Writer's PAGELEN option to specify a page length of about 20 lines (experiment to get the exact number for your PC). Also add a blank FOOTNOTE statement to your report. (That forces trailing blank lines at the end of each page to ensure that all pages are exactly the same length.) For example:

OPTION: PAGELEN(20)
FOOTNOTE:

Then adjust the size of your Web browser window as necessary to exactly match the length of your report pages. You can also adjust the font size in your Browser's "preferences" to fit more or fewer lines of the report on each screen.

Note: Be aware that even when this method puts the report pages in sync with *your* PC's Web browser window, it may still scroll differently on other PCs.

To Remove the Underscore Line from the Column Headings

Specify the NOUNDERSCORES option to eliminate the underscore line that appears after the column headings. This often looks better on PC screens and also saves valuable screen space for actual data. For example:

OPTION: NOUNDERSCORES

To Solve Alignment Problems

Remember, you can always use explicit spacing factors to force items into the place you desire. Also, specifying a single slash (/) at the end of the TITLE statement will prevent Spectrum Writer from trying to center- or right-justify your title. For example:

TITLE: '<H2>' 0 #TODAY 5 'SALES REPORT' 4 'PAGE' #PAGENUM '</H2>' /

The numbers in the above statement are spacing factors that tell Spectrum Writer how many spaces to put between each item in the title. The trailing slash (/) tells Spectrum Writer to leave the title left-justified.

Customizing the Web Report's Data Columns

You can put HTML tags (as literals) in the COLUMNS statement to customize individual columns of the report. For example, to make the AMOUNT and TAX columns bold, we could use this statement:

```
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER '<B>' AMOUNT TAX '</B>'
```

The above statement causes the literal text "" to appear in each report line before the AMOUNT column. And the literal text "" will appear as the last item on each line. Of course the Web browser will recognize these special HTML tags and process them as formatting instructions, rather than including them as part of the report shown to the viewer. The viewer won't see these literal texts in the online report, but the AMOUNT and TAX values will now appear in bold letters on the Web page.

The Web report produced by the above COLUMNS statement would have two blank spaces (instead of just one) between the CUSTOMER column and the AMOUNT column. That is because Spectrum Writer defaults to putting one blank space between all report columns. Thus, there would be one blank space after the CUSTOMER column and one blank space after the "" column. The Web browser removes that HTML literal, but not the blank space after it. To solve this problem (which uses up valuable screen space), specify an override spacing factor of 0 as needed in COLUMNS statements that contain HTML literals:

```
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER '<B>' 0 AMOUNT TAX 0 '</B>'
```

Figure 37 uses the above statement to make the AMOUNT and TAX columns bold. Notice in that report that the *column headings* and *total values* for the AMOUNT and TAX fields are also bold. When the HTML (or HTMLAID) option is specified, Spectrum Writer automatically copies your COLUMNS statement HTML tags into the column heading lines and into the total lines for you. Thus, the formatting information that you specify for a data column is also applied to the column headings and total value (if any) for that column.

You can use HTML tags in the COLUMNS statement to change a column's font, size and color and to specify bold, underlined and italicized text. There are also HTML tags to make columns blink or scroll. The appropriate HTML tags are shown in the table that begins on [page 321](#).

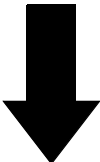
Using Other Fonts in Your Report

The HTML option causes Spectrum Writer to embed your entire report within <PRE> ("preformatted") tags. This tells the Web browser to preserve the report's line breaks and alignment. To accomplish this, the Web browser chooses a non-proportional font for your report. (A non-proportional font is one where all letters have exactly the same width.) If you override this by specifying the name of a proportional font (in a tag in your COLUMNS statement), your report columns will probably not line up correctly. Therefore, it is usually best not to change the font of the body of the report. Or, if you do, be sure to change it to another non-proportional font. (On the other hand, it is generally safe to specify any kind of font for the report titles, since they are not in columnar format.)

Customizing the Web Report's Data Columns

These Control Statements:

```
OPTION: HTML('ABC COMPANY -- RECENT SALES')
TITLE: ' <CENTER><H1><FONT COLOR=RED>ABC COMPANY</FONT></H1></CENTER>'
TITLE: ' <CENTER><H2><FONT COLOR=BLUE>RECENT SALES</FONT></H2></CENTER>'
INPUT: SALES-FILE
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME
          CUSTOMER ' <B>' 0 AMOUNT TAX 0 ' </B>'
```



Result in this Web Report:

REGION	EMPL NAME	SALES DATE	SALES TIME	CUSTOMER	AMOUNT	TAX
SOUTH	JOHNSON	03/12/95	10:23:00	ACE ELECTRICAL	101.38	6.09
WEST	BAKER	03/26/95	12:09:09	JACKS CAFE	137.00	8.22
EAST	MORRISON	03/29/95	15:30:22	STAR MARKET	44.35	2.66
EAST	MORRISON	03/30/95	19:05:41	A1 PHOTOGRAPHY	29.65	1.78
EAST	SIMPSON	04/01/95	08:17:57	EUROPEAN DELI	14.99	0.90
NORTH	JOHNSON	04/01/95	17:02:47	VILLA HOTEL	234.45	14.07
NORTH	JOHNSON	04/05/95	14:33:10	MARYS ANTIQUES	9.98	0.60
WEST	BAKER	04/12/95	14:31:12	JACKS CAFE	135.75	8.15
WEST	THOMAS	04/14/95	15:41:38	YOGURT CITY	9.98	0.60
NORTH	JONES	04/15/95	07:58:32	EE GROCERY	10.25	0.62
NORTH	JONES	04/15/95	08:01:59	TOY TOMM	121.76	7.31
NORTH	JONES	04/15/95	13:52:41	TOY TOMM	10.25	0.62
SOUTH	JOHNSON	04/16/95	11:48:33	ACME BUILDING	500.00	30.00
EAST	SIMPSON	04/30/95	15:30:21	J & S LUMBER	23.87	1.43
*** GRAND TOTAL (14 ITEMS)					1,383.66	83.05

Figure 37. A Web report with two bold columns

Customizing Control Breaks and Grand Totals

You can also put your own HTML tags in the lines printed at control breaks (including the Grand Total "control break"). For example, to make the whole total line at a control break appear in bold, underlined, italicized text, you could specify:

```
BREAK: REGION NOTOTALS  
FOOTING(' <B><U><I>TOTALS FOR' REGION 36 AMOUNT(TOTAL) TAX(TOTAL) ' </I></U></B>')
```

The report in [Figure 38](#) uses the above BREAK statement.

Notice that we did not use the TOTAL parm to customize the total line text (as you would for a normal report). Instead, we specified NOTOTALS (to suppress the default total line) and then added a FOOTING parm to create our own total line. There are two reasons for this.

First, while the TOTAL parm would allow us to specify our *opening* HTML tags, it offers no way to specify the *closing* tags, since these must come after the totalled numeric columns.

Second, putting HTML tags in the TOTAL parm throws off the alignment of the totalled columns. The Web browser strips the HTML tags from the first part of the total line, which causes the totalled numeric columns to be shifted left.

By using a FOOTING parm (as in the statement above), you can specify both opening and closing HTML tags. You can also use an explicit spacing factor (the "36") to force the totalled columns into their proper location. Experimentation is the best way to determine the correct spacing factor for a particular report.

Tip: When experimenting, use the MAXINCLUDE option to limit the number of records included in your report. This can greatly speed up your trial runs, especially if you're working with large input files.

HTML tags can also be used in the BREAK statement's HEADING parm. See [Figure 40](#) (page 309) for an example of this.

You can use HTML tags in the BREAK statement to change a total line's font, size and color and to specify bold, underlined and italicized text. There are also HTML tags to make text blink or scroll. The appropriate HTML tags are shown in the table that begins on [page 321](#).

Following are some additional suggestions for certain situations that may come up when customizing your Web report's control breaks.

Putting a Divider Line at Control Breaks

You may want to use the <HR> tag to put a "horizontal rule" (line) in your report. For example, to add lines to a report to separate the regions, you could specify:

```
BREAK: REGION FOOTING(' <HR>')
```

The report in [Figure 41](#) (page 310) has such a line at the control breaks.

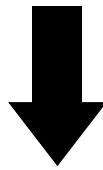
Why is the Total Line Split into Two Lines?

Including HTML tags in the COLUMNS statement sometimes causes the default total line to be split into two lines. This is because all HTML literals from the COLUMNS statement are also copied into the total lines. (This insures that if a report column is bold, for example,

Customizing Control Breaks and Grand Totals

These Control Statements:

```
OPTION: HTML('ABC COMPANY -- RECENT SALES BY REGION')
TITLE: '<CENTER><H1><FONT COLOR=RED>ABC COMPANY</FONT></H1></CENTER>'
TITLE: '<CENTER><H2><FONT COLOR=BLUE>RECENT SALES BY REGION' 0
      '</FONT></H2></CENTER>'
INPUT:  SALES-FILE
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME
          CUSTOMER '<B>' 0 AMOUNT TAX 0 '</B>'
SORT:   REGION
BREAK:  REGION NOTOTALS
FOOTING(' <B><U><I>TOTALS FOR' REGION 36 AMOUNT(TOTAL) TAX(TOTAL)
        '</I></U></B>')
```



Result in this Web Report:

The screenshot shows a web browser window with the title 'ABC COMPANY -- RECENT SALES BY REGION - Netscape'. The report content is as follows:

ABC COMPANY						
RECENT SALES BY REGION						
REGION	EMPL NAME	SALES DATE	SALES TIME	CUSTOMER	AMOUNT	TAX
EAST	SIMPSON	04/30/95	15:30:21	J & S LUMBER	23.07	1.43
EAST	SIMPSON	04/01/95	08:17:57	EUROPEAN DELI	14.99	0.90
EAST	MORRISON	03/30/95	19:03:41	AL PHOTOGRAPHY	29.65	1.78
EAST	MORRISON	03/29/95	15:30:22	STAR MARKET	44.35	2.66
TOTALS FOR EAST					112.06	6.77
NORTH	JONES	04/15/95	13:52:41	TOY TOWN	10.25	0.62
NORTH	JONES	04/15/95	08:01:59	TOY TOWN	121.76	7.31
NORTH	JONES	04/15/95	07:58:32	ES GROCERY	10.25	0.62
NORTH	JOHNSON	04/05/95	14:33:10	MARYS ANTIQUES	9.98	0.60
NORTH	JOHNSON	04/01/95	17:02:47	VILLA HOTEL	234.45	14.07
TOTALS FOR NORTH					386.69	23.22

Figure 38. A Web report with customized total lines

the totals for that column will also be bold.) The problem occurs when there is not enough room to put the entire total line text before the first HTML tag. Spectrum Writer then puts the total line text on a separate line so that the HTML tags can appear in the correct location on the total line. If you want to force the totals onto a single line, specify the NOTOTALS option on your BREAK statement. Then use a FOOTING parm to specify exactly how your total line should look. The report in [Figure 38](#) demonstrates this.

Putting Graphics in Your Web Report

Use the ("image") tag to display a graphic image in your Web report. The graphic might be a logo, a drawing, a photograph, a chart, a graph, etc. When you include an tag in your report output, the Web browser will insert the image into the Web page right at that point.

The format of the tag is:

```
<IMG SRC="url" [WIDTH=nnn] [ALIGN=TOP/MIDDLE/BOTTOM]>
```

The "url" identifies the source file that contains your graphic image. The graphic image file will usually be in GIF or JPEG format and will be named with an extension of ".gif" or ".jpg". The optional WIDTH parm specifies how wide (in pixels) the image should be. The optional ALIGN parm tells the Web browser how to vertically align any text that follows the image. The tag also has other optional parms not shown above.

Putting Graphics in Your Report Title

Put an tag in your TITLE statement to include a graphic among your titles. For example, to include a corporate logo along with a title at the top of each report page, we could use these statements:

```
TITLE: ' <CENTER><IMG SRC="LOGO.GIF" WIDTH=150></CENTER>'
TITLE: ' <H2><CENTER>SALES REPORT</CENTER></H2>'
```

Of course, for this to work your Web site must have a file named LOGO.GIF that contains your company's logo.

[Figure 39](#) (page 307) uses TITLE statements similar to the ones shown above. (So do the reports on [page 312](#) and [page 317](#).)

You can also put an tag in the same TITLE statement as your title text. For example:

```
TITLE: ' <IMG SRC="LOGO.GIF" ALIGN=MIDDLE WIDTH=50>SALES REPORT'
```

Notice the ALIGN=MIDDLE parm in the above tag. It tells the Web browser to align the subsequent text with the middle (height wise) of the graphic.

A more powerful (but more complicated) way of combining text and graphics in the title is to use an HTML table. That technique is discussed under ["Using HTML Tables in your Web Report"](#) (page 312).

Putting Graphics in the Body of Your Report

You can also put tags in your COLUMNS statement to include a graphic in each detail line of the report. In other words, your report can have one or more columns of graphics, instead of text. In this case, of course, you do not want to show the *same* graphic in every detail line. You want a graphic that is related to the data in the rest of that particular report line. Use a COMPUTE statement to dynamically build an appropriate tag for each report line.

For example, assume that we want to produce a personnel directory for the Web. We want the directory to include each employee's photograph. Further assume that our Web site has photographs of the employee's stored in JPG files. The photograph files are named EMPnnn.JPG, where nnn is the employee's employee number. We would use the following COMPUTE statement to build the correct tag for the report detail lines:

```
COMPUTE: PICTURE-TAG = '<IMG SRC="EMP' + EMPL-NUM + '.JPG" WIDTH=60>'
```

The COMPUTE statement above builds a unique tag for each employee, based on their employee number. For example, the PICTURE-TAG field for employee number 044 will contain this text: .

We can then use this PICTURE-TAG field in our COLUMNS statement:

```
COLUMNS: PICTURE-TAG LAST-NAME FIRST-NAME EMPL-NUM HIRE-DATE
```

The first column of the resulting Web report is a column of employee pictures. [Figure 39](#) uses the above statements.

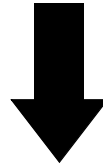
Notice how we handled the column headings in [Figure 39](#). We specified NOCOLHDGS to suppress the default column headings. Then we used extra TITLE statements to build our own column headings. (We also used an empty TITLE statement to put a blank line between the report titles and the column headings.)

Why didn't we just use Spectrum Writer's default column headings? Because adding a "picture" column to a report tends to throw off the alignment of all of the following column headings (and of the total values, as well). Remember that when Spectrum Writer formats its report, the "picture" column just contains text (the tag) like any other column. Later, the Web browser strips away these tags in the detail lines and replaces them with actual images. Those images (photographs, in this example) take up a different amount of space than the tag took up. Since no similar substitution takes place in the column heading lines (or in the total lines) their spacing no longer matches the detail line spacing.

When including graphics in the body of your report, you will probably want to use this technique to control exactly how your column headings look. Similarly, use the BREAK statement FOOTING parm to precisely control how your total lines look (as we also did in [Figure 38](#) on page 304).

These Control Statements:

```
OPTION: HTML('ABC COMPANY PERSONNEL DIRECTORY')
OPTION: NOCOLHDGS
INPUT: EMPL-FILE
TITLE: '<CENTER><IMG SRC="ABCLGO02.GIF"></CENTER>'
TITLE: '<CENTER><FONT FACE="LUCIDA HANDWRITING" SIZE=5>' 0
      'ONLINE PERSONNEL DIRECTORY</FONT></CENTER>'
TITLE:
TITLE: '
TITLE: ' PHOTO      LAST NAME      FIRST NAME      EMPL   HIRE' /
      ' CITY              ST ZIP' /
COMPUTE: PICTURE-TAG='<IMG SRC="EMP' + EMPL-NUM + '.JPG" WIDTH=60>'
COLUMNS: PICTURE-TAG LAST-NAME FIRST-NAME EMPL-NUM HIRE-DATE
          CITY STATE ZIP
SORT:     LAST-NAME
```



Result in this Lotus 1-2-3 Spreadsheet:

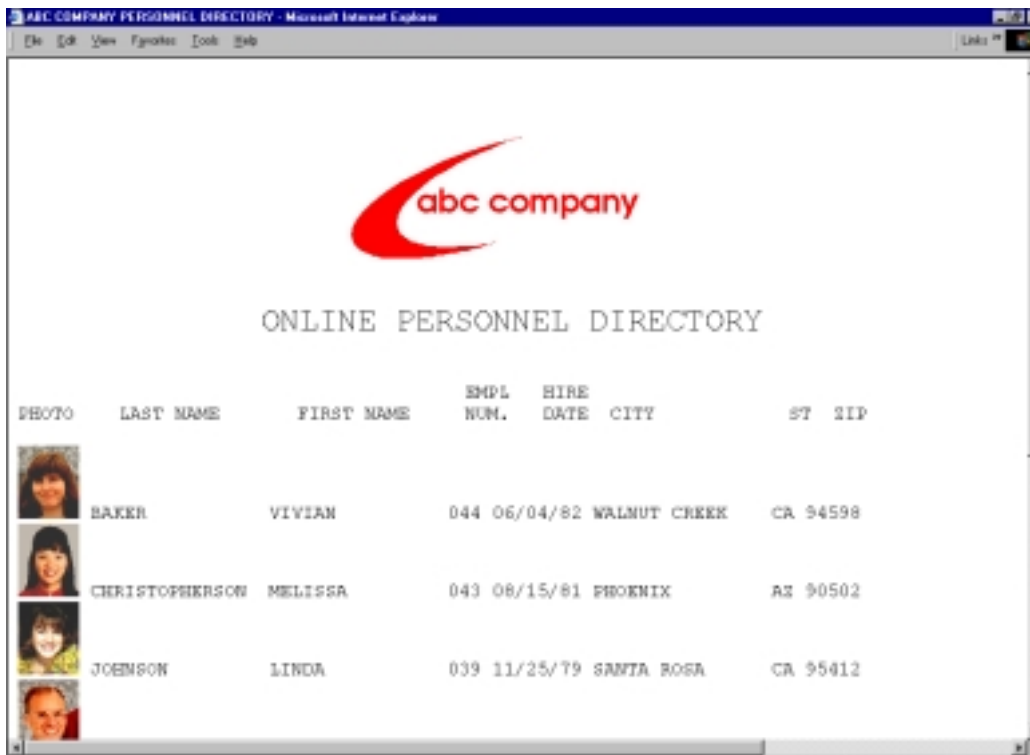


Figure 39. A Web report containing graphics in the title and body

Putting Graphics at Control Breaks

You can also use tags in the BREAK statement, to place graphics at your report's control breaks. The report in [Figure 41](#) (page 310) shows an example of this.

Putting Hot Links in your Web Report

Hot links (or "hypertext links") are areas of your Web report that viewers can click on to obtain some action. For example, clicking on a hot link might cause the browser to jump to a different part of the report (the Grand Totals, for example). Or it might jump to a different Web page altogether (for example, to a Web page describing a particular product in a sales report).

Hot links can also invoke an audio player or even a video clip player. Viewers could click on a product name in your report, for example, and automatically be shown a video demonstrating that product in use.

To create a hot link in your report, surround your "hot text" (that is, the text that the viewer should click on) with the and ("anchor") tags. The url in the tag will determine what action is taken when the viewer clicks on the text.

If the url is the name of a Web page (for example, an HTML file), the browser will jump to that Web page when the viewer clicks on the hot link:

```
TITLE: ' <A HREF="WWW.ABC.COM/INDEX.HTML">' 0
      ' CLICK HERE TO READ MORE ABOUT THE ABC COMPANY</A>'
```

If the url is the name of a label within an HTML file (even within the Web report itself) the browser will jump to that specific location within that Web page:

```
TITLE: ' <A HREF=#EAST>CLICK HERE TO JUMP TO EAST REGION</A>'
```

If the url is the name of an audio file (such as a .WAV, .AU or .MID file), the browser will play that sound file when the viewer clicks the hot link:

```
TITLE: ' <A HREF="WELCOME.WAV">' 0
      ' CLICK HERE TO HEAR THE CHAIRMAN' 'S GREETING</A>'
```

If the url is the name of a video clip (such as an .AVI file) the browser will play that video clip when the viewer clicks the hot link.

```
TITLE: ' <A HREF="PRODUCT.AVI">' 0
      ' CLICK HERE TO SEE A VIDEO ABOUT ABC COMPANY' 'S PRODUCTS</A>'
```

As mentioned, hot links can be used to let viewers easily move around within the report itself. The report in [Figure 41](#) (page 310) shows an example of using hot links in this way. Viewers can click on one of the hot texts at the beginning of the report to go directly to the region they are interested in, or to the Grand Totals. And at the bottom of the report, there is another hot link to take them back to the top of the report.


```

OPTION: HTML('ABC COMPANY -- SALES BY REGION')
NOUNDERSCORES
PRESCRIPT(' <A NAME=TOP>')
PRESCRIPT(' <A HREF=#EAST>CLICK HERE TO JUMP TO EAST REGION</A>')
PRESCRIPT(' <A HREF=#NORTH>CLICK HERE TO JUMP TO NORTH REGION</A>')
PRESCRIPT(' <A HREF=#SOUTH>CLICK HERE TO JUMP TO SOUTH REGION</A>')
PRESCRIPT(' <A HREF=#WEST>CLICK HERE TO JUMP TO WEST REGION</A>')
PRESCRIPT(' <A HREF=#GRAND>CLICK HERE TO JUMP TO GRAND TOTALS</A>')
POSTSCRIPT(' <A HREF=#TOP>CLICK HERE TO RETURN TO TOP OF REPORT</A>')
TITLE: ' <H1><CENTER>ABC COMPANY SALES BY REGION</CENTER></H1>'
INPUT: SALES-FILE
COMPUTE: REGION-IMG = ' <IMG SRC=REG' + #LEFT(REGION,1) +
                    ' .GIF ALIGN=MIDDLE>'
COLUMNS: REGION(NOREPEAT) EMPL-NAME SALES-DATE SALES-TIME
          CUSTOMER AMOUNT TAX
SORT:     REGION
BREAK:    REGION NOTOTALS
HEADING(' <A NAME=' 0 REGION 0 '>')
HEADING(REGION-IMG)
HEADING(' ')
FOOTING(' ')
FOOTING(' <FONT SIZE=5><B><I> ' 0
        ' TOTALS FOR' REGION ' REGION' 7 AMOUNT(TOTAL) TAX(7,TOTAL)
        ' </I></B></FONT>')
FOOTING(' <HR>')
BREAK: #GRAND
FOOTING(' <A NAME=GRAND>')

```

Figure 40. Control statements used to create a Web report with “hot links”

Figure 40 shows the control statements used to create the report in **Figure 41**. As you can see in **Figure 40**, we used PRESCRIPT options to put five hot link texts ahead of the report. (The PRESCRIPT option puts lines of text before the beginning of the report itself.) The urls in these links are HTML “labels” within our own report named EAST, NORTH, SOUTH, WEST and GRAND. We used a POSTSCRIPT option to write one other hot link text at the end of the report. It references a label called TOP.

To make these links work, the output must contain the six labels (TOP, EAST, NORTH, etc.) at the appropriate place within the report. HTML labels are assigned with tags. We used another PRESCRIPT option to put the TOP label at the top of the report. And we used BREAK statement HEADING parms to put the four regional labels at the beginning of their respective control groups. Finally we used the Grand Total control break FOOTING parm to supply the GRAND label at the Grand Total lines.

Figure 42 (page 311) shows the actual HTML output file created by Spectrum Writer.

Note: We were able to provide hot links to each control group in this report because we knew ahead of time all the values that the REGION variable would have. This particular technique can only be used when you know the values of your control break variable in advance.

Putting Hot Links in your Web Report

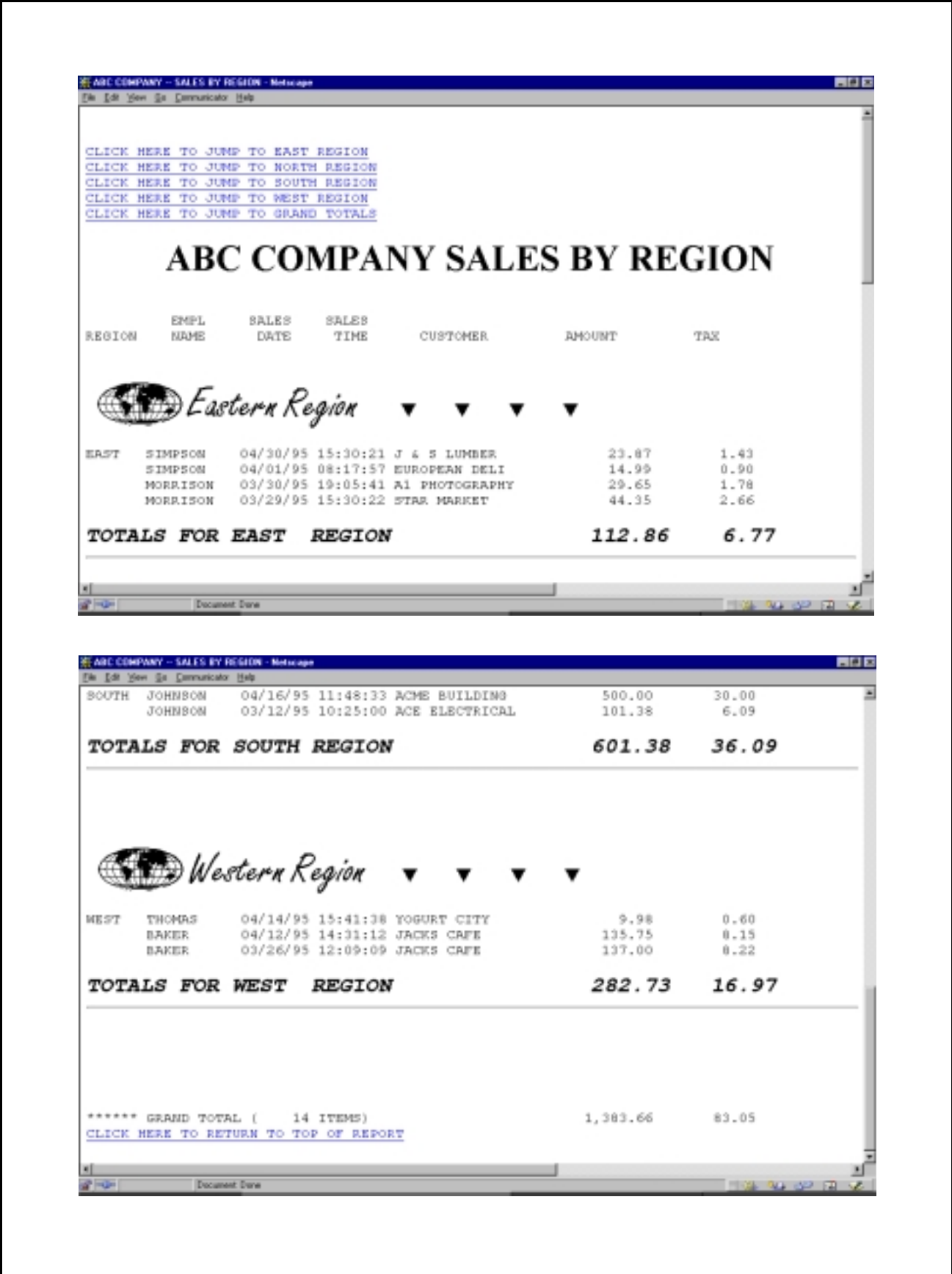


Figure 41. Two screens from a Web report with “hot links”

```

<HTML>
<HEAD>
<TITLE>ABC COMPANY -- SALES BY REGION</TITLE>
</HEAD>
<BODY>
<PRE>
<A NAME=TOP>
<A HREF=#EAST>CLICK HERE TO JUMP TO EAST REGION</A>
<A HREF=#NORTH>CLICK HERE TO JUMP TO NORTH REGION</A>
<A HREF=#SOUTH>CLICK HERE TO JUMP TO SOUTH REGION</A>
<A HREF=#WEST>CLICK HERE TO JUMP TO WEST REGION</A>
<A HREF=#GRAND>CLICK HERE TO JUMP TO GRAND TOTALS</A>
      <H1><CENTER>ABC COMPANY SALES BY REGION</CENTER></H1>

      EMPL      SALES      SALES
REGION  NAME      DATE      TIME      CUSTOMER      AMOUNT      TAX

<A NAME=EAST >
<IMG SRC=REGE.GIF ALIGN=MIDDLE>

EAST  SIMPSON    04/30/95 15:30:21 J & S LUMBER      23.87      1.43
      SIMPSON    04/01/95 08:17:57 EUROPEAN DELI      14.99      0.90
      MORRISON   03/30/95 19:05:41 A1 PHOTOGRAPHY      29.65      1.78
      MORRISON   03/29/95 15:30:22 STAR MARKET      44.35      2.66

<FONT SIZE=5><B><I>TOTALS FOR EAST  REGION      112.86      6.77 </I></B></FONT>
<HR>

<A NAME=NORTH>
<IMG SRC=REGN.GIF ALIGN=MIDDLE>

NORTH JONES      04/15/95 13:52:41 TOY TOWN      10.25      0.62
      JONES      04/15/95 08:01:59 TOY TOWN      121.76      7.31
      JONES      04/15/95 07:58:32 EZ GROCERY      10.25      0.62
      JOHNSON    04/05/95 14:33:10 MARYS ANTIQUES      9.98      0.60
      JOHNSON    04/01/95 17:02:47 VILLA HOTEL      234.45      14.07

<FONT SIZE=5><B><I>TOTALS FOR NORTH REGION      386.69      23.22 </I></B></FONT>
<HR>

      (additional lines not shown)

<A NAME=WEST >
<IMG SRC=REGW.GIF ALIGN=MIDDLE>

WEST  THOMAS      04/14/95 15:41:38 YOGURT CITY      9.98      0.60
      BAKER      04/12/95 14:31:12 JACKS CAFE      135.75      8.15
      BAKER      03/26/95 12:09:09 JACKS CAFE      137.00      8.22

<FONT SIZE=5><B><I>TOTALS FOR WEST  REGION      282.73      16.97 </I></B></FONT>
<HR>

<A NAME=GRAND>
***** GRAND TOTAL (    14 ITEMS)      1,383.66      83.05
<A HREF=#TOP>CLICK HERE TO RETURN TO TOP OF REPORT</A>
</PRE>
</BODY>
</HTML>

```

Figure 42. the HTML output file for a Web report with “hot links”

Using HTML Tables in your Web Report

If you like, you can use HTML tags to format your Web report as a table. HTML tables are often used as a tool for organizing data on a Web page. One common use of tables is to align multiple lines of text with a single image.

HTML tables consist of a number of rows. Within each row is one or more data cells (which form the columns). Each data cell can contain report text and/or an image.

Surround your HTML table with `<TABLE>` and `</TABLE>` tags. Use the `<TR>` tag to start a new table row. Use the `<TD>` tag to start each new data cell within a row.

The Web report in [Figure 43](#) incorporates two separate HTML tables. ([Figure 44](#) (page 313) shows the control statements used to create this report. [Figure 45](#) (page 314) shows the actual HTML output file created by Spectrum Writer for this report.)

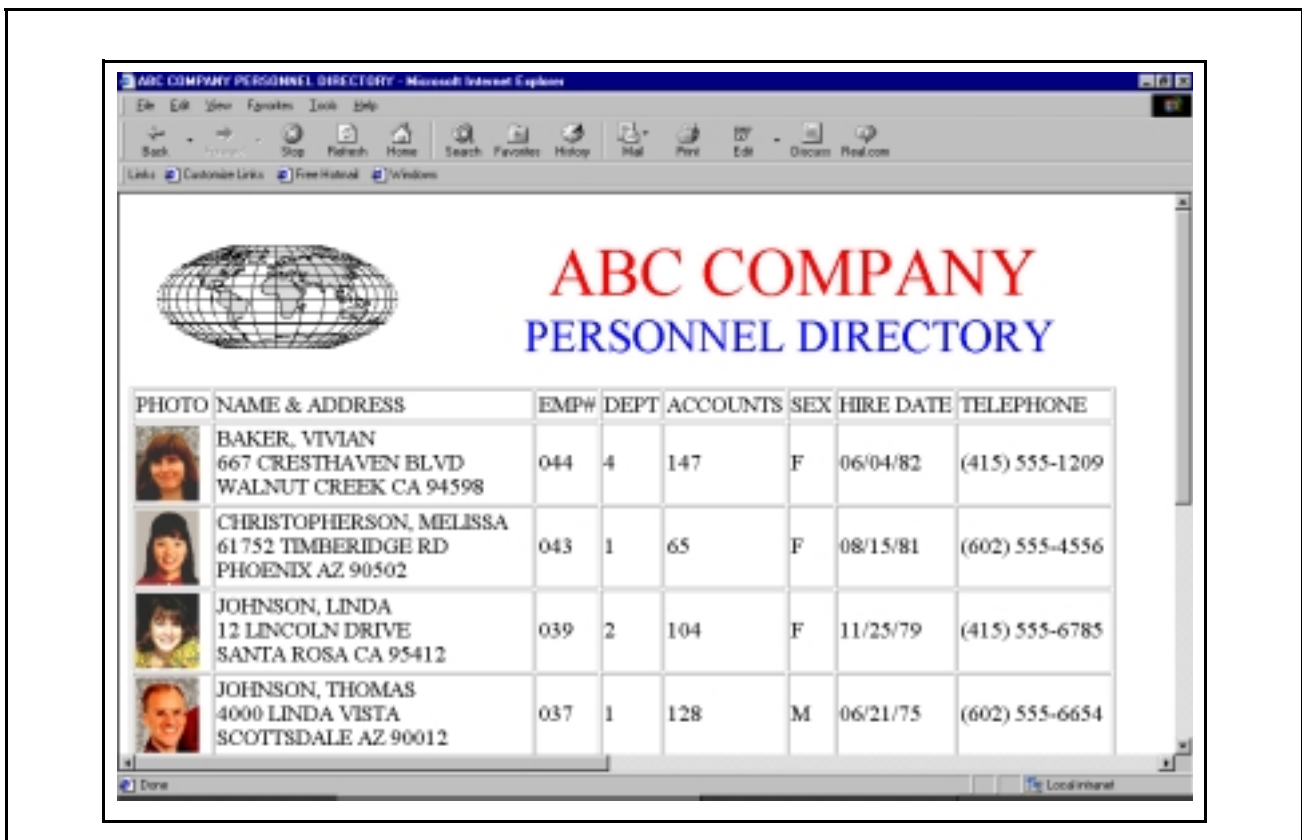


Figure 43. A Web report that uses “tables”

The first table contains only the title information. (This table was drawn without borders, so it doesn't look like a typical table.) We used the table as a convenient way to align multiple lines of text with the logo graphic. This table has just one row. That row contains two cells. The first cell contains the logo image. The second cell contains the two lines of title text. Here are the statements we used to build this table:

```
PRESRIPT(' <TABLE><TR><TD><IMG SRC="ABC.GIF" WIDTH=275 ALIGN=MIDDLE>')
PRESRIPT(' <TD ALIGN=CENTER><FONT COLOR=RED SIZE=7 FACE="CUPERTINO">')
PRESRIPT(' ABC COMPANY</FONT></TD></TR></TABLE>')
```

```
PRESCTPT(' <FONT COLOR=BLUE SIZE=6 FACE=BASSOON>PERSONNEL DIRECTORY' )
PRESCTPT(' </FONT></TABLE>' )
```

Notice that we used PRESCTPT options to write out this HTML code once at the beginning of the report. That means that our title will only appear once at the top of the whole report. By specifying the NOTITLES option (see [Figure 44](#)), we suppressed the regular report titles at the top of each page.

```
OPTION: NOTITLES NOGRANDTOTALS COLSPACE(0)
HTML(' ABC COMPANY PERSONNEL DIRECTORY' )
PRESCTPT(' <TABLE><TR><TD><IMG SRC=WORLD.GIF WIDTH=275 ALIGN=MIDDLE>' )
PRESCTPT(' <TD ALIGN=CENTER><FONT COLOR=RED SIZE=7 FACE="CUPERTINO">' )
PRESCTPT(' ABC COMPANY</FONT>' )
PRESCTPT(' <FONT COLOR=BLUE SIZE=6 FACE=BASSOON>PERSONNEL DIRECTORY' )
PRESCTPT(' </FONT></TABLE>' )
PRESCTPT(' <TABLE BORDER=1 CELLPADDING=2>' )
PRESCTPT(' <TR><TD>PHOTO<TD>NAME & ADDRESS<TD>EMP#<TD>DEPT<TD>ACCOUNTS<
TD>SEX<TD>HIRE DATE<TD>TELEPHONE' )
POSTSCRIPT(' </TABLE>' )
INPUT: EMPL-FILE
COMPUTE: PICTURE-TAG = ' <IMG SRC=EMP' + EMPL-NUM + '.JPG WIDTH=60>'
COMPUTE: NAME = #COMPRESS(LAST-NAME 0 ',' 1 FIRST-NAME)
SORT: NAME
COLUMNS: ' <TR><TD>' PICTURE-TAG
          ' <TD>' NAME ' <BR>' ADDRESS ' <BR>' CITY 1 STATE 1 ZIP
          ' <TD>' EMPL-NUM
          ' <TD>' DEPT-NUM
          ' <TD>' NUM-ACCOUNTS
          ' <TD>' SEX
          ' <TD>' HIRE-DATE
          ' <TD>' TELEPHONE
```

Figure 44. Control statements used to create a Web report with “tables”

In the PRESCTPT statements above, the <TABLE> tag defines the beginning of the table. Since we did not specify a “BORDER=n” parm in this tag, this table has no visible border or grid lines. The first (and only) row in this table is defined with the <TR> tag. The <TD> tag marks the beginning of the first cell in this row. In this cell, we just specified the tag for the company's logo graphic. We then started the second cell with another <TD> tag. Within this cell are two lines of text. We put different tags around each line to make them different colors, fonts and sizes. Finally, we ended the table by specifying the </TABLE> tag.

The second table in our report contains the main body of the report. We also used PRESCTPT options to write the HTML code for the *initial part* of this table:

```
PRESCTPT(' <TABLE BORDER=1 CELLPADDING=2>' )
PRESCTPT(' <TR><TD>PHOTO<TD>NAME & ADDRESS<TD>EMP#<TD>DEPT<TD>ACCOUNTS<
TD>SEX<TD>HIRE DATE<TD>TELEPHONE' )
```

For this table, we chose to show grid lines by specifying BORDER=1 in the <TABLE> tag. The first row of our table, also defined by a PRESCTPT option, contains our table's column headings. (Notice that this PRESCTPT text was too long to fit onto a single line. We continued the literal by typing up to column 72 in the first line, and then continuing in column 2 of the next line.)

```
<HTML>
<HEAD>
<TITLE>ABC COMPANY PERSONNEL DIRECTORY</TITLE>
</HEAD>
<BODY>
<PRE>
<TABLE><TR><TD><IMG SRC=\\WORLD.GIF WIDTH=275 ALIGN=MIDDLE>
<TD ALIGN=CENTER><FONT COLOR=RED SIZE=7 FACE="CURPTINO">
ABC COMPANY</FONT>
<FONT COLOR=BLUE SIZE=6 FACE=BASSOON><PERSONNEL DIRECTORY
</FONT></TABLE>
<TABLE BORDER=1 CELLPADDING=2>
<TR><TD><PHOTO><TD><NAME & ADDRESS><TD><EMP#><TD><DEPT><TD><ACCOUNTS><TD><SEX><TD><HIRE DATE><TD><TELEPHONE
<TR><TD><IMG SRC=EMP044.JPG WIDTH=60><TD><BAKER, VIVIAN
<TR><TD><IMG SRC=EMP043.JPG WIDTH=60><TD><CHRISTOPHERSON, MELISSA
<TR><TD><IMG SRC=EMP039.JPG WIDTH=60><TD><JOHNSON, LINDA
<TR><TD><IMG SRC=EMP037.JPG WIDTH=60><TD><JOHNSON, THOMAS
<TR><TD><IMG SRC=EMP036.JPG WIDTH=60><TD><JONES, JERRY
<TR><TD><IMG SRC=EMP040.JPG WIDTH=60><TD><MACDONALD, RICHARD
<TR><TD><IMG SRC=EMP042.JPG WIDTH=60><TD><MORRISON, MICHAEL
<TR><TD><IMG SRC=EMP041.JPG WIDTH=60><TD><SIMPSON, TIMOTHY
<TR><TD><IMG SRC=EMP045.JPG WIDTH=60><TD><THOMAS, MARTIN
</TABLE>
</PRE>
</BODY>
</HTML>
```

147<TD><F<TD><06/04/82<TD><(415) 555-1209	4<TD>	CA 94598<TD><044<TD>	 <667 CRESTHAVEN BLVD <WALNUT CREEK
65<TD><F<TD><08/15/81<TD><(602) 555-4556	1<TD>	AZ 90502<TD><043<TD>	 <61752 TIMBERIDGE RD <PHOENIX
104<TD><F<TD><11/25/79<TD><(415) 555-6785	2<TD>	CA 95412<TD><039<TD>	 <12 LINCOLN DRIVE <SANTA ROSA
128<TD><M<TD><06/21/75<TD><(602) 555-6654	1<TD>	AZ 90012<TD><037<TD>	 <4000 LINDA VISTA <SCOTTSDALE
78<TD><M<TD><01/31/80<TD><(415) 555-7653	2<TD>	CA 94012<TD><036<TD>	 <125 MAIN STREET <SAN FRANCISCO
6<TD><M<TD><07/04/82<TD><(415) 555-9887	2<TD>	CA 94568<TD><040<TD>	 <525 FOOTHILL DRIVE <PLEASANTON
154<TD><M<TD><11/30/79<TD><(818) 555-4748	3<TD>	CA 91202<TD><042<TD>	 <98 SOUTH LAKESIDE DR <GLENDALE
16<TD><M<TD><12/01/82<TD><(818) 555-1887	3<TD>	CA 91006<TD><041<TD>	 <89876 WEST 53 STREET <ARCADIA
118<TD><M<TD><06/04/82<TD><(415) 555-1152	4<TD>	CA 94519<TD><045<TD>	 <77812 S. HUNTINGTON <CONCORD

Figure 45. HTML file for a Web report with "tables"

We used a POSTSCRIPT option to specify the *closing* `</TABLE>` tag:

```
POSTSCRIPT(' </TABLE>')
```

The POSTSCRIPT text will appear after the entire report. It simply closes the main report table.

Between the PRESCRIPT lines and the POSTSCRIPT line will come the actual report lines specified by the COLUMNS statement:

```
COLUMNS: ' <TR><TD>' PICTURE-TAG  
          ' <TD>' NAME ' <BR>' ADDRESS ' <BR>' CITY 1 STATE 1 ZIP  
          ' <TD>' EMPL-NUM  
          ' <TD>' DEPT-NUM  
          ' <TD>' NUM-ACCOUNTS  
          ' <TD>' SEX  
          ' <TD>' HIRE-DATE  
          ' <TD>' TELEPHONE
```

The COLUMNS statement above begins with a literal containing a `<TR>` tag. That means that each detail line of our report will begin with a `<TR>` tag and become a new table row. We then used `<TD>` tags to put each data item into its own separate cell — with one exception. In the "Name & Address" table column, we included multiple data items in the same cell. We used `
` (break) tags to specify where a new line should be started *within* the cell. Note that we used spacing factors of "1" between the fields within the same cell. That overrides the default spacing factor of "0" that we specified with the `COLSPACE(0)` option (see [page 313](#).) We specified the `COLSPACE(0)` option just to keep from having to type 0's around all of the HTML tag literals in the COLUMNS statement. The PICTURE-TAG field used in the above statement was discussed on [page 306](#).

Using Dynamic HTML Tags

When you put a literal containing an HTML tag in your COLUMNS statement, that tag appears in all of the report detail lines. (It is also copied into the column heading lines and control break total lines.) What if you want to use a different HTML tag for different report lines? You can build dynamic HTML tags to do that.

Dynamic HTML tags are tags whose contents vary, depending on the other data in the report line. Dynamic HTML tags are assigned to COMPUTE fields instead of being specified as literals.

We have already seen some examples of dynamic HTML tags. The reports on [page 307](#) and [page 312](#) built dynamic `` tags for each employee's photograph. And on [page 310](#) we built dynamic `` tags to display regional logos at the beginning of each region's data.

Now let's look at another use of dynamic HTML tags. Assume that we want our Web report to show sales amounts that are over \$100 in green and all other sales amounts in red. Instead of putting a literal `` tag in our COLUMNS statement, we would use these statements to compute a dynamic `` tag:

```
COMPUTE: COLOR-TAG = WHEN(AMOUNT > 100) ASSIGN(' <FONT COLOR=GREEN>')  
          ELSE                               ASSIGN(' <FONT COLOR=RED >')
```

Using Dynamic HTML Tags

Notice that we padded the "red" tag with enough blanks to make it the same size as the "green" tag. This is important to keep the resulting report properly aligned. Remember that the Web browser strips all HTML tags from the report before displaying it to the viewer. To preserve column alignment, the same number of bytes must be stripped from *all* report lines.

In our COLUMNS statement, we now use this COLOR-TAG field instead of a literal tag:

```
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER  
          COLOR-TAG(' <!--xxxxxxxxxxx-->' ) 0 AMOUNT TAX 0 ' </FONT>' (' <!---->')
```

In the report detail lines, either "" or "" will now appear before the AMOUNT field. (You can see this on [page 318](#).) In both cases, the same closing tag ("") appears after the TAX column.

The report in [Figure 46](#) uses the above statements.

Notice the unusual column heading for the COLOR-TAG column. Tags that begin with "<!--" and end with "-->" are considered HTML "comment" tags. You can place any number of other characters between these tags. Comment tags do not affect the formatting of the Web report in any way. However, they are stripped from the Web page like all other HTML tags.

This special column heading was needed to preserve correct alignment of the report. Remember that when you put an HTML tag *literal* in the COLUMNS statement, Spectrum Writer automatically copies the literal tag into all of the column heading lines as well. That ensures that the same amount of HTML text is stripped from the column headings as from the detail lines, which allows the column headings to remain aligned over the data.

However, Spectrum Writer does not copy dynamic HTML tags (that is, tags contained within COMPUTE fields) into the column heading. This means that in the resulting report, the detail lines would have an HTML tag which the column heading lines do not have. When those HTML tags are stripped from the detail lines (but not from the column heading lines) the column headings become skewed.

Thus, for our dynamic COLOR-TAG field we specified our own column heading containing an HTML comment tag. We made sure that the HTML comment in the column heading was the *same length* as the HTML tags appearing in the detail lines. (You can see this by looking at the output on [page 318](#).) As a result, the same number of HTML bytes are removed from the column headings lines and from the detail lines. The resulting report remains aligned in your web browser.

If you look at the actual output on [page 318](#), you will notice something else. Although we only specified one line of override column headings, Spectrum Writer used it for *all three* column heading lines. This is necessary to keep *all* column headings properly aligned. Spectrum Writer does this for you automatically (as long as either the HTML or HTMLAID option is specified).

We also specified an HTML comment as the column heading for the closing literal. Without this override column heading, Spectrum Writer would have copied the literal itself into the column headings. That would have caused an HTML error since there is no corresponding *opening* tag in the column heading lines.

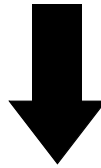
The default Grand Total line would also have been skewed for similar reasons. (It would not contain the dynamic tag, and *would* contain an unmatched

These Control Statements:

```

OPTION: HTML('ABC COMPANY -- COLOR CODED SALES')
TITLE: '<CENTER><IMG SRC="LOGO.GIF"></CENTER>'
TITLE: '<CENTER><H2><I>SALES OVER $100 IN GREEN</I></H2></CENTER>'
INPUT: SALES-FILE
COMPUTE: COLOR-TAG = WHEN(AMOUNT > 100) ASSIGN('<FONT COLOR=GREEN>')
          ELSE ASSIGN('<FONT COLOR=RED >')
COLUMNS: REGION EMPL-NAME SALES-DATE SALES-TIME CUSTOMER
          COLOR-TAG('<!--xxxxxxxxxx-->') 0 AMOUNT TAX 0 '</FONT>'('<!-->')
BREAK: #GRAND NOTOTALS
FOOTING('GRAND TOTALS' 40 AMOUNT(TOTAL) TAX(TOTAL))

```

**Result in this Web Report:**

REGION	EMPL NAME	SALES DATE	SALES TIME	CUSTOMER	AMOUNT	TAX
SOUTH	JOHNSON	03/12/95	10:25:00	ACE ELECTRICAL	101.38	6.09
WEST	BAKER	03/26/95	12:09:09	JACKS CAFE	137.00	8.22
EAST	MORRISON	03/29/95	15:30:22	STAR MARKET	44.35	2.66
EAST	MORRISON	03/30/95	19:03:41	AI PHOTOGRAPHY	29.65	1.78
EAST	SIMPSON	04/01/95	08:17:57	EUROPEAN DELI	14.99	0.90
NORTH	JOHNSON	04/01/95	17:02:47	VILLA HOTEL	234.45	14.07
NORTH	JOHNSON	04/05/95	14:33:10	MARYS ANTIQUES	9.98	0.60
WEST	BAKER	04/12/95	14:31:12	JACKS CAFE	135.75	8.15
WEST	THOMAS	04/14/95	15:41:38	YOGURT CITY	9.98	0.60
NORTH	JONES	04/15/95	07:58:32	EE GROCERY	10.25	0.62
NORTH	JONES	04/15/95	08:01:59	TOY TOWN	121.76	7.31
SOUTH	TOYNE	04/18/95	12:53:23	BOY TOWN	10.38	0.63
GRAND TOTALS					40	

Figure 46. A Web report that uses dynamic HTML tags

Using Dynamic HTML Tags

```
<HTML>
<HEAD>
<TITLE>ABC COMPANY -- COLOR CODED SALES</TITLE>
</HEAD>
<BODY>
<PRE>
                <CENTER><IMG SRC="ABCLOGO.GIF"></CENTER>
                <CENTER><H2><I>SALES OVER $100 IN GREEN</I></H2></CENTER>

REGION      EMPL      SALES      SALES      CUSTOMER      <!--xxxxxxxxxx-->      AMOUNT      TAX      <!-->
            NAME      DATE      TIME
-----
SOUTH  JOHNSON    03/12/95  10:25:00  ACE ELECTRICAL  <FONT COLOR=GREEN>      101.38      6.09</FONT>
WEST   BAKER     03/26/95  12:09:09  JACKS CAFE      <FONT COLOR=GREEN>      137.00      8.22</FONT>
EAST   MORRISON   03/29/95  15:30:22  STAR MARKET    <FONT COLOR=RED  >      44.35      2.66</FONT>
EAST   MORRISON   03/30/95  19:05:41  A1 PHOTOGRAPHY  <FONT COLOR=RED  >      29.65      1.78</FONT>
EAST   SIMPSON    04/01/95  08:17:57  EUROPEAN DELI   <FONT COLOR=RED  >      14.99      0.90</FONT>
NORTH  JOHNSON    04/01/95  17:02:47  VILLA HOTEL     <FONT COLOR=GREEN>      234.45      14.07</FONT>
NORTH  JOHNSON    04/05/95  14:33:10  MARYS ANTIQUES  <FONT COLOR=RED  >      9.98      0.60</FONT>
WEST   BAKER     04/12/95  14:31:12  JACKS CAFE      <FONT COLOR=GREEN>      135.75      8.15</FONT>
WEST   THOMAS    04/14/95  15:41:38  YOGURT CITY     <FONT COLOR=RED  >      9.98      0.60</FONT>
NORTH  JONES      04/15/95  07:58:32  EZ GROCERY      <FONT COLOR=RED  >      10.25      0.62</FONT>
NORTH  JONES      04/15/95  08:01:59  TOY TOWN        <FONT COLOR=GREEN>      121.76      7.31</FONT>
NORTH  JONES      04/15/95  13:52:41  TOY TOWN        <FONT COLOR=RED  >      10.25      0.62</FONT>
SOUTH  JOHNSON    04/16/95  11:48:33  ACME BUILDING   <FONT COLOR=GREEN>      500.00      30.00</FONT>
EAST   SIMPSON    04/30/95  15:30:21  J & S LUMBER    <FONT COLOR=RED  >      23.87      1.43</FONT>

GRAND TOTALS                                1,383.66      83.05
</PRE>
</BODY>
</HTML>
```

Figure 47. HTML file with dynamic HTML tags

 tag.) Therefore, we used a BREAK statement to suppress the default Grand Totals. Then we used a FOOTING parm to specify precisely how the Grand Total line should look.

There is also another technique you can use to align the column headings in reports that have dynamic HTML tags. That is to suppress the default column headings by specifying the NOCOLHDGS option. Then use TITLE statements to specify your own column headings. The report in Figure 39 (page 307) illustrates this.

Using the PRESCRIPT and POSTSCRIPT Options

The PRESCRIPT option is used to write one or more lines of text before the beginning of the report. The POSTSCRIPT option is used to write lines of text after the end of the report.

In Figure 41 (page 310), we used these options to put hot link texts and HTML labels at the beginning and end of the report.

Using the PRESCRIPT and POSTSCRIPT Options

Another use of the PRESCRIPT and POSTSCRIPT options is to provide formatting tags that apply to your entire report. For example:

```
OPTION: HTML
OPTION: PRESCRIPT(' <FONT COLOR=RED>')
OPTION: POSTSCRIPT(' </FONT>')
```

The first statement above tells Spectrum Writer to begin your output file with the standard opening HTML tags. These tags will be followed by the tag specified in the PRESCRIPT option above. Then the actual report will follow. After the report, the closing tag specified by the POSTSCRIPT option will appear, followed by Spectrum Writer's other standard closing HTML tags. When viewed on the Web, all text in the report (titles, column headings, data, Grand Totals) will be red.

The above example uses the PRESCRIPT and POSTSCRIPT options in addition to the HTML option. You can also use these options *instead* of the standard HTML option. This allows you to specify *all* of the HTML tags yourself. For example:

```
OPTION: HTMLAID
OPTION: PRESCRIPT(' <HTML>')
OPTION: PRESCRIPT(' <HEAD>')
OPTION: PRESCRIPT(' <TITLE>ABC COMPANY SALES REPORT</TITLE>')
OPTION: PRESCRIPT(' </HEAD>')
OPTION: PRESCRIPT(' <BODY BACKGROUND="BACKLOGO.JPG">')
OPTION: PRESCRIPT(' <PRE>')
*
OPTION: POSTSCRIPT(' </PRE>')
OPTION: POSTSCRIPT(' </BODY>')
OPTION: POSTSCRIPT(' </HTML>')
```

The statements above show an alternative to using the HTML option. These statements result in a Web report similar to that produced by just using the HTML option. However, we did add one special option to the BODY tag. The BACKGROUND option names an image file on the Web site that contains a corporate logo graphic. That image will be repeated as necessary and used as the background for the Web report. This option can point to any image file on your site. For example, instead of a logo, you might want to use an image file that contains a "textured" background for your report.

Another option you can specify in the BODY tag is the BGCOLOR option. Use it to specify a solid background color for your Web report. For example:

```
OPTION: PRESCRIPT(' <BODY BGCOLOR=PI NK>')
```

Note: If you choose to write all of your own HTML tags (rather than use the HTML option), you should also specify the HTMLAID option ([page 564](#)). That option helps solve some potential alignment problems in reports that contain HTML tags.

Summary of Options for Web Reports

The following table summarizes some of the OPTIONS statement options that are useful when creating Web reports.

OPTIONS RELATED TO WEB REPORTS	
OPTION	DESCRIPTION
HTMLAID	<p>Tells Spectrum Writer that you will be putting HTML tags within your report and that Spectrum Writer should recognize and support those tags. This option itself does not cause Spectrum Writer to add any HTML code to your report. See the OPTIONS statement syntax for a complete description of the HTMLAID option (page 564). Example:</p> <p>OPTION: HTMLAID</p>
HTML['title']	<p>Tells Spectrum Writer to wrap standard HTML code around the report. It also lets you specify an optional HTML title for the Web page. This option also turns on the HTMLAID option. Examples:</p> <p>OPTION: HTML</p> <p>OPTION: HTML('ABC COMPANY SALES REPORT')</p>
NOCC	<p>Tells Spectrum Writer not to begin each output line with a "carriage control" character. Such characters are only necessary when the output is being sent to a mainframe printer. You do not need to specify NOCC if you specify either the HTML or HTMLAID options, since these options automatically invoke the NOCC option. Example:</p> <p>OPTION: NOCC</p>
NOCOLHDGS	<p>Tells Spectrum Writer to not print column headings for the report. When the default column headings have alignment problems, specify this option and then use TITLE statements to specify your own column headings. Example:</p> <p>OPTION: NOCOLHDGS</p>
NOUNDERSCORES	<p>Tells Spectrum Writer not to underscore the column headings in the report. This is often desirable for reports that will be viewed online, since the underscore line uses up an additional line on the screen. Example:</p> <p>OPTION: NOUNDERSCORES</p>

OPTIONS RELATED TO WEB REPORTS (CONTINUED)	
OPTION	DESCRIPTION
POSTSCRIPT('text')	<p>Tells Spectrum Writer to write this text once, after the actual report. This option allows you to specify any closing HTML tags that you need. You can have as many POSTSCRIPT options as you want. If the HTML option is also specified, the POSTSCRIPT text(s) will appear <i>after</i> the report and immediately <i>before</i> the closing HTML tags created by the HTML option. Example:</p> <p>OPTION: POSTSCRIPT(<' /FONT>')</p>
PRESCRIPT('text')	<p>Tells Spectrum Writer to write this text once, before the actual report. This option allows you to specify any opening HTML tags that you need. You can have as many PRESCRIPT options as you want. If the HTML option is also specified, the PRESCRIPT text(s) will appear <i>before</i> the report and immediately <i>after</i> the initial HTML tags created by the HTML option. Example:</p> <p>OPTION: PRESCRIPT(' ')</p>
TITLEONCE	<p>Tells Spectrum Writer to write out the report titles (and column headings) only once — at the beginning of the report. This prevents titles from seeming to appear "randomly" throughout the report when a viewer pages through it online. Example:</p> <p>OPTION: TITLEONCE</p>

Common HTML Tags

The following table lists some common HTML tags that you may want to use in your Web reports. Please note, however, that there are a number of different versions of HTML and it is constantly evolving as new versions of Web browsers are released. In addition, different browsers sometimes process the same HTML tag in slightly different ways. At the time of publication, the HTML tags shown below work as described. Of course, we cannot guarantee they will always work the same way in the future. Also, future versions of HTML will undoubtedly include many new features not documented here. You can always perform a web search to learn what features are available in the current version of HTML. At the time of this writing, one online resource for the current HTML specifications is at WWW.W3.ORG.

Common HTML Tags

COMMON HTML TAGS	
HTML TAG	DESCRIPTION
 ... 	<p>Specifies the beginning of a hypertext link. When the user clicks on the text or graphic within these tags, the action implied by the "url" will be performed.</p> <p>TITLE: ' ' 0 ' CLICK HERE TO READ ABOUT THE ABC COMPANY'</p>
	<p>Specifies a label that can be referred to in a hypertext link. It requires no closing tag.</p> <p>OPTION: PRESCRIPT(' ')</p>
 ... 	<p>Makes the text within the tags bold.</p> <p>TITLE: ' SALES BY REGION'</p>
<BLINK> ... </BLINK>	<p>Makes the text within the tags blink on and off. (This option may only work with Netscape browsers.)</p> <p>TITLE: ' SALES BY <BLINK>REGION</BLINK>'</p>

	<p>Inserts a line "break" in the report. It requires no closing tag.</p> <p>COLUMNS: REGION CUSTOMER AMOUNT '
'</p>
<CENTER>...</CENTER>	<p>Centers the text within the tags.</p> <p>TITLE: ' <CENTER>SALES BY REGION</CENTER>'</p>
<FONT [SIZE=n] [FACE="fontname"] [COLOR=color] ...	<p>This specifies information about the font to be used for the text within the tags. Font sizes between 1 (smallest) and 7 (largest) are currently supported. For example:</p> <p>TITLE: ' SALES BY REGION'</p> <p>The 16 colors currently supported by name in most browsers are: black, olive, teal, red, blue, maroon, navy, gray, lime, fuchsia, white, green, purple, silver, yellow and aqua. Example:</p> <p>TITLE: ' SALES BY REGION'</p>

COMMON HTML TAGS (CONTINUED)	
HTML TAG	DESCRIPTION
<FONT [SIZE=n] [FACE="fontname"] [COLOR=color] ...	<p>The face parm specifies the name of the font to be used. However, to preserve the column alignment of your report, in most cases you should only use <i>non-proportional</i> fonts. Another caution: the font you specify may not be available on your viewer's PC, and a substitute font may be substituted by the browser, causing unexpected results. Use this option with caution. Example:</p> <p>TITLE: 'SALES BY REGION'</p>
<H1>...</H1> <H2>...</H2> ... <H6>...</H6>	<p>This formats the text within the tags as a "level 1" (or 2, 3, etc.) header.</p> <p>TITLE: ' <H1>SALES BY REGION</H1> '</p>
<HR>	<p>This produces a "horizontal rule" (a line) in your report. It requires no closing tag.</p> <p>TITLE: 'ABC COMPANY SALES REPORT' TITLE: ' <HR> '</p>
<I> ... </I>	<p>This makes the text within the tags italic.</p> <p>TITLE: 'SALES BY <I>REGION</I> '</p>
	<p>Specifies that a graphic image should be placed here. The "url" must be the name of the graphic file, which will often be a .gif or a .jpg file. The optional ALIGN parm determines how any text that follows the image will be aligned with it. The optional WIDTH parm specifies the size that the image should take up in the display. This tag requires no closing tag.</p> <p>TITLE: ' '</p>
<MARQUEE>...</MARQUEE>	<p>Makes the text within the tag scroll across the screen. (This option may only work with Microsoft browsers.)</p> <p>TITLE: 'SALES BY<MARQUEE>REGION</MARQUEE>'</p>
<SMALL> ... </SMALL>	<p>Makes the text within the tags smaller.</p> <p>TITLE: 'SALES BY <SMALL>REGION</SMALL>'</p>

Common HTML Tags

COMMON HTML TAGS (CONTINUED)	
HTML TAG	DESCRIPTION
<code><SUB> ... </SUB></code>	Makes the text within the tags subscripts (smaller and lower than the regular baseline). TITLE: 'SALES BY _{REGION}'
<code><SUP> ... </SUP></code>	Makes the text within the tags superscripts (smaller and higher than the regular baseline). TITLE: 'SALES BY ^{REGION}'
<code><TABLE [BORDER=n] [CELLPADDING=n] [CELLSPACING=n]> ... </TABLE></code>	Specifies that the text and images within these tags should be formatted as a table. The optional BORDER parm determines whether there will be a border around the cells of the table. The optional CELLPADDING value tells how much space to leave between the border of a cell and its contents. The optional CELLSPACING value tells how much space to leave between the cells of the table. For example: OPTION: PRESCRIPT(' <TABLE BORDER=1 CELLPADDING=2>')
<code><TD></code>	Specifies the beginning of a new table item ("data cell") within a table row. It requires no closing tag. For example: COLUMNS: ' <TR>' ' <TD>' REGION ' <TD>' EMPL-NAME
<code><TR></code>	Specifies the beginning of a new table row (within a table). It requires no closing tag. For example: COLUMNS: ' <TR>' ' <TD>' REGION ' <TD>' EMPL-NAME
<code><U> ... </U></code>	Underlines the text within the tags. TITLE: 'SALES BY <U>REGION</U>'
<code><!-- ... --></code>	This is an HTML comment. It has no effect on what the viewer sees on the Web browser. It can be used to document your HTML code. It can also be used in some situations to help align a Web report. Use it to strip excess bytes from a line of Spectrum Writer output. (See page 315 for an example of this.)