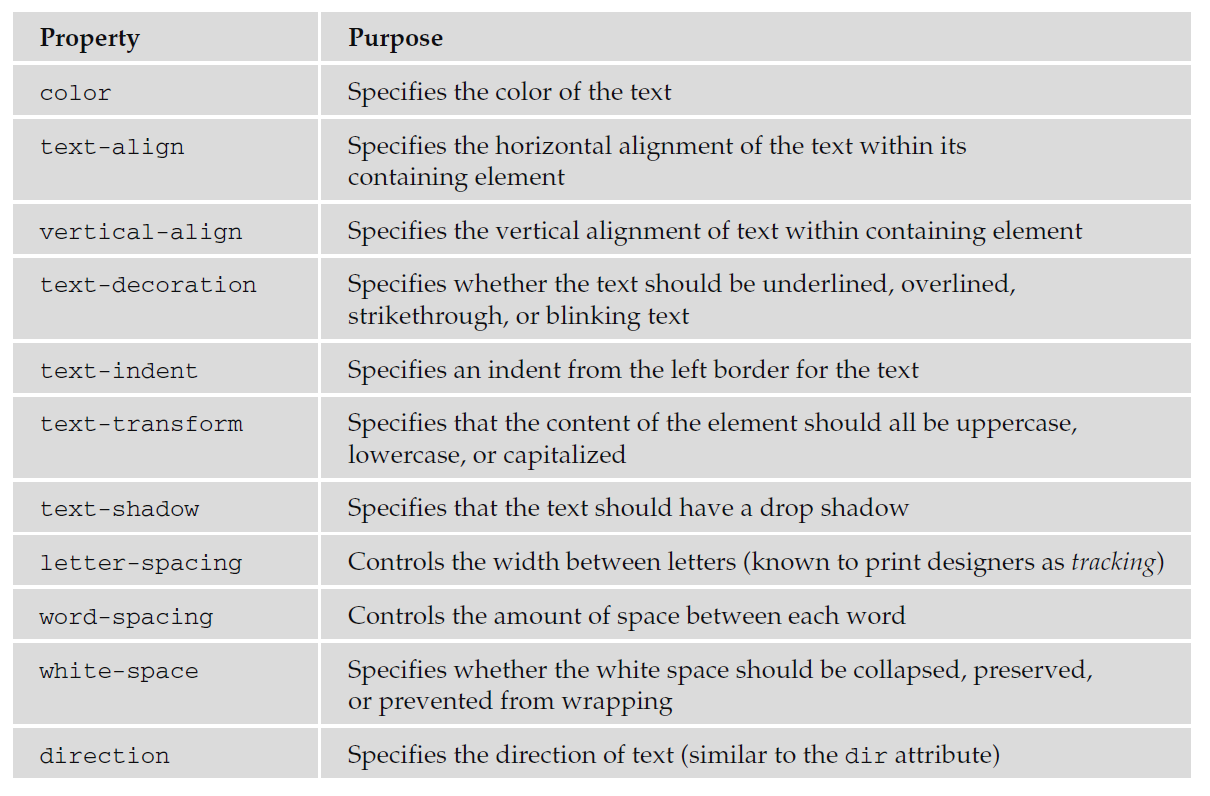
**Text Formatting**

In addition to the propertiesthat affect the font, there are several properties to affect the appearance or formatting of your text (independently from the font it is shown in). They are listed in the table that follows.



**The Color Property:**

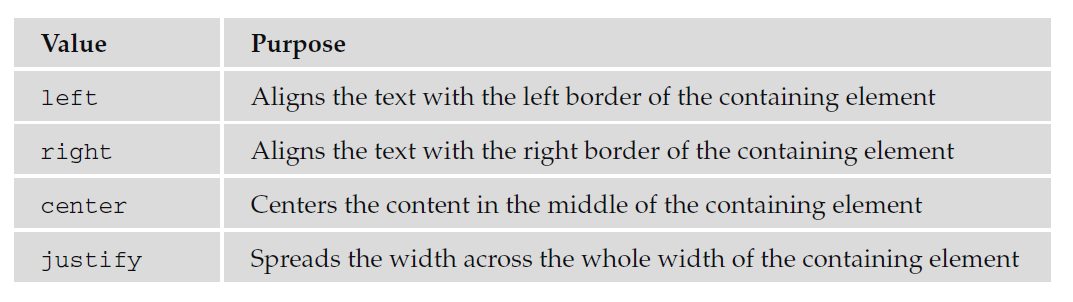
The color property allows you to specify the color of the text. The value of this property can either be a hex code for a color or a color name.

For example, the following rule would make the content of paragraph elements red:

p {color: #ff0000;}

**The text-align Property:**

The text - align property works like the deprecated align attribute would with text. It aligns the text within its containing element or the browser window. The table that follows displays possible values.



Here are the rules we can apply:

* Left Align - {text-align: left;}
* Right Align - {text-align:right;}
* center - {text-align:center;}
* justify - {text-align:justify;}

**<scroll more>**

**<scroll more>**

**<scroll more>**

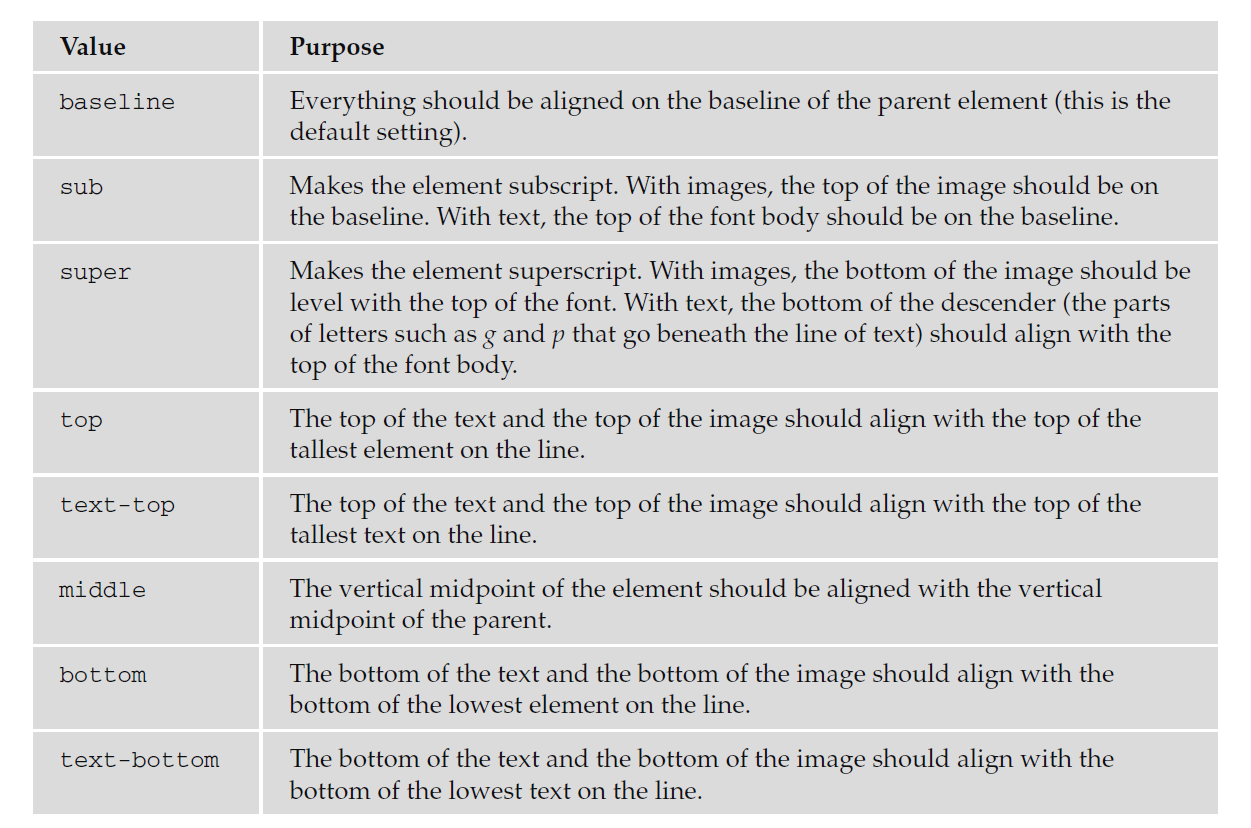
**The vertical - align Property:**

The vertical - align property is useful when working with inline elements, in particular images

and portions of text. It allows you to control their vertical positioning within the containing element, for example:

span.footnote {vertical-align: sub;}

It can take several values, as you can see in the table that follows.



**<scroll more>**

**<scroll more>**

**<scroll more>**

Example:

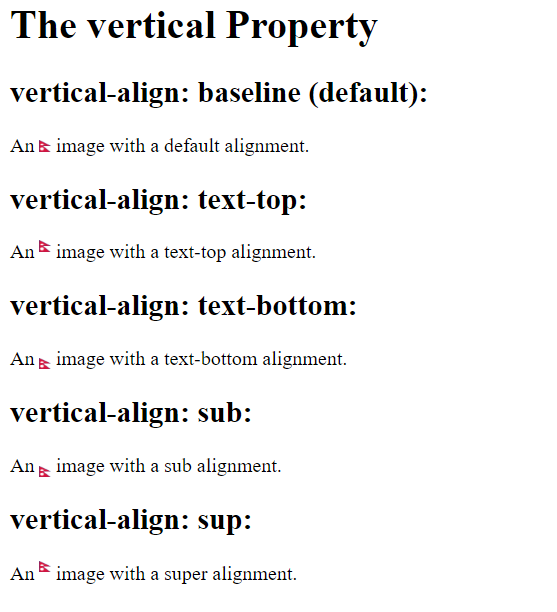


**<scroll more>**

**<scroll more>**

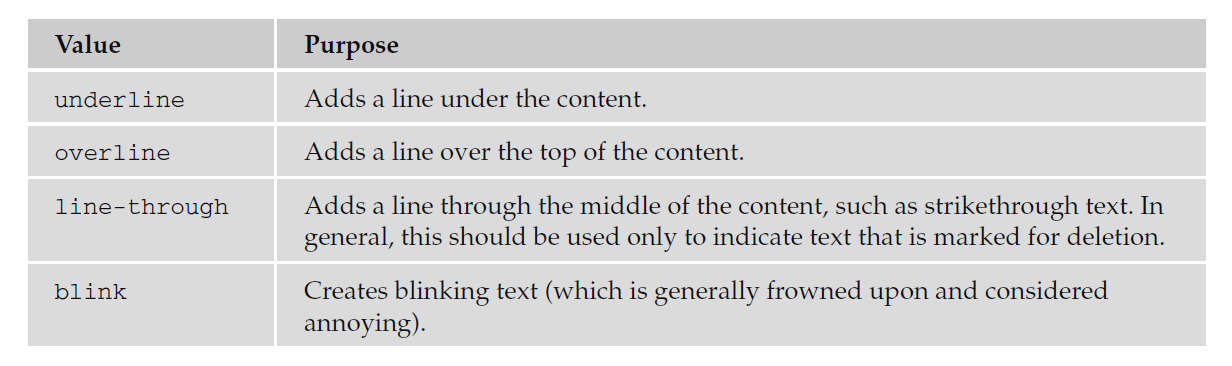
**<scroll more>**

Output:

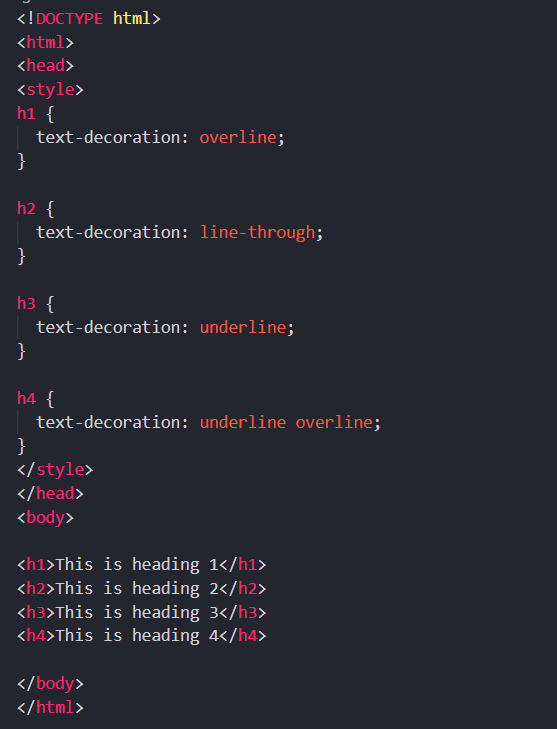


**The text - decoration Property:**

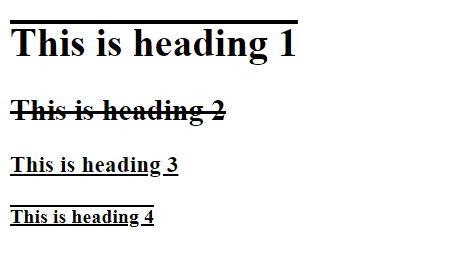
The text - decoration property allows you to specify the values shown in the following table.



Example:



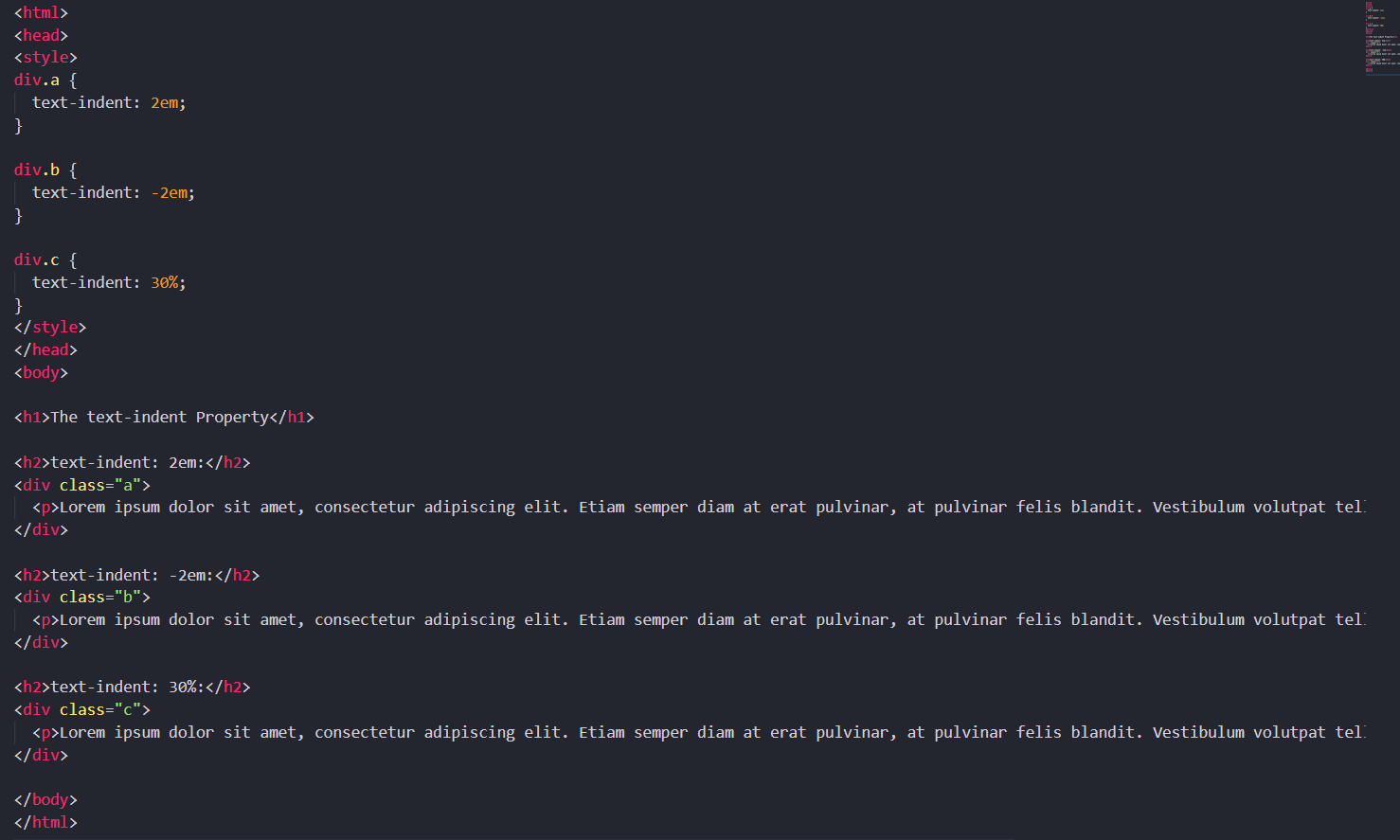
Output:



**The text-indent Property:**

The text - indent property allows you to indent the first line of text within an element. In the following example it has been applied to the second paragraph:

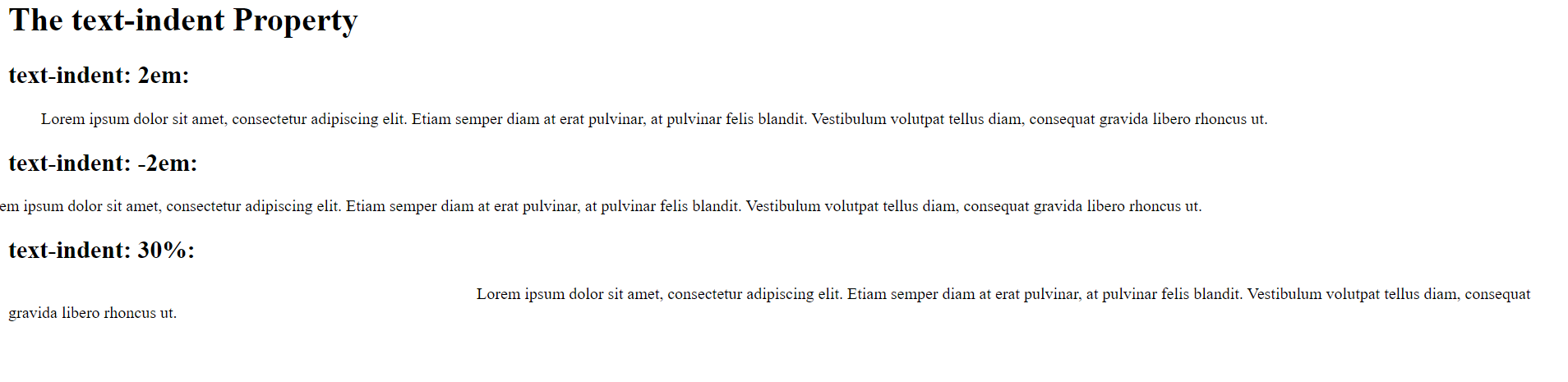
Example:



**<scroll more>**

**<scroll more>**

Output:



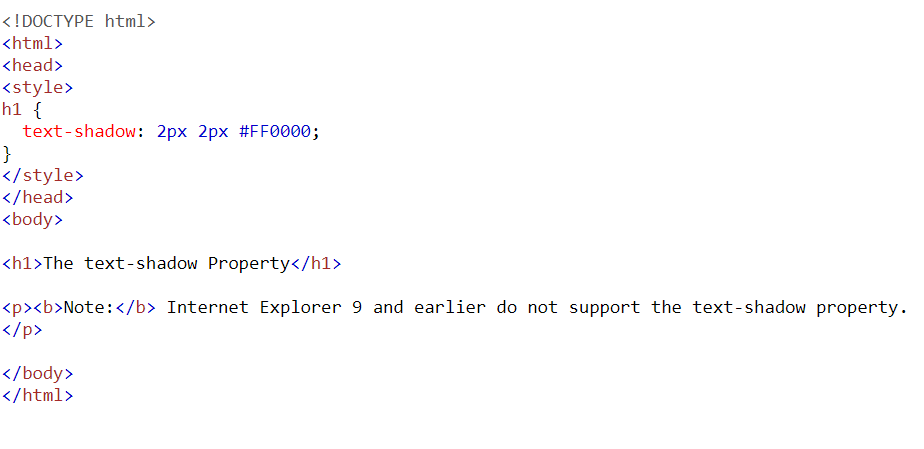
**The text-shadow Property:**

The text - shadow property is supposed to create a *drop shadow*, which is a dark version of the word just behind it and slightly offset. This has often been used in print media, and its popularity has meant that is has gained its own CSS property. The value for this property is quite complicated because it can take a color followed by three lengths:

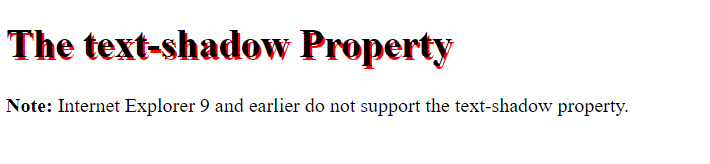
.dropShadow {text-shadow: #999999 10px 10px 3px;}

After the color has been specified, the first two lengths specify how far from the original text the drop shadow should fall (using X and Y coordinates), while the third specifies how blurred the drop shadow should be.

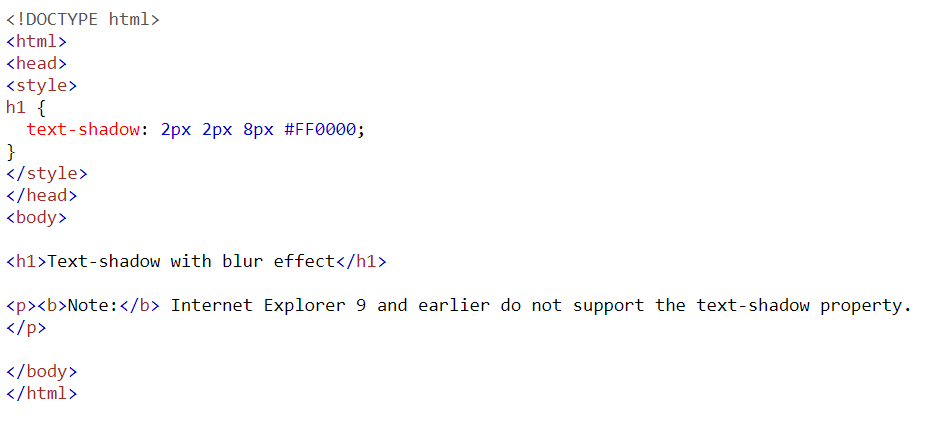
Example:



Output:



Example (With blur effect):

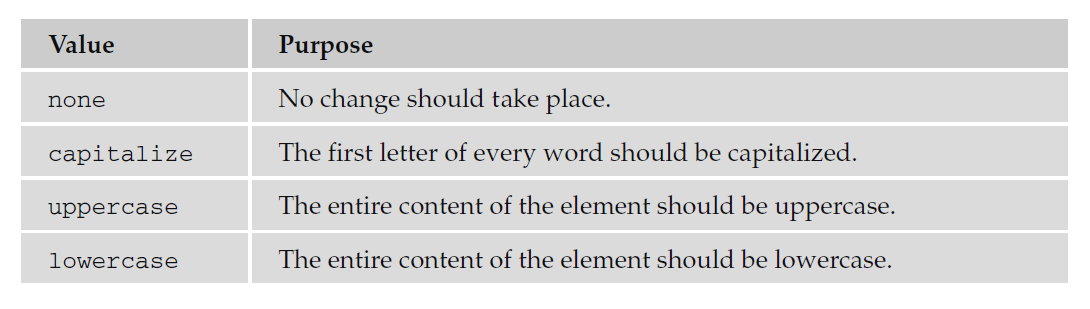


Output:

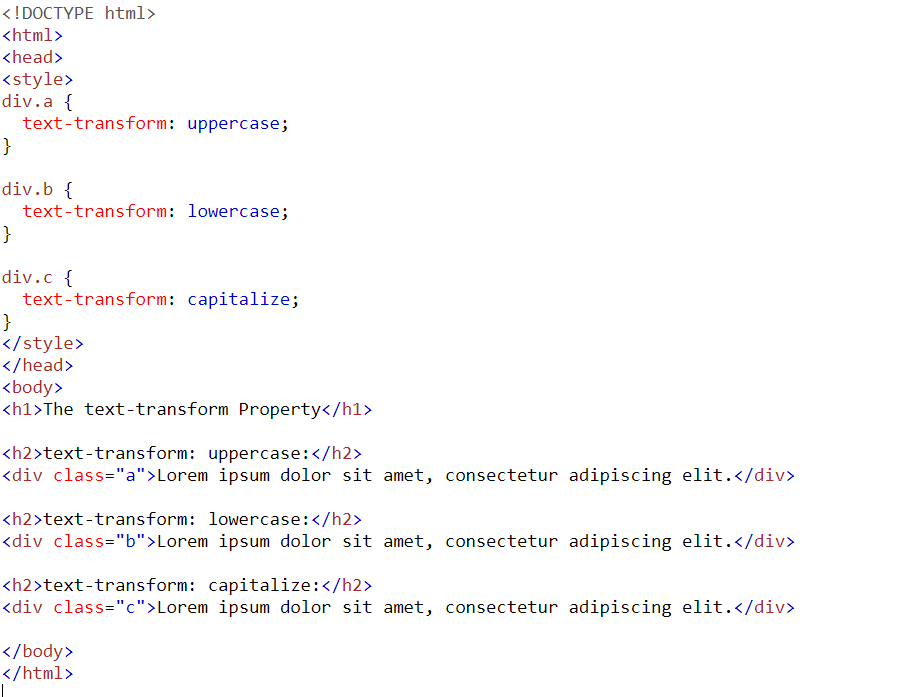


**The text - transform Property:**

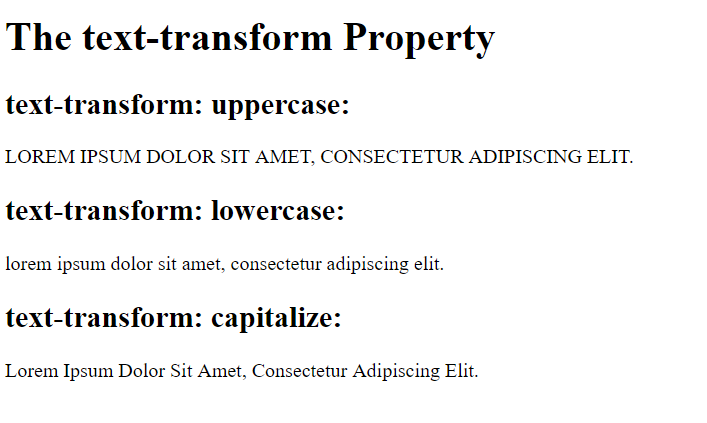
The text - transform property allows you to specify the case for the content of an element. The possible values are shown in the table that follows.



Example:



Output:

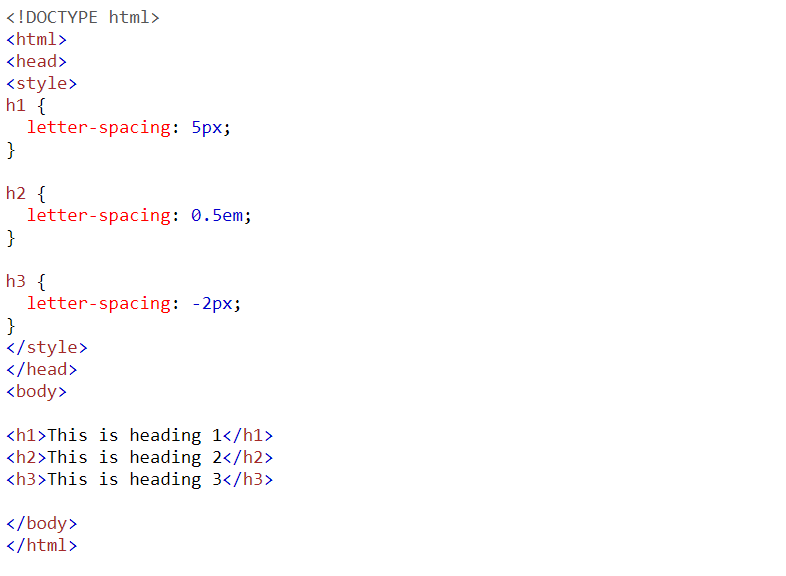


**The letter-spacing Property:**

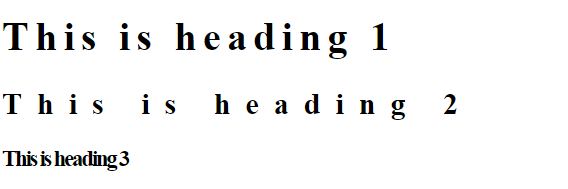
The letter - spacing property controls something that print designers refer to as ***tracking***: the gap between letters. **Loose tracking** indicates that there is a lot of space between letters, whereas **tight tracking** refers to letters being squeezed together. No tracking refers to the normal gap between letters for that font.

If you want to increase or decrease the spacing between letters, you are most likely to specify this in pixels or something known as an em (although it can be any unit of length that CSS supports)

Example:



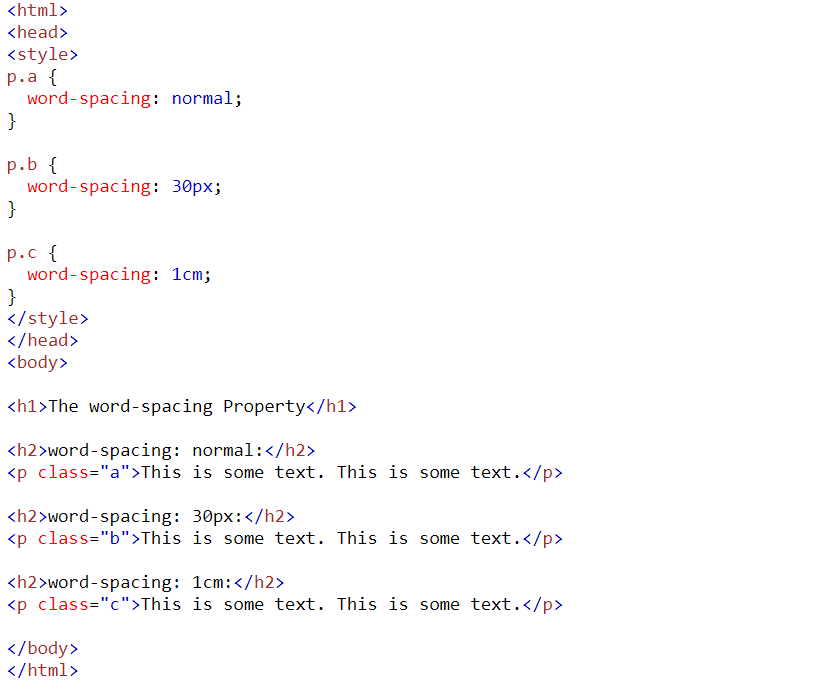
Output:



**The word-spacing Property:**

The word - spacing property sets the gap between words, and its value should be a unit of length.

Example:

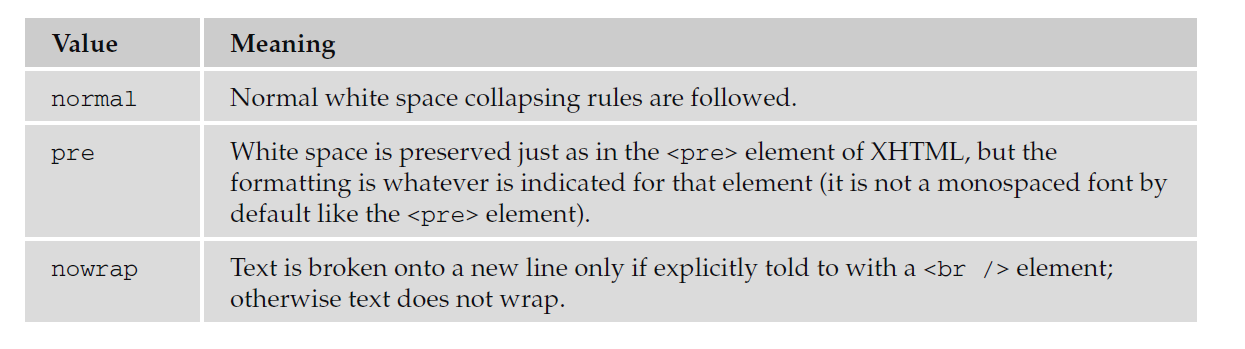
****

Output:

****

**The white-space Property:**

As you saw in Chapter 1, browsers change any two or more spaces next to each other into a single space and make any carriage returns a single space, too. The white - space property controls whether or not white space is preserved, offering the same results as the XHTML <pre> element where white space is preserved and the nowrap attribute where text is only broken onto a new line if explicitly told to. See the table that follows for the possible values for this property.



For example, you can use the white - space property like so:

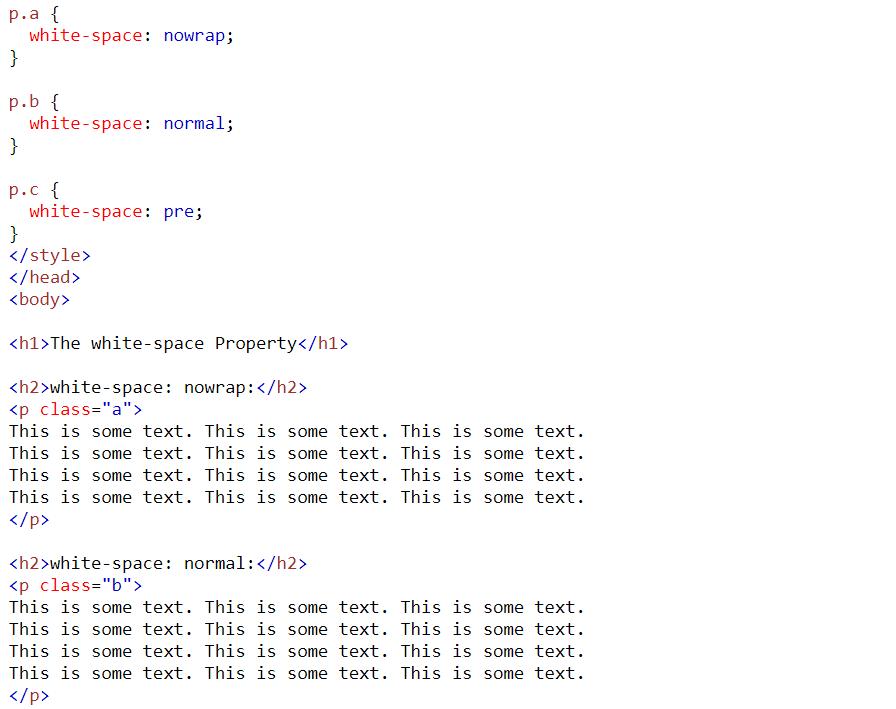
* pre {white-space:pre;}
* nowrap {white-space:nowrap;}

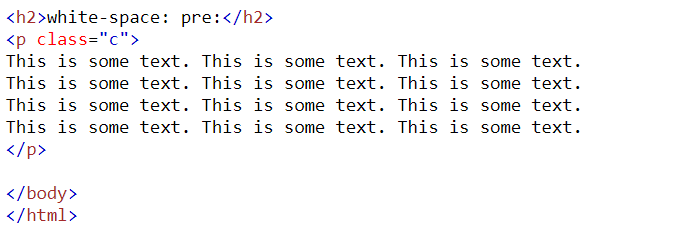
**<scroll more>**

**<scroll more>**

**<scroll more>**

Example:



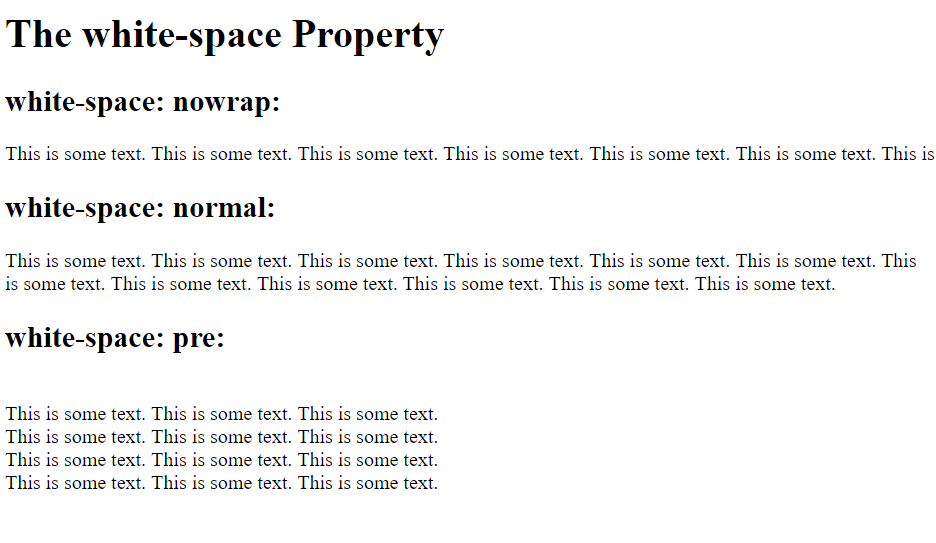


**<scroll more>**

**<scroll more>**

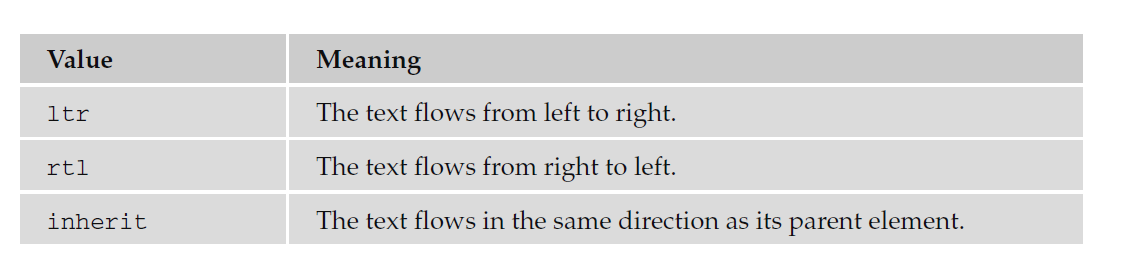
**<scroll more>**

Output:



**The direction Property:**

The direction property is rather like the dir attribute and specifies the direction in which the text should flow. The following table shows the possible values.

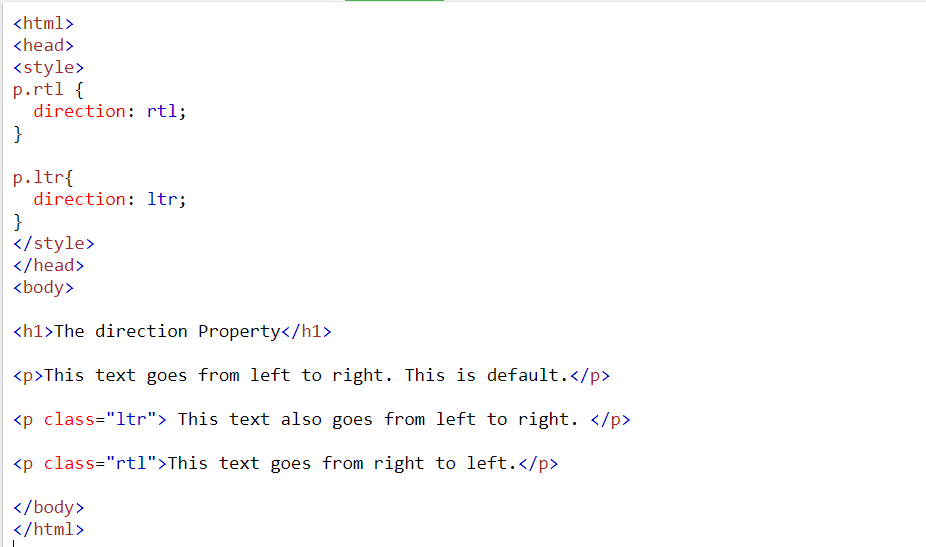


**<scroll more>**

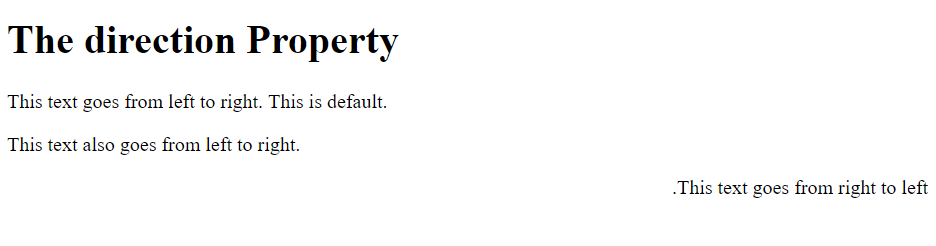
**<scroll more>**

**<scroll more>**

Example:



Output:



Q. Write a HTML and CSS code to generate the following output

*Hint: (with inherit value)*

