

TIP

Including too many fonts can make your page difficult to read. Don't use more than two or three typefaces within a single page.

TIP

Use absolute units only when you can predict or fix the size and dimensions of the output device.

Setting the Font Size

Tammy would like the Welcome heading on her home page to be displayed in slightly smaller text than is generally set by a browser's internal style sheet. The style to change the font size is

```
font-size: size;
```

where *size* is a length measurement. Lengths can be specified in four different ways:

- with a unit of measurement
- as a percentage of the size of the containing element
- with a keyword description
- with a keyword expressing the size relative to the size of the containing element

If you choose to specify lengths using measurement units, you can use absolute units or relative units. **Absolute units** are units that are fixed in size regardless of the device rendering the Web page. They are specified in one of five standard units of measurement: mm (millimeters), cm (centimeters), in (inches), pt (points), and pc (picas). Points and picas might not be as familiar to you as inches, millimeters, and centimeters. For comparison, there are 72 points in an inch, 12 points in a pica, and 6 picas in an inch. Size values for any of these measurements can be whole numbers (0, 1, 2 ...) or decimals (0.5, 1.6, 3.9 ...). For example, if you want your text to be 1/2 inch in size, you can use any of the following styles:

```
font-size: 0.5in  
font-size: 36pt  
font-size: 3pc
```

Note that you should not insert a space between the size value and the unit abbreviation.

Absolute measurements are appropriate when you know the physical properties of an output device and want to fix a size to a specific value. Of course, this is not often the case with Web pages that can be displayed on a variety of devices and under several possible screen or page resolutions. To cope with the uncertainty about how their pages will be viewed, many Web page designers opt to use **relative units**, which are expressed relative to the size of other objects within the Web page. One commonly used relative unit is the **em unit**. The exact meaning of the em unit depends on its use in the style sheet. If the em unit is used for setting font size, it expresses the size relative to the font size of the parent element. For an h1 heading, the parent element is the Web page body. Thus, the style rule

```
h1 {font-size: 2em;}
```

sets the font size of h1 headings to twice the font size of body text. If body text is displayed in a 12-point font, this style will cause h1 headings to be displayed in a 24-point font. On the other hand, if the h1 heading is nested within another element, such as a section element, the size of the h1 heading will be twice the size of text in that parent element. Context is important when interpreting the effect of the em unit.

One of the great advantages of relative units like the em unit is that they can make your page **scalable**, allowing the page to be rendered the same way no matter what font size is used by the browser. Setting the font size of h1 headings to 1.5em ensures the heading will be 50% larger than the body text for all users.

Another way to create relative font sizes is to express the font size as a percentage. Like the em unit, percentages are based on the font size of the parent element. The style

```
h1 {font-size: 200%;}
```

sets the font size of h1 headings to 200%, or twice the font size of body text.

Another unit of measurement widely used on the Web is the **pixel**, which represents a single dot on the output device. The size or **resolution** of most output devices is typically expressed in terms of pixels. Thus a 1280 × 720 screen resolution on a computer monitor is 1280 pixels wide by 720 pixels tall, for a total of 921,600 pixels or 0.92 megapixels. A pixel is a relative unit because the actual rendered size depends on the **density** of the output device. A Windows PC, for example, has a density of 96 dpi (dots per inch), while a Macintosh computer has a density of 72 dpi. Some mobile phones have densities as high as 200 or 300 dpi. The pixel measure is the most precise unit of measure and gives designers the most control over the appearance of a page; however, pixels are not scalable. This can pose a problem for visually impaired users who need larger fonts, or for users of mobile devices with very dense screens.

Finally, you also can express font sizes using one of the following keywords: `xx-small`, `x-small`, `small`, `medium`, `large`, `x-large`, `xx-large`, `larger`, or `smaller`. The size corresponding to each of these keywords is determined by the browser. Note that the `larger` and `smaller` keywords are relative sizes, making the font size of the element one size larger or smaller than the surrounding text. For example, the following set of styles causes the `body` text to be displayed in a small font, while `h2` text is displayed in a font one size larger (medium):

```
body {font-size: small;}
h2   {font-size: larger;}
```

Tammy suggests that you set the size of the `h1` headings to 1.7em, making the headings 70% larger than the default size of the body text in the document.

To set the font size of the `h1` headings:

1. Return to the `sa_styles.css` file in your text editor.
2. Add the following style to the style rule for `h1` headings in the `section` element (see Figure 3-29):

```
font-size: 1.7em;
```

Figure 3-29

Setting the font size of `h1` headings

```
section h1 {
  background-color: rgb(125, 186, 240);
  color: white;
  font-size: 1.7em;
}
```

3. Save your changes to the file and then reload the `home.htm` file in your Web browser. Verify that the font size of the `h1` heading appears slightly smaller under the revised style sheet.