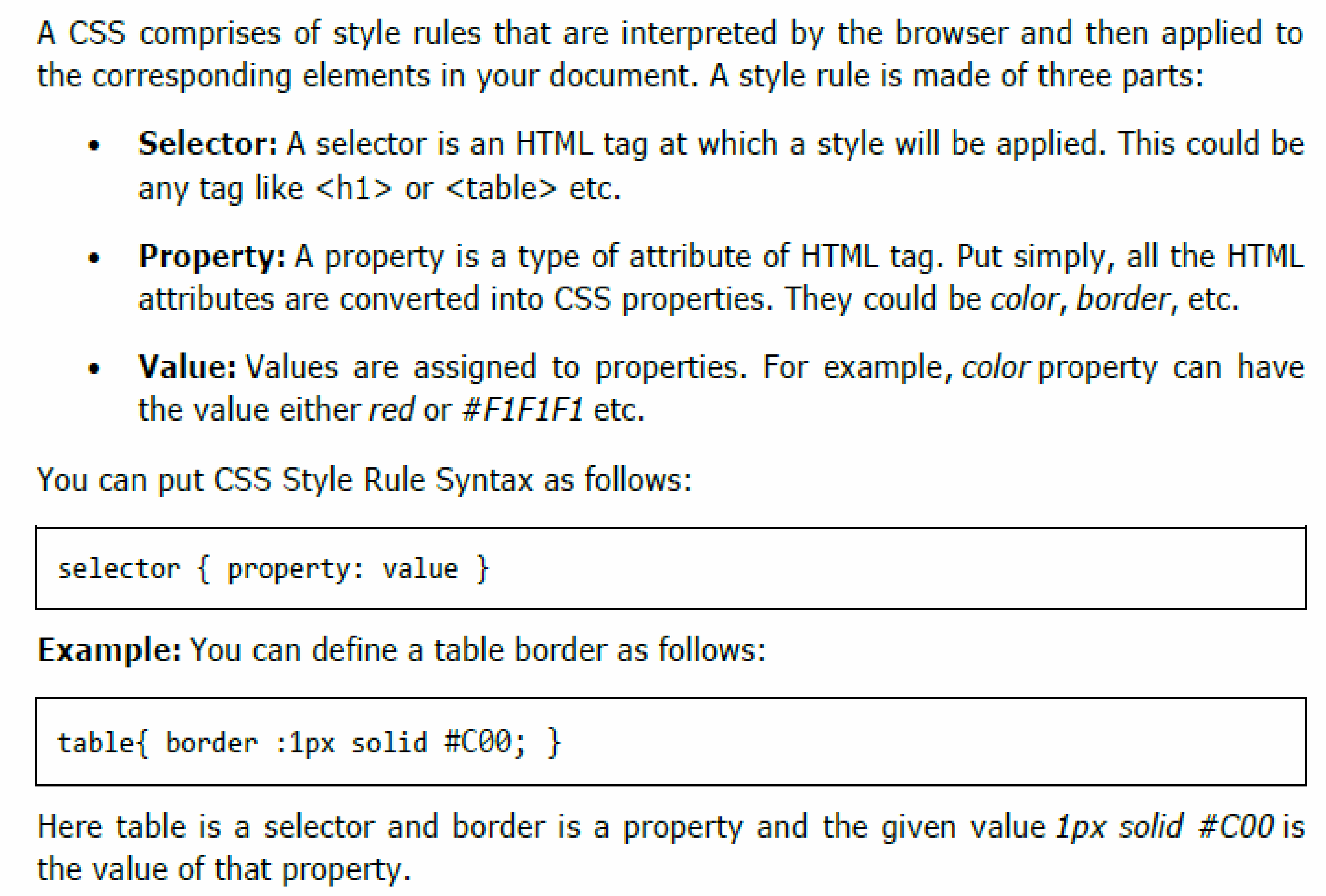
Cascading Style Sheets

CSS — Syntax



A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document. A style rule is made of three parts:

- **Selector:** A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.
- **Property:** A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border*, etc.
- **Value:** Values are assigned to properties. For example, *color* property can have the value either *red* or *#F1F1F1* etc.

You can put CSS Style Rule Syntax as follows:

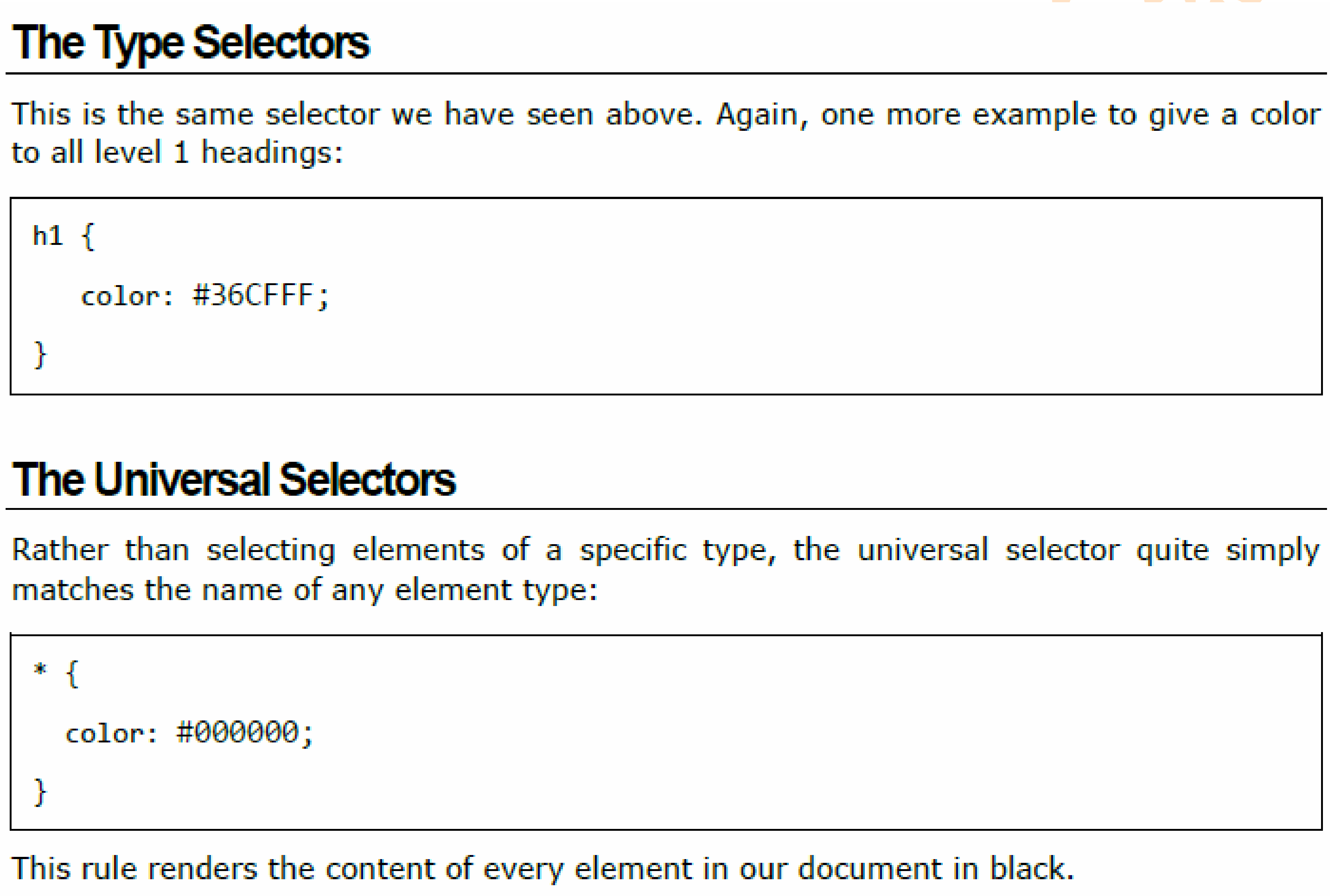
```
selector { property: value }
```

Example: You can define a table border as follows:

```
table{ border :1px solid #C00; }
```

Here table is a selector and border is a property and the given value *1px solid #C00* is the value of that property.

CSS — Syntax



The Type Selectors

This is the same selector we have seen above. Again, one more example to give a color to all level 1 headings:

```
h1 {

   color: #36CFFF;

}
```

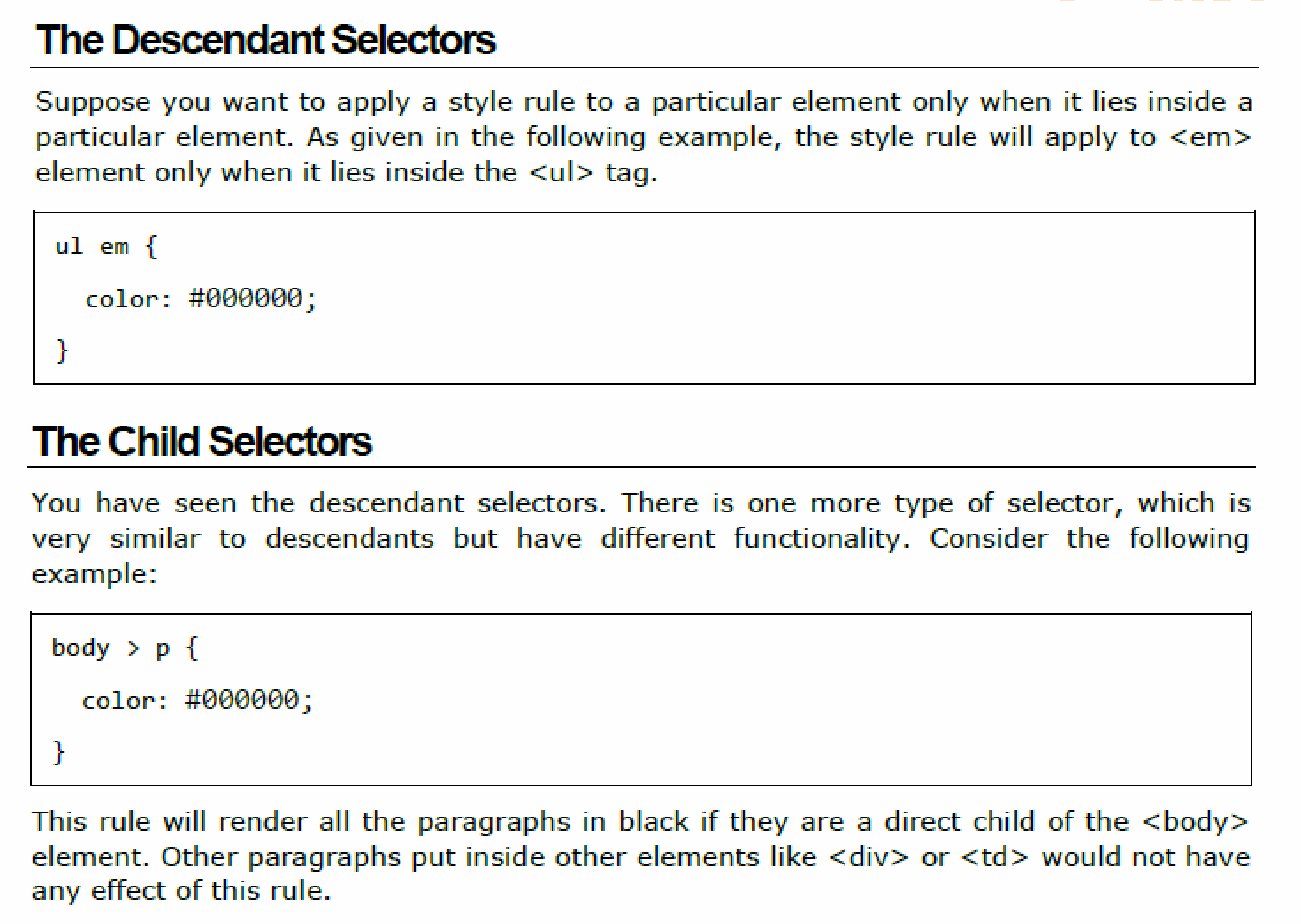
The Universal Selectors

Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type:

```
* {

   color: #000000;

}
```

This rule renders the content of every element in our document in black.

CSS — Syntax



The Descendant Selectors

Suppose you want to apply a style rule to a particular element only when it lies inside a particular element. As given in the following example, the style rule will apply to <em> element only when it lies inside the <ul> tag.

```
ul em {
  color: #000000;
}
```

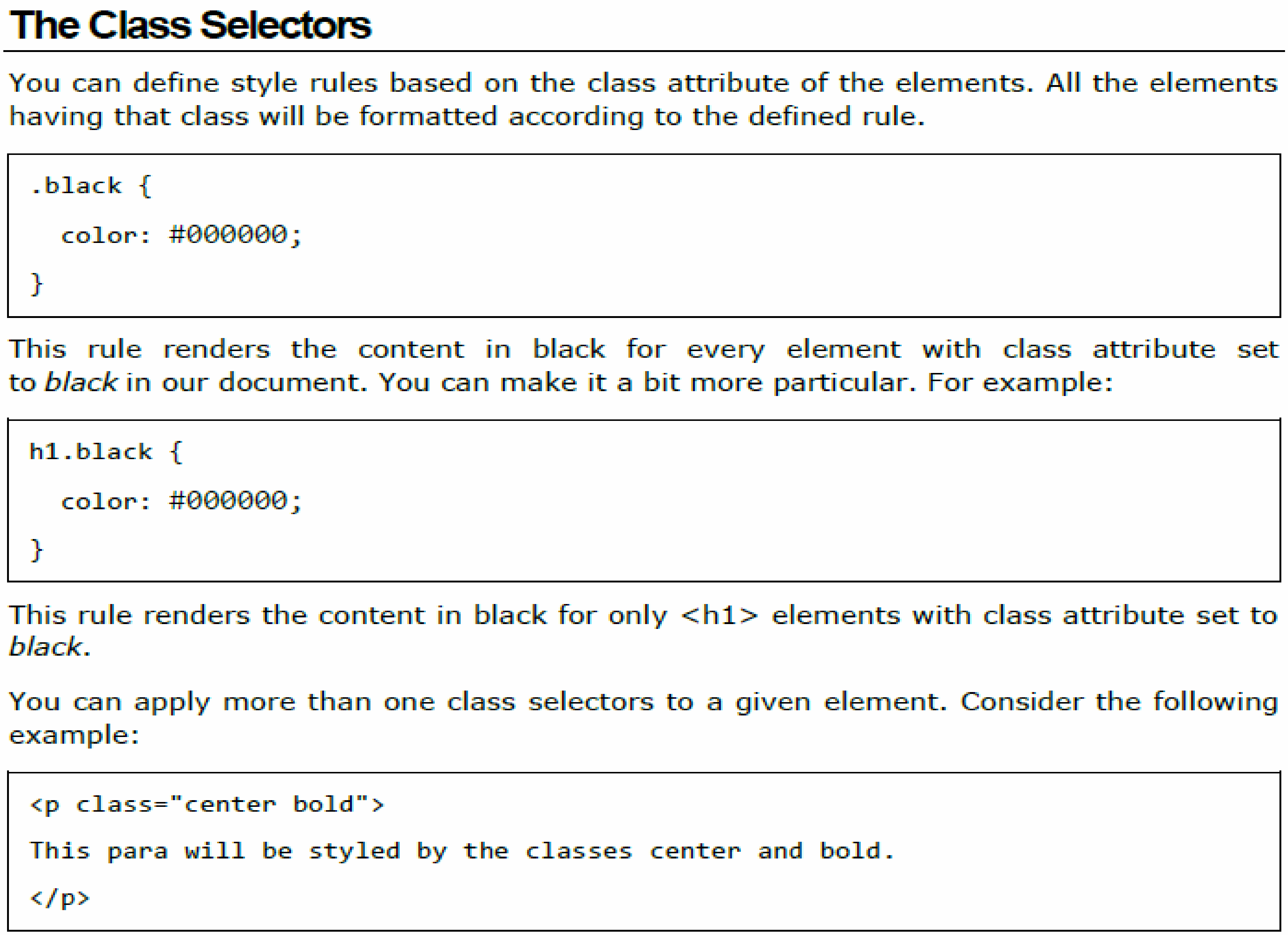
The Child Selectors

You have seen the descendant selectors. There is one more type of selector, which is very similar to descendants but have different functionality. Consider the following example:

```
body > p {
  color: #000000;
}
```

This rule will render all the paragraphs in black if they are a direct child of the <body> element. Other paragraphs put inside other elements like <div> or <td> would not have any effect of this rule.

CSS — Syntax



The Class Selectors

You can define style rules based on the class attribute of the elements. All the elements having that class will be formatted according to the defined rule.

```
.black {
   color: #000000;
}
```

This rule renders the content in black for every element with class attribute set to *black* in our document. You can make it a bit more particular. For example:

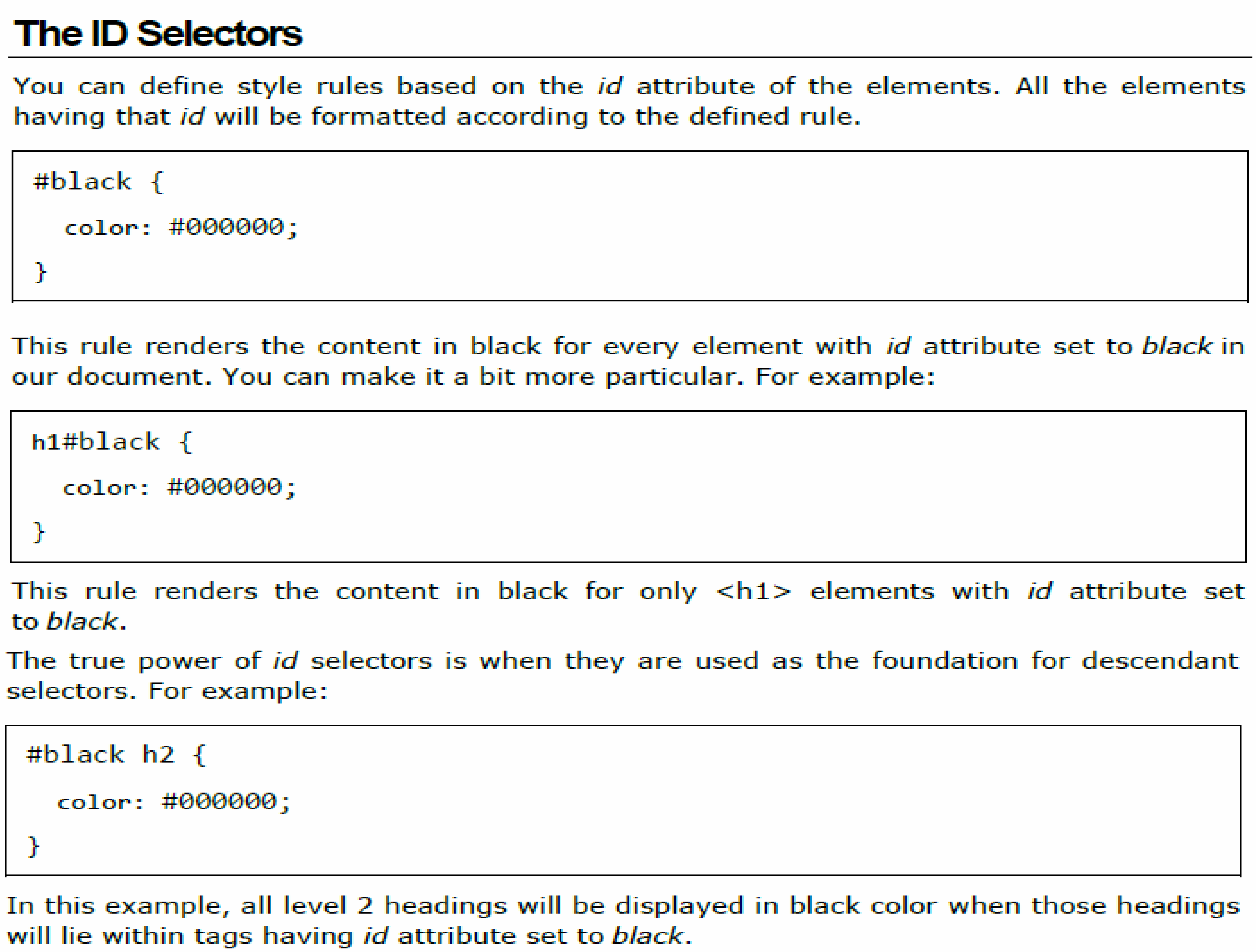
```
h1.black {
   color: #000000;
}
```

This rule renders the content in black for only <h1> elements with class attribute set to *black*.

You can apply more than one class selectors to a given element. Consider the following example:

```
<p class="center bold">
This para will be styled by the classes center and bold.
</p>
```

CSS — Syntax



The ID Selectors

You can define style rules based on the *id* attribute of the elements. All the elements having that *id* will be formatted according to the defined rule.

```
#black {
   color: #000000;
}
```

This rule renders the content in black for every element with *id* attribute set to *black* in our document. You can make it a bit more particular. For example:

```
h1#black {
   color: #000000;
}
```

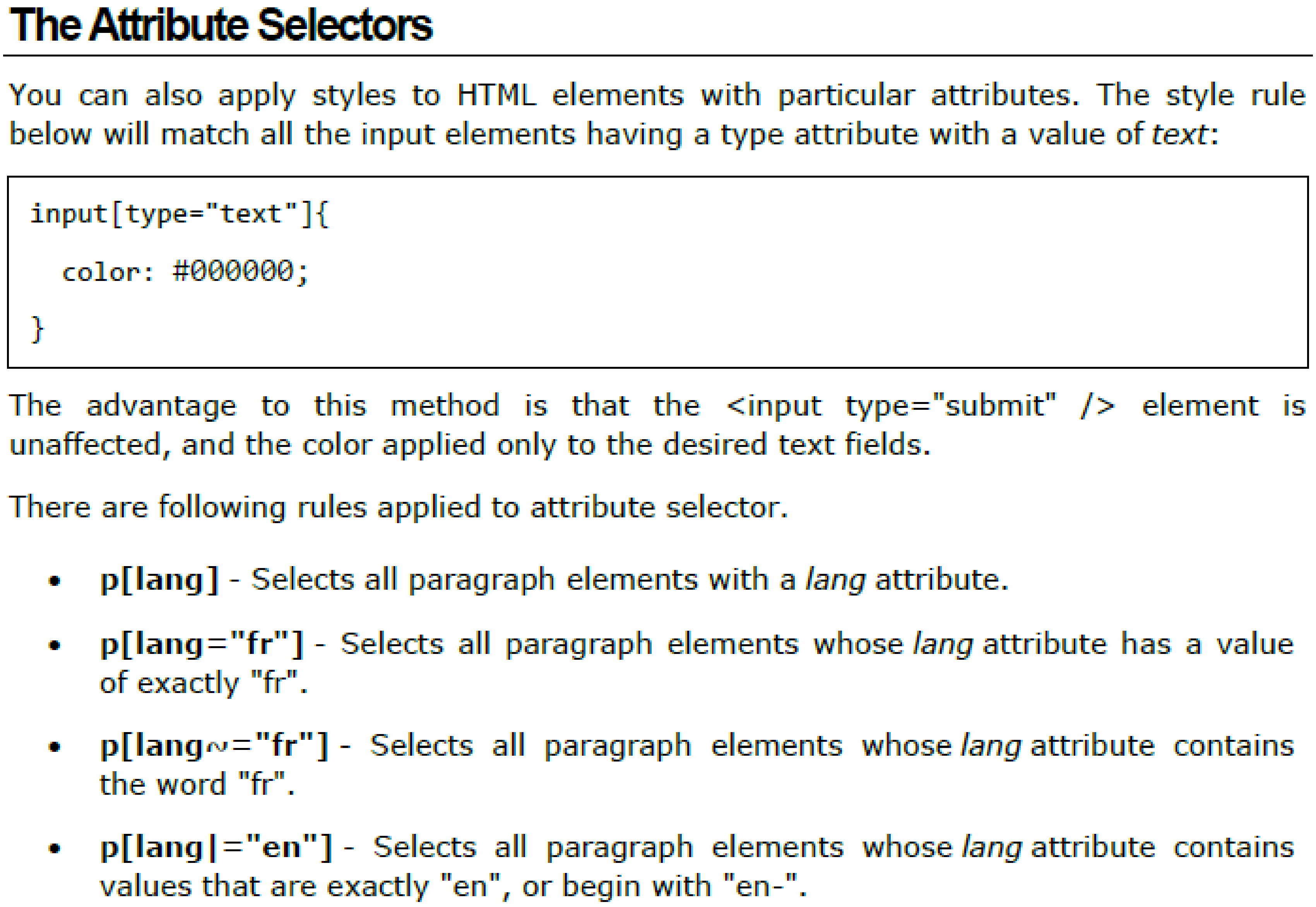
This rule renders the content in black for only <h1> elements with *id* attribute set to *black*.

The true power of *id* selectors is when they are used as the foundation for descendant selectors. For example:

```
#black h2 {
   color: #000000;
}
```

In this example, all level 2 headings will be displayed in black color when those headings will lie within tags having *id* attribute set to *black*.

CSS — Syntax



The Attribute Selectors

You can also apply styles to HTML elements with particular attributes. The style rule below will match all the input elements having a type attribute with a value of *text*:

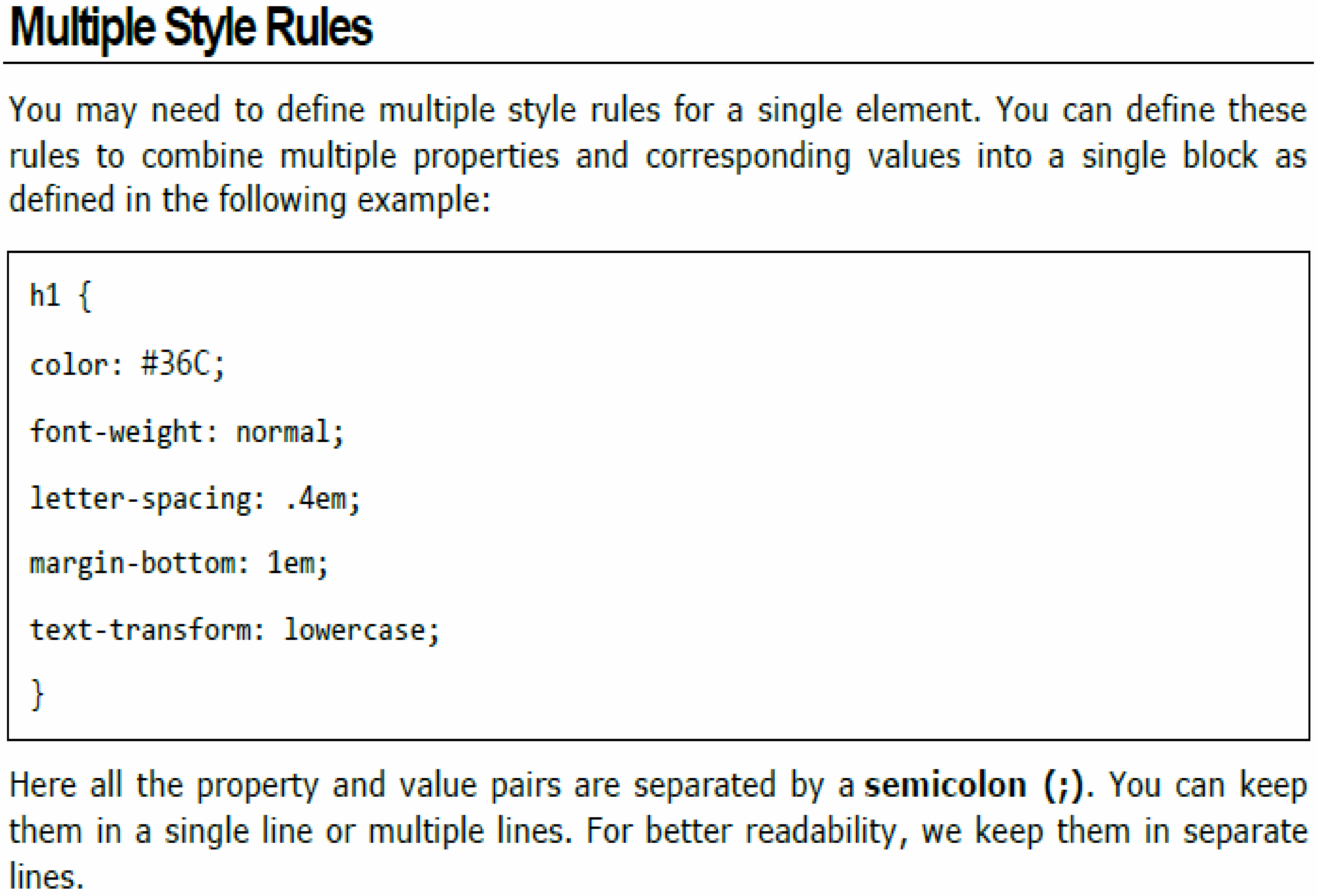
```
input[type="text"]{
  color: #000000;
}
```

The advantage to this method is that the <input type="submit" /> element is unaffected, and the color applied only to the desired text fields.

There are following rules applied to attribute selector.

- **p[lang]** - Selects all paragraph elements with a *lang* attribute.
- **p[lang="fr"]** - Selects all paragraph elements whose *lang* attribute has a value of exactly "fr".
- **p[lang~="fr"]** - Selects all paragraph elements whose *lang* attribute contains the word "fr".
- **p[lang|="en"]** - Selects all paragraph elements whose *lang* attribute contains values that are exactly "en", or begin with "en-".

CSS — Syntax



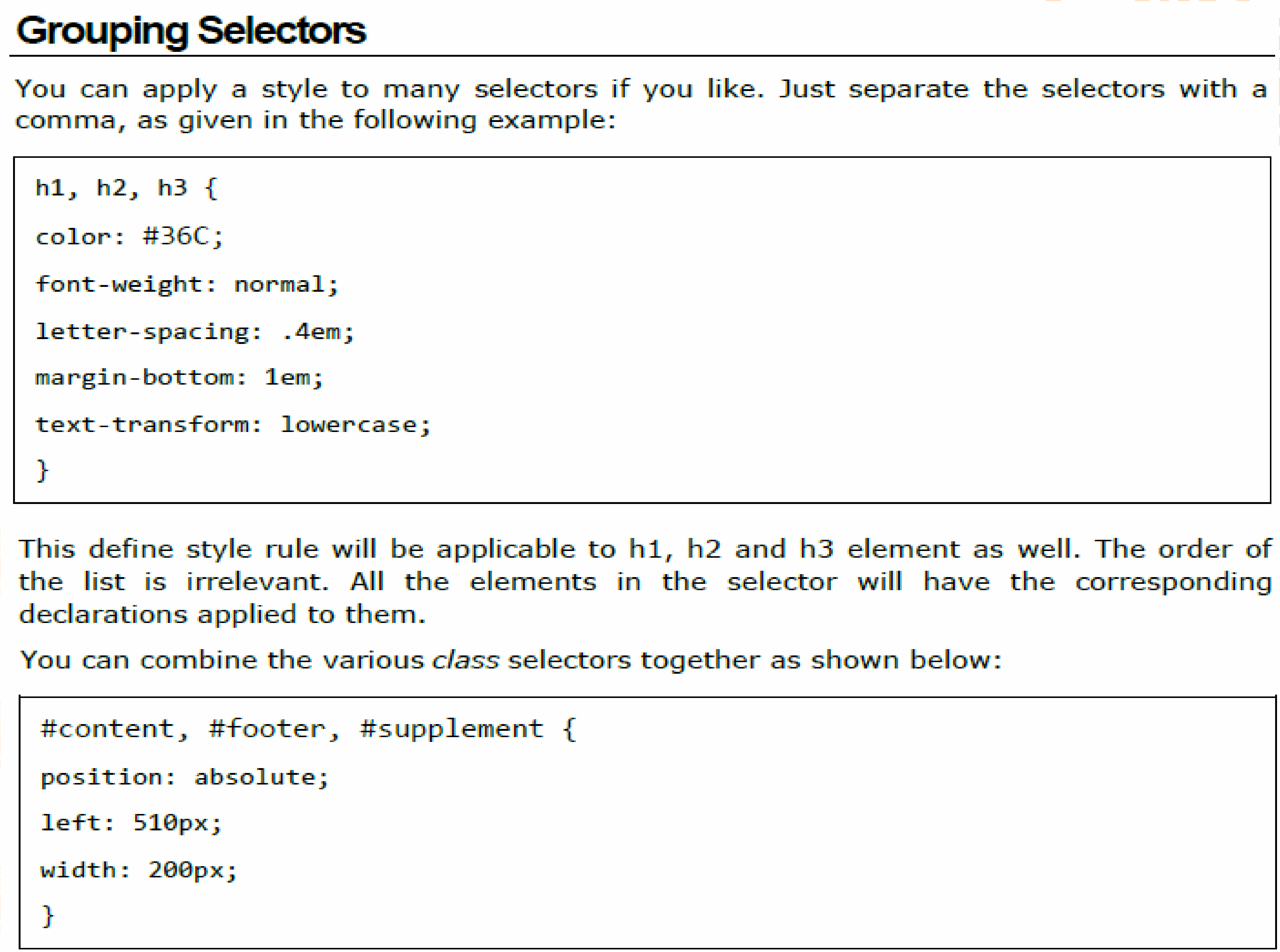
Multiple Style Rules

You may need to define multiple style rules for a single element. You can define these rules to combine multiple properties and corresponding values into a single block as defined in the following example:

```
h1 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}
```

Here all the property and value pairs are separated by a **semicolon (;)**. You can keep them in a single line or multiple lines. For better readability, we keep them in separate lines.

CSS — Syntax



Grouping Selectors

You can apply a style to many selectors if you like. Just separate the selectors with a comma, as given in the following example:

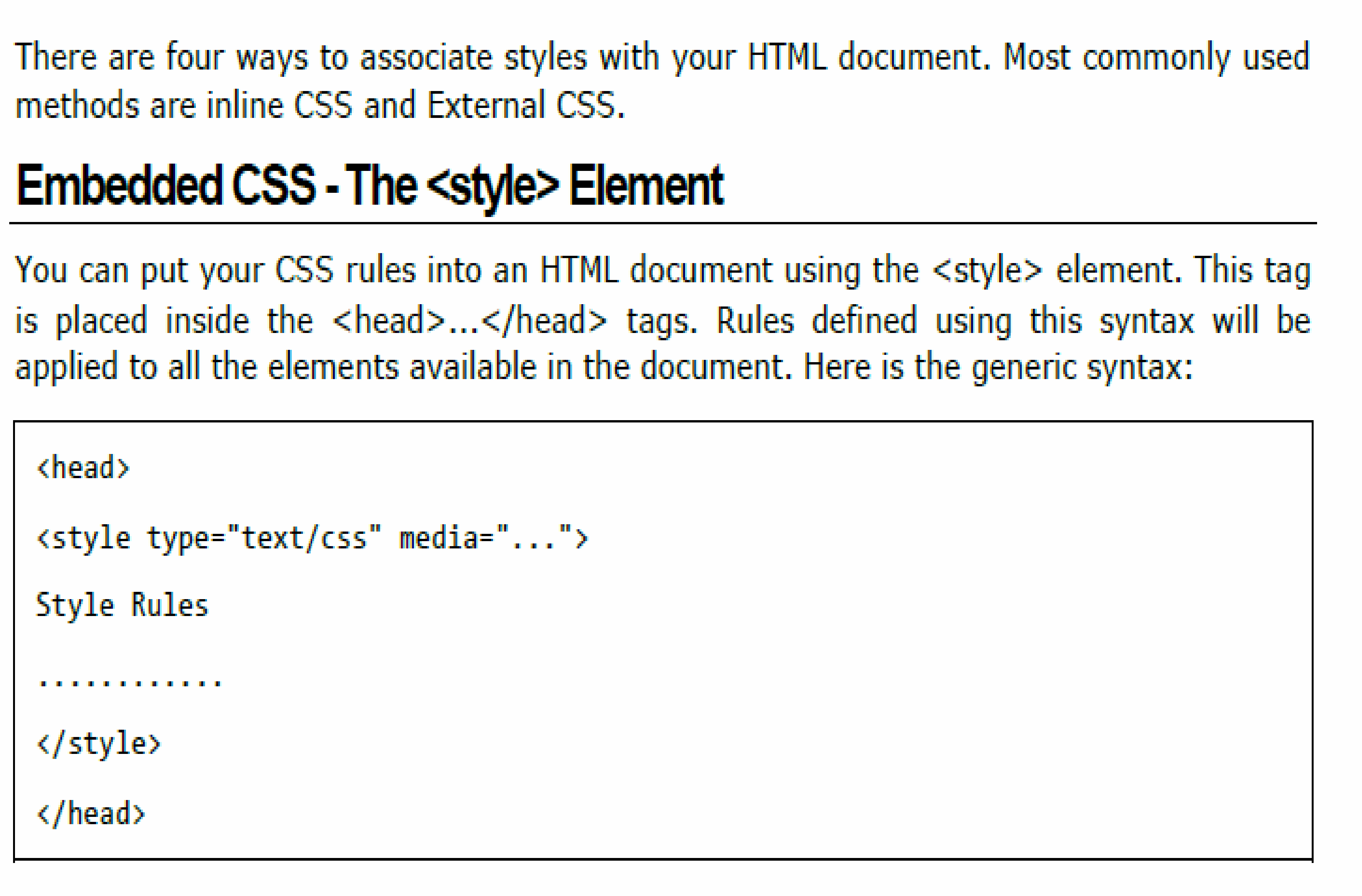
```
h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}
```

This define style rule will be applicable to h1, h2 and h3 element as well. The order of the list is irrelevant. All the elements in the selector will have the corresponding declarations applied to them.

You can combine the various *class* selectors together as shown below:

```
#content, #footer, #supplement {

position: absolute;

left: 510px;

width: 200px;

}
```

CSS — Inclusion



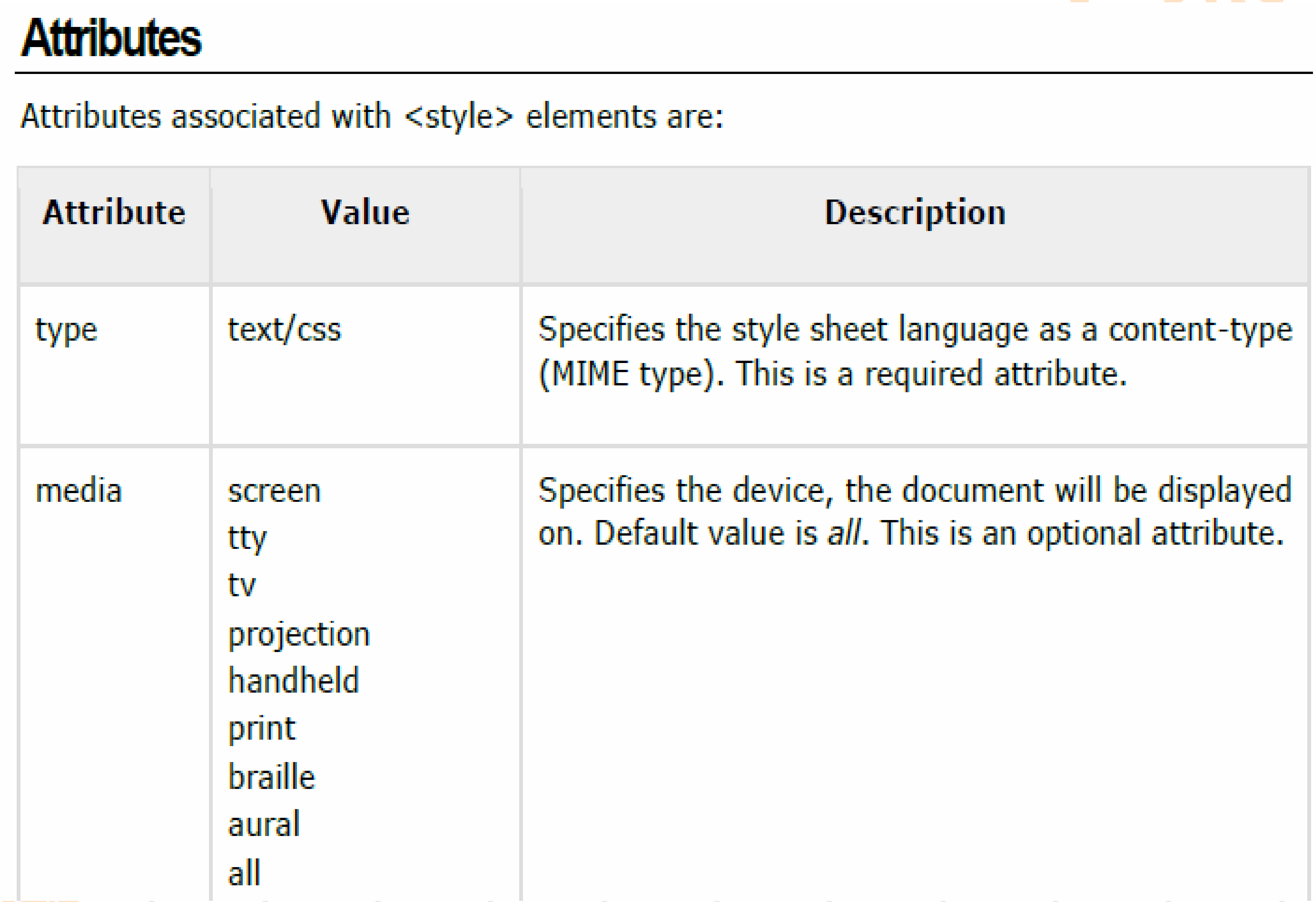
There are four ways to associate styles with your HTML document. Most commonly used methods are inline CSS and External CSS.

Embedded CSS - The <style> Element

You can put your CSS rules into an HTML document using the <style> element. This tag is placed inside the <head>...</head> tags. Rules defined using this syntax will be applied to all the elements available in the document. Here is the generic syntax:

```
<head>

<style type="text/css" media="...">

Style Rules

............

</style>

</head>
```

CSS — Inclusion

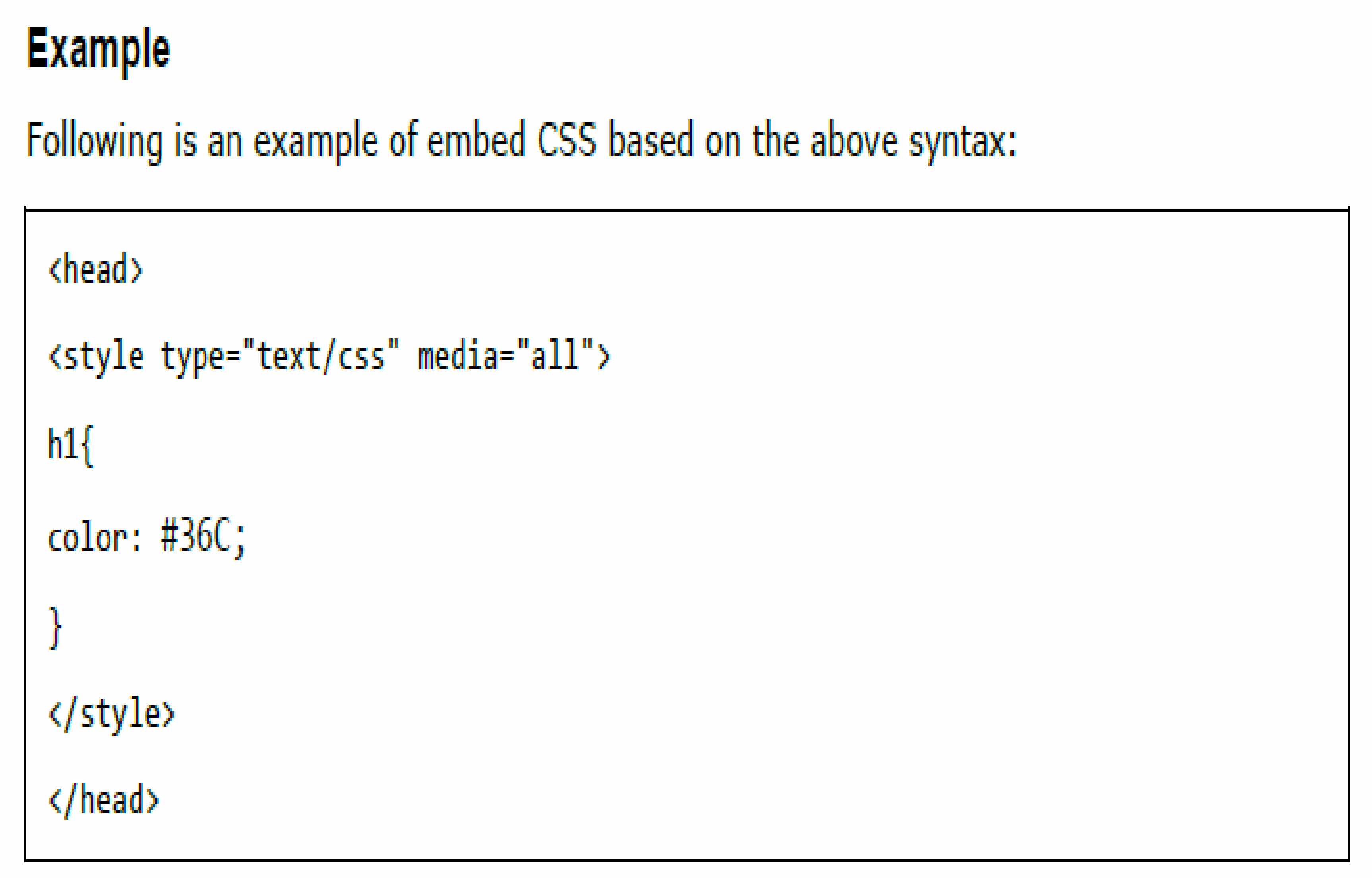


Attributes

Attributes associated with <style> elements are:

Attribute	Value	Description
type	text/css	Specifies the style sheet language as a content-type (MIME type). This is a required attribute.
media	screen tty tv projection handheld print braille aural all	Specifies the device, the document will be displayed on. Default value is *all*. This is an optional attribute.

CSS — Inclusion

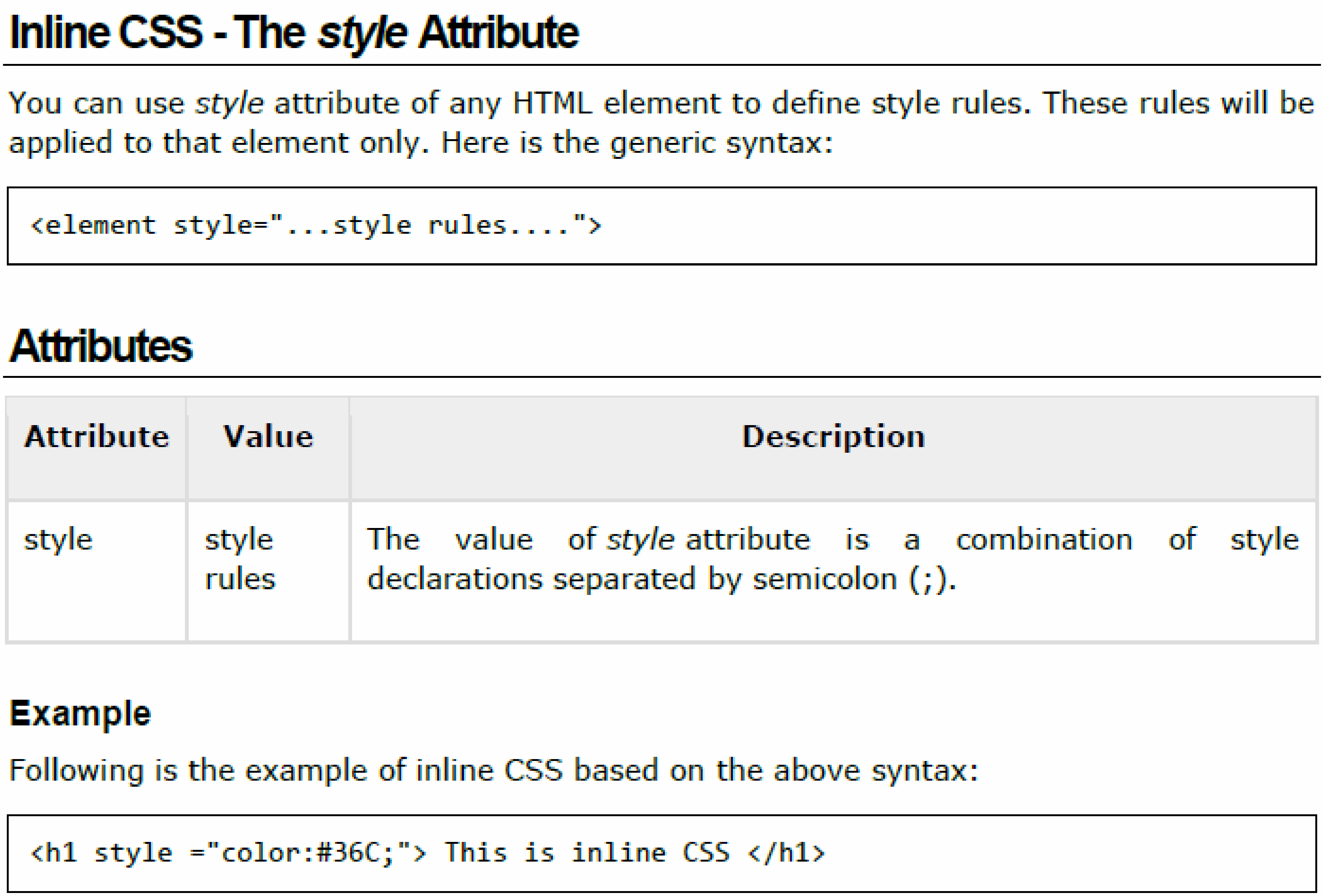


Example

Following is an example of embed CSS based on the above syntax:

```
<head>

<style type="text/css" media="all">

h1{

color: #36C;

}

</style>

</head>
```

CSS — Inclusion



Inline CSS - The *style* Attribute

You can use *style* attribute of any HTML element to define style rules. These rules will be applied to that element only. Here is the generic syntax:

```
<element style="...style rules....">
```

Attributes

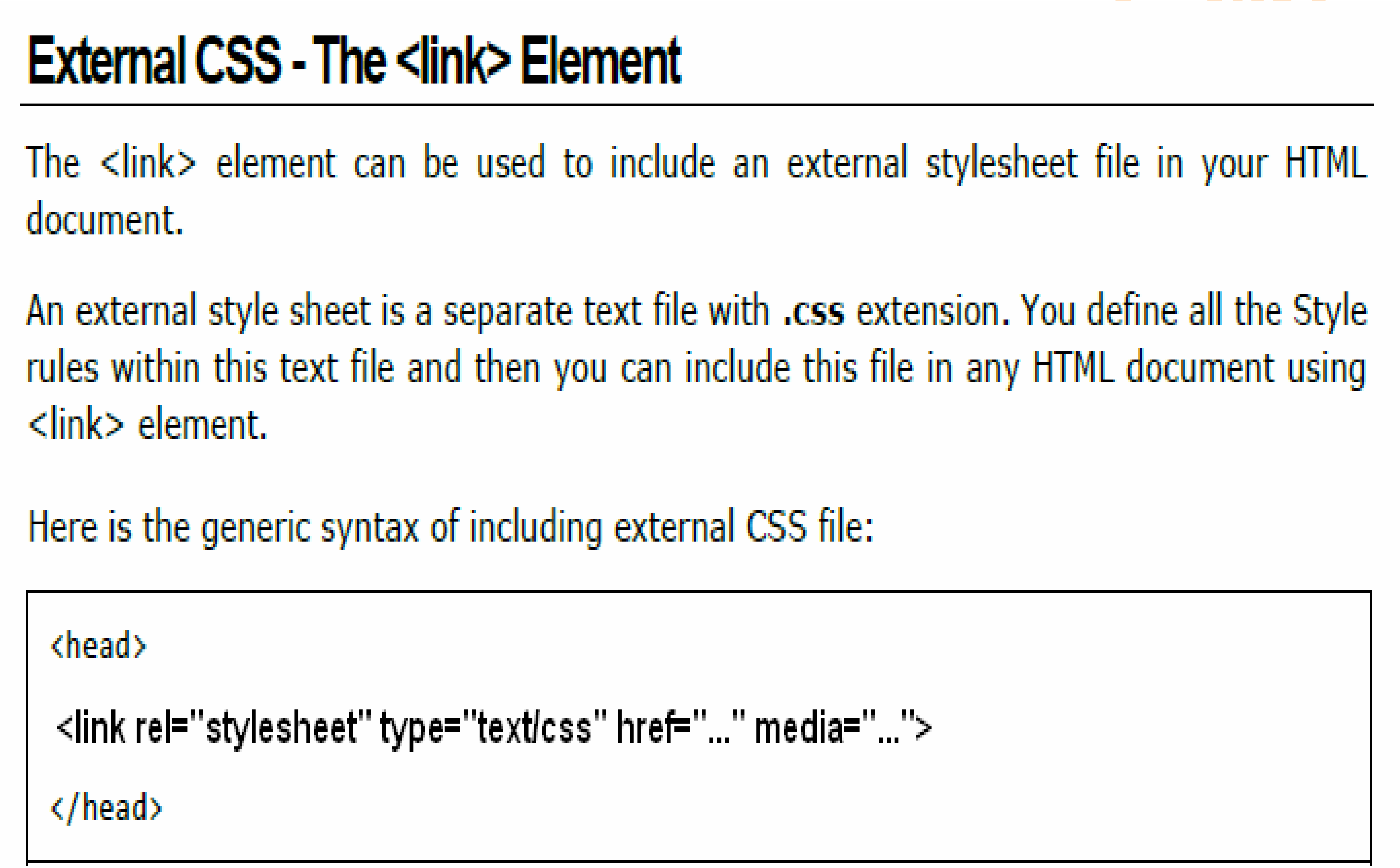
Attribute	Value	Description
style	style rules	The value of *style* attribute is a combination of style declarations separated by semicolon (;).

Example

Following is the example of inline CSS based on the above syntax:

```
<h1 style ="color:#36C;"> This is inline CSS </h1>
```

CSS — Inclusion



External CSS - The <link> Element

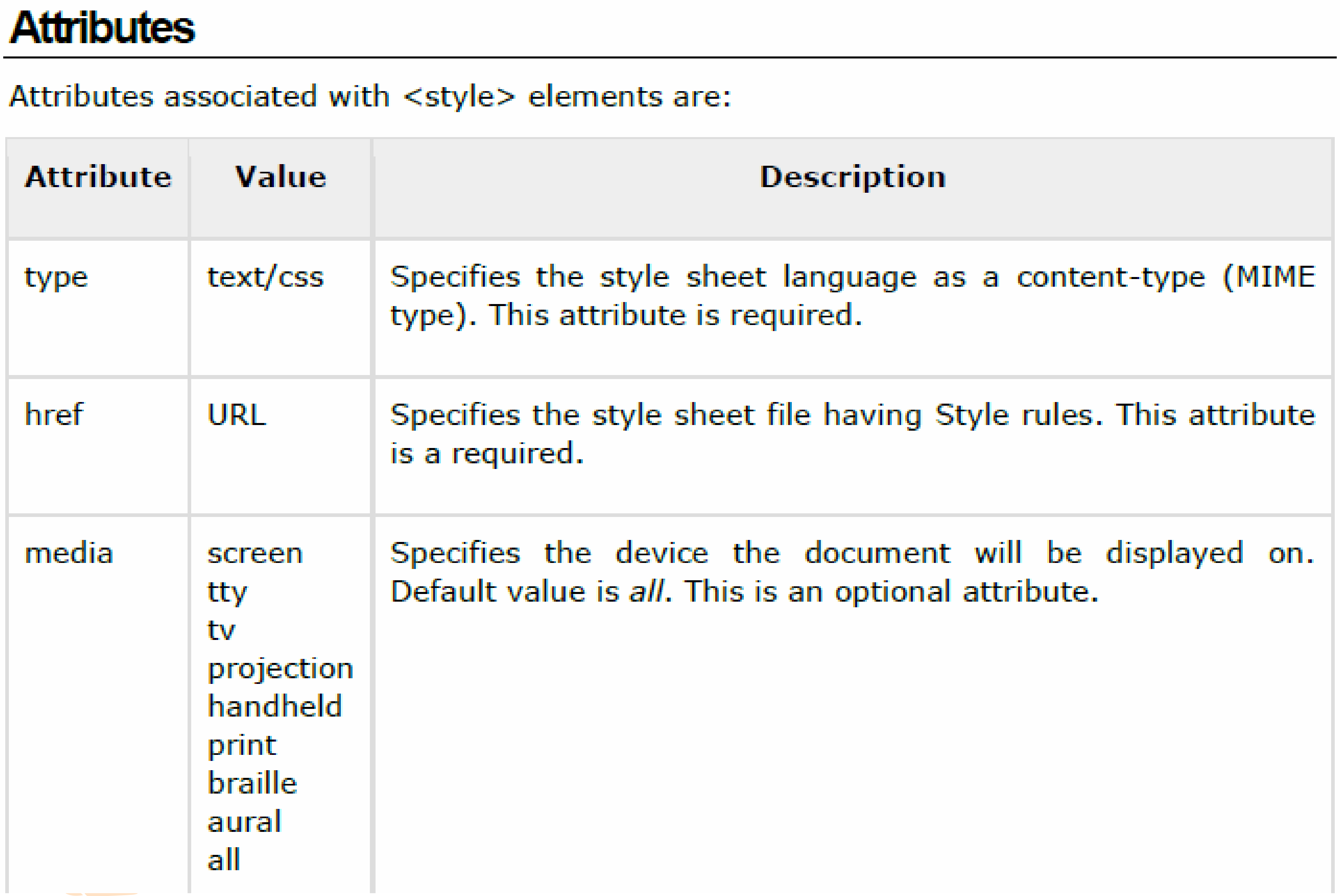
The <link> element can be used to include an external stylesheet file in your HTML document.

An external style sheet is a separate text file with **.css** extension. You define all the Style rules within this text file and then you can include this file in any HTML document using <link> element.

Here is the generic syntax of including external CSS file:

```
<head>
<link rel="stylesheet" type="text/css" href="..." media="...">
</head>
```

CSS — Inclusion

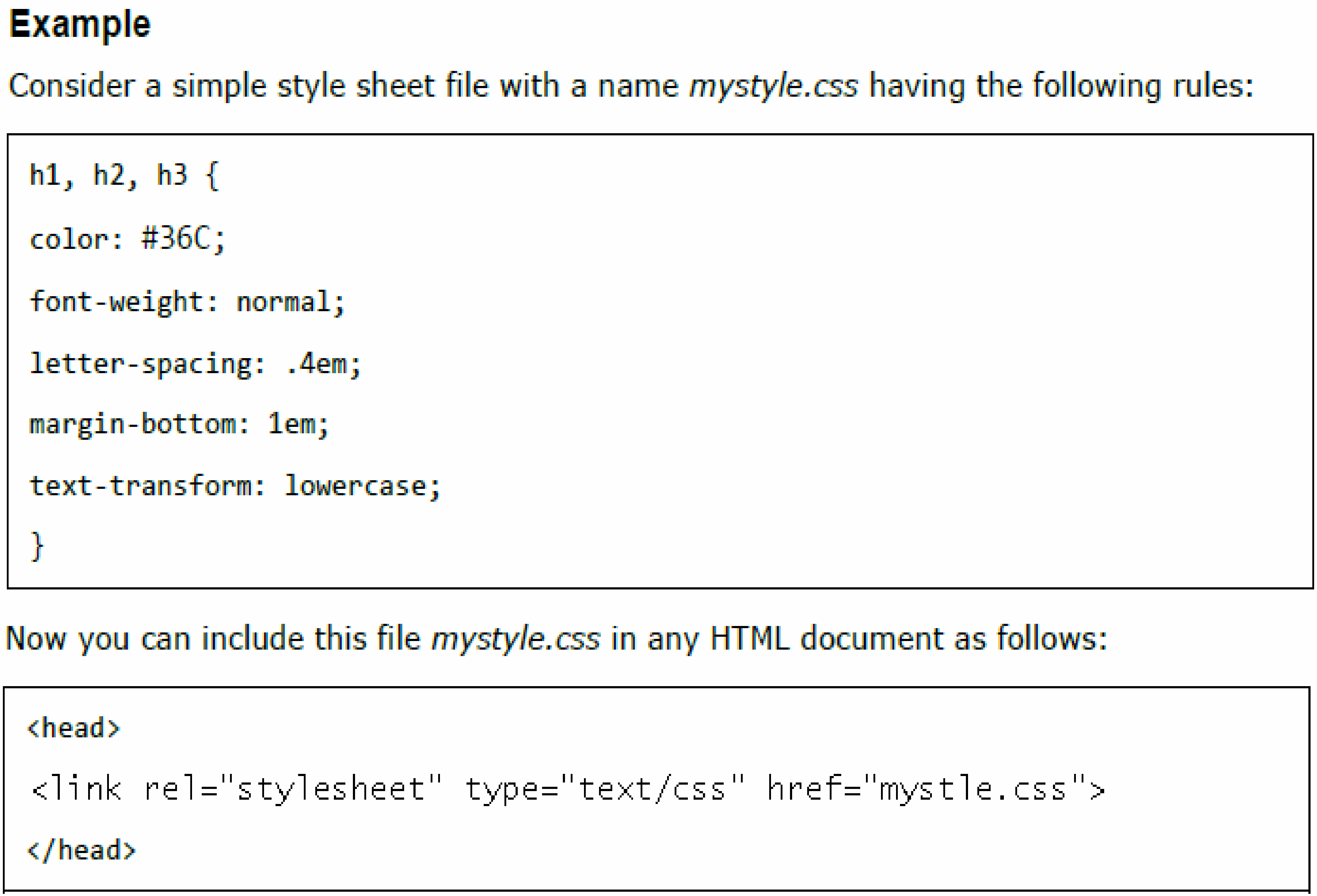


Attributes

Attributes associated with <style> elements are:

Attribute	Value	Description
type	text/css	Specifies the style sheet language as a content-type (MIME type). This attribute is required.
href	URL	Specifies the style sheet file having Style rules. This attribute is a required.
media	screen tty tv projection handheld print braille aural all	Specifies the device the document will be displayed on. Default value is *all*. This is an optional attribute.

CSS — Inclusion



Example

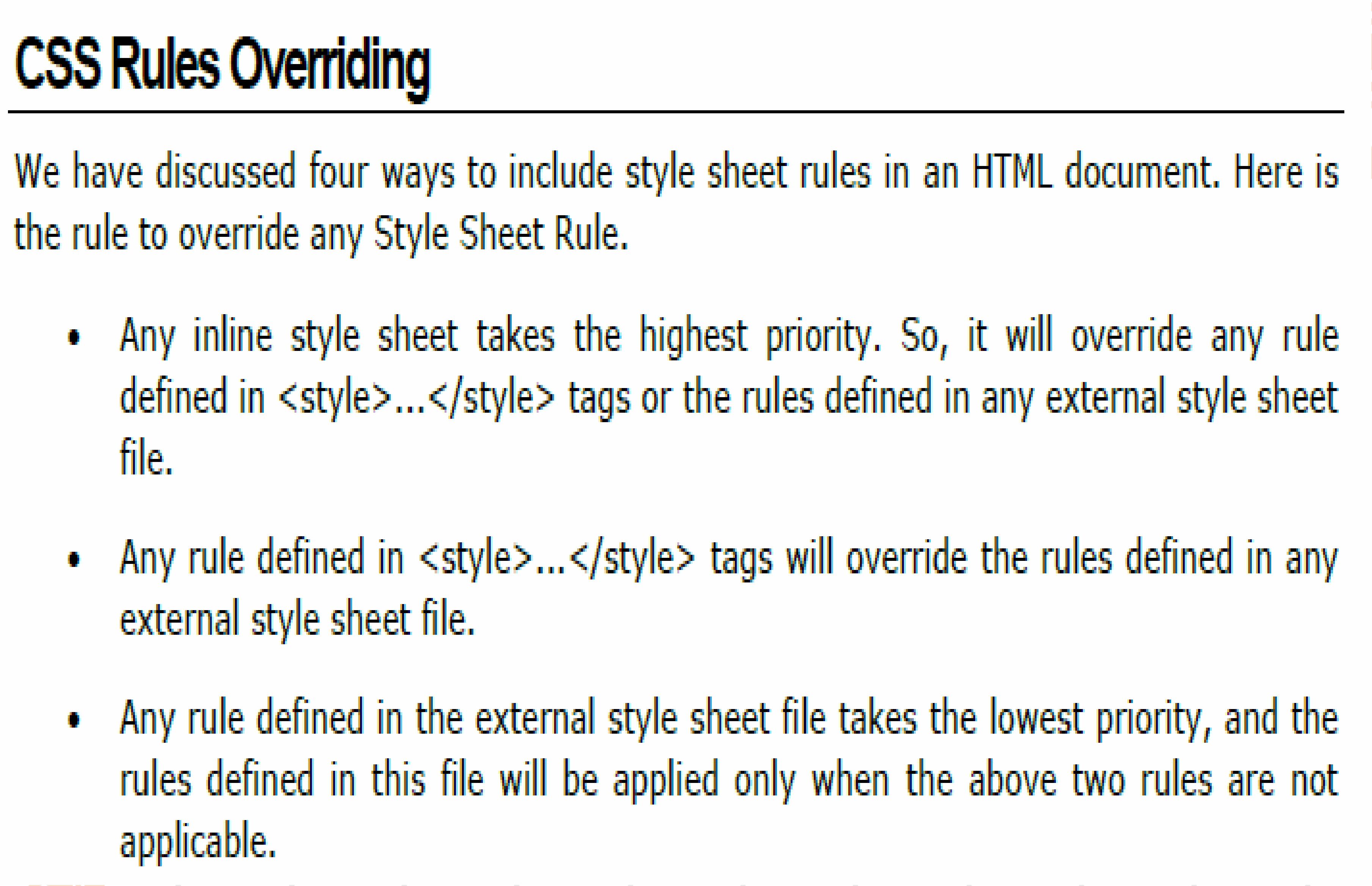
Consider a simple style sheet file with a name *mystyle.css* having the following rules:

```
h1, h2, h3 {

color: #36C;

font-weight: normal;

letter-spacing: .4em;

margin-bottom: 1em;

text-transform: lowercase;

}
```

Now you can include this file *mystyle.css* in any HTML document as follows:

```
<head>

<link rel="stylesheet" type="text/css" href="mystle.css">

</head>
```

CSS — Inclusion

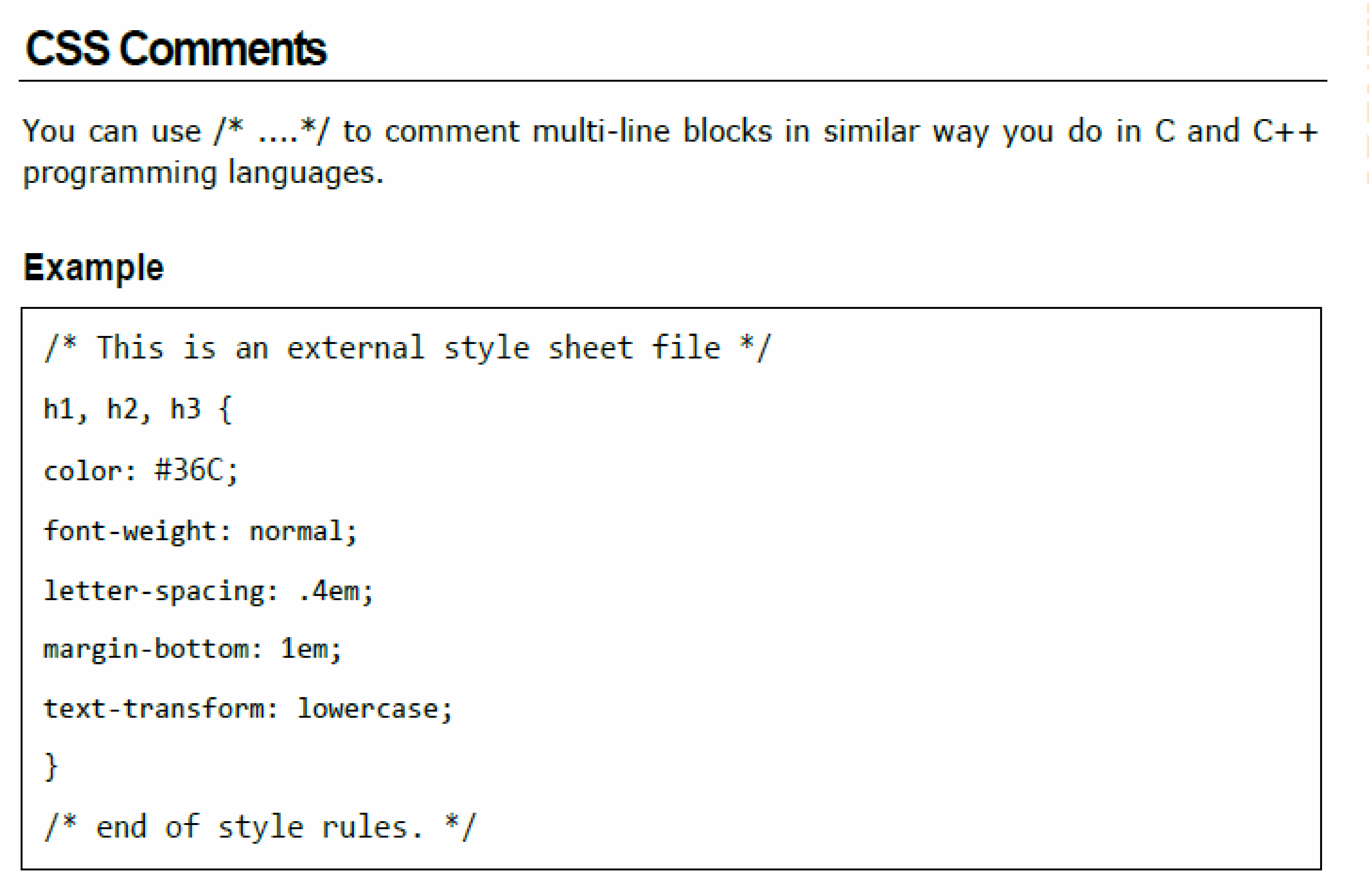


CSS Rules Overriding

We have discussed four ways to include style sheet rules in an HTML document. Here is the rule to override any Style Sheet Rule.

- Any inline style sheet takes the highest priority. So, it will override any rule defined in <style>...</style> tags or the rules defined in any external style sheet file.
- Any rule defined in <style>...</style> tags will override the rules defined in any external style sheet file.
- Any rule defined in the external style sheet file takes the lowest priority, and the rules defined in this file will be applied only when the above two rules are not applicable.

CSS — Inclusion



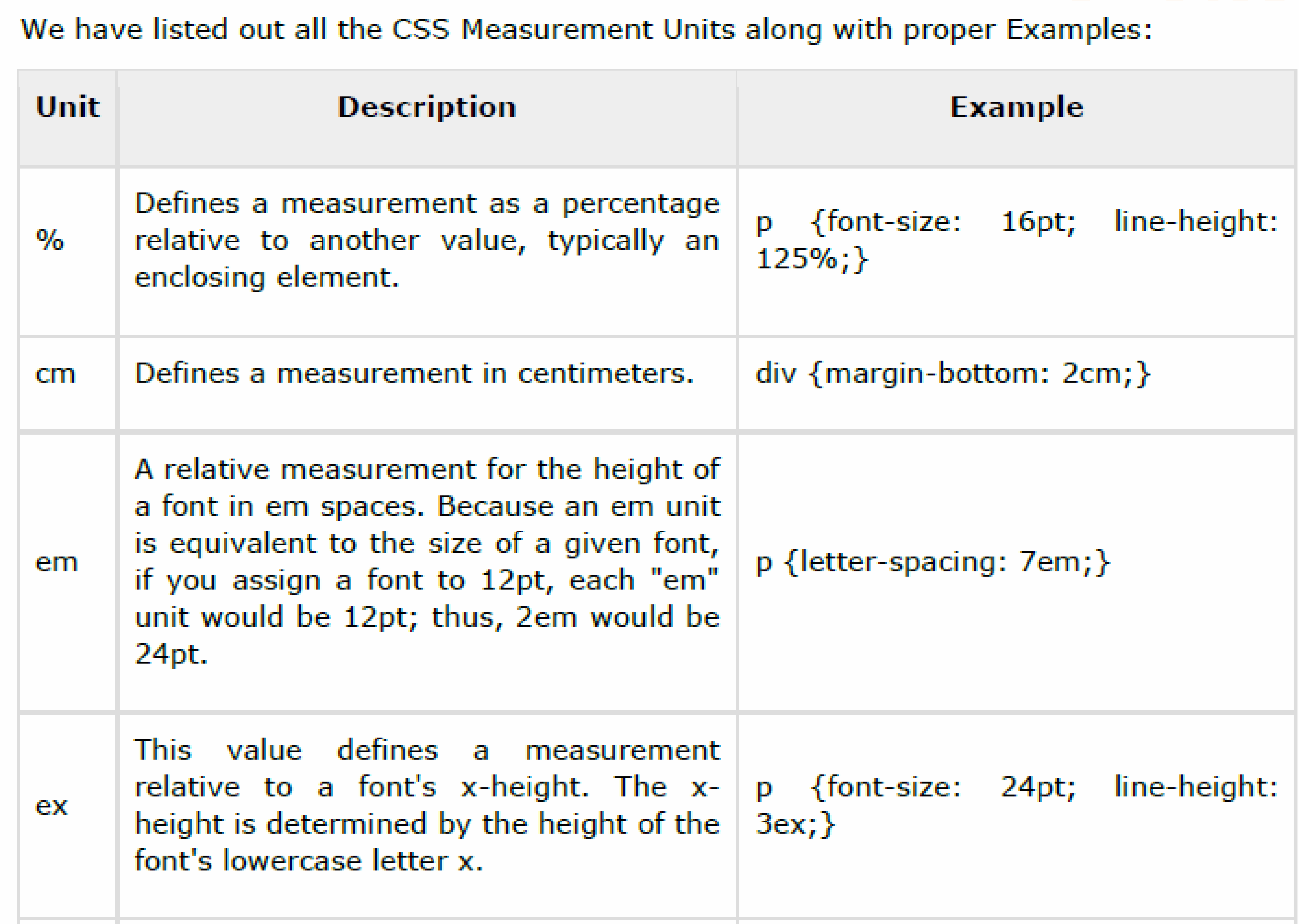
CSS Comments

You can use /* ....*/ to comment multi-line blocks in similar way you do in C and C++ programming languages.

Example

```
/* This is an external style sheet file */
h1, h2, h3 {
color: #36C;
font-weight: normal;
letter-spacing: .4em;
margin-bottom: 1em;
text-transform: lowercase;
}
/* end of style rules. */
```

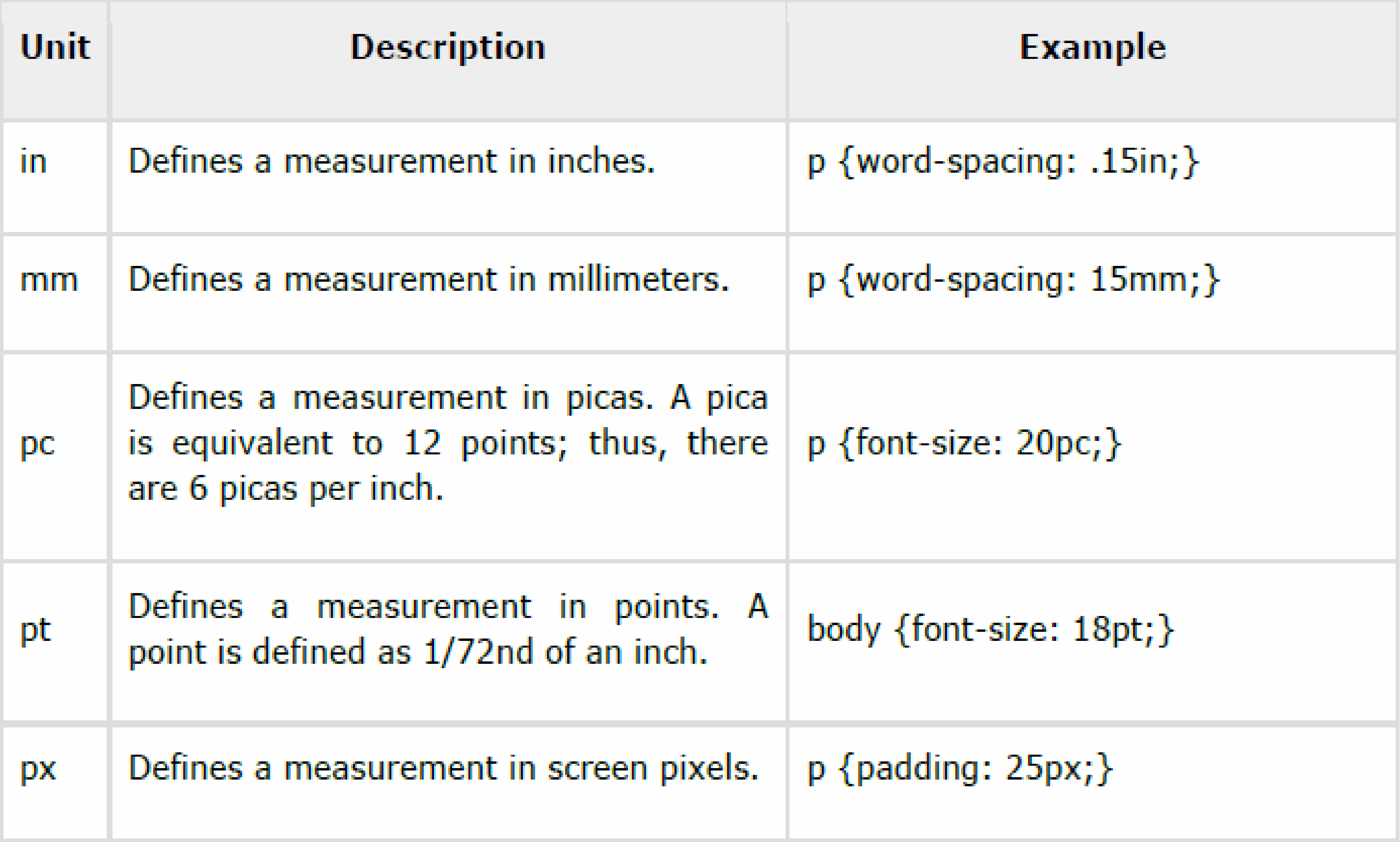
CSS — Measurement Units



We have listed out all the CSS Measurement Units along with proper Examples:

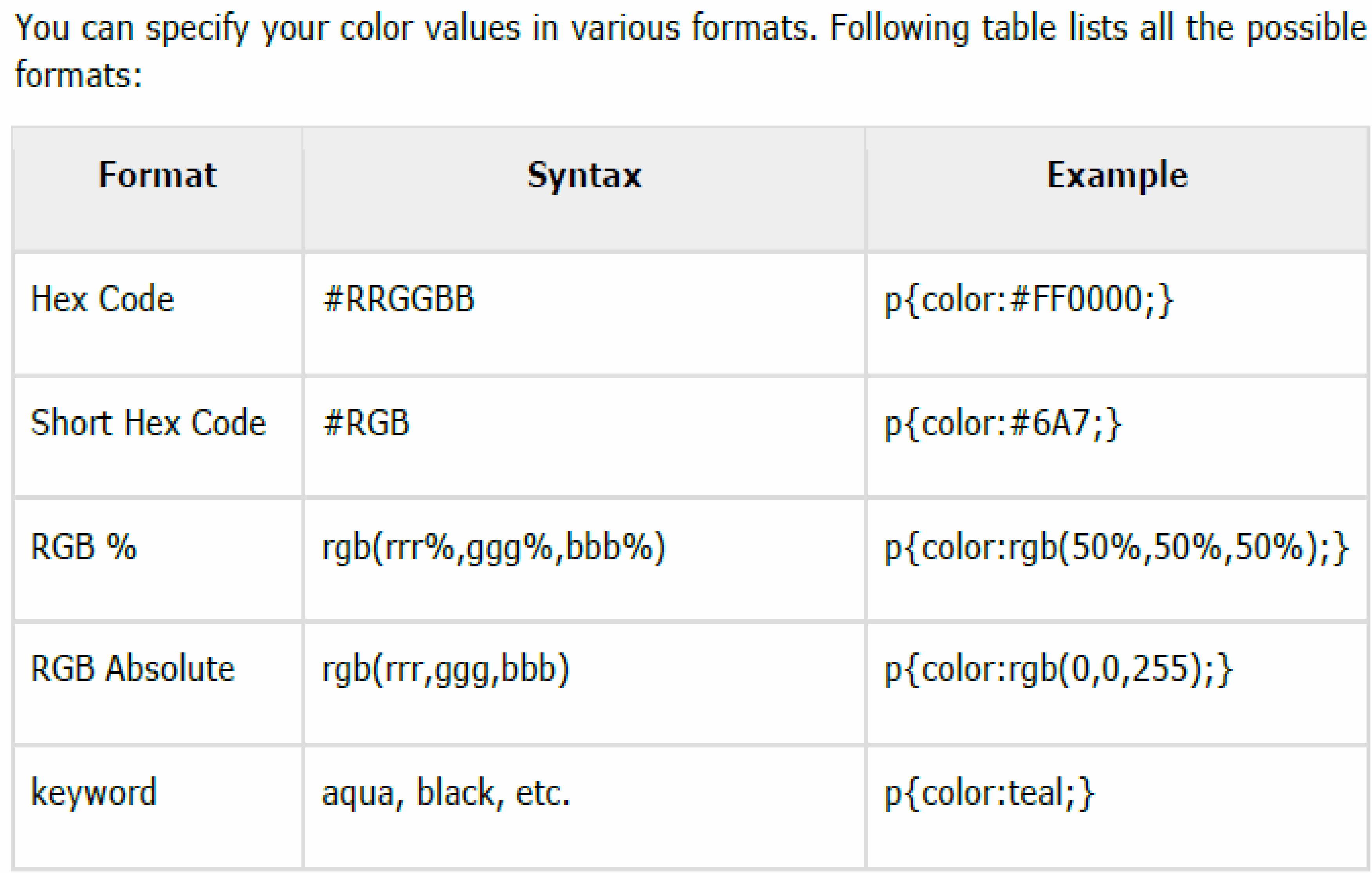
Unit	Description	Example
%	Defines a measurement as a percentage relative to another value, typically an enclosing element.	p {font-size: 16pt; line-height: 125%;}
cm	Defines a measurement in centimeters.	div {margin-bottom: 2cm;}
em	A relative measurement for the height of a font in em spaces. Because an em unit is equivalent to the size of a given font, if you assign a font to 12pt, each "em" unit would be 12pt; thus, 2em would be 24pt.	p {letter-spacing: 7em;}
ex	This value defines a measurement relative to a font's x-height. The x-height is determined by the height of the font's lowercase letter x.	p {font-size: 24pt; line-height: 3ex;}

CSS — Measurement Units



Unit	Description	Example
in	Defines a measurement in inches.	p {word-spacing: .15in;}
mm	Defines a measurement in millimeters.	p {word-spacing: 15mm;}
pc	Defines a measurement in picas. A pica is equivalent to 12 points; thus, there are 6 picas per inch.	p {font-size: 20pc;}
pt	Defines a measurement in points. A point is defined as 1/72nd of an inch.	body {font-size: 18pt;}
px	Defines a measurement in screen pixels.	p {padding: 25px;}

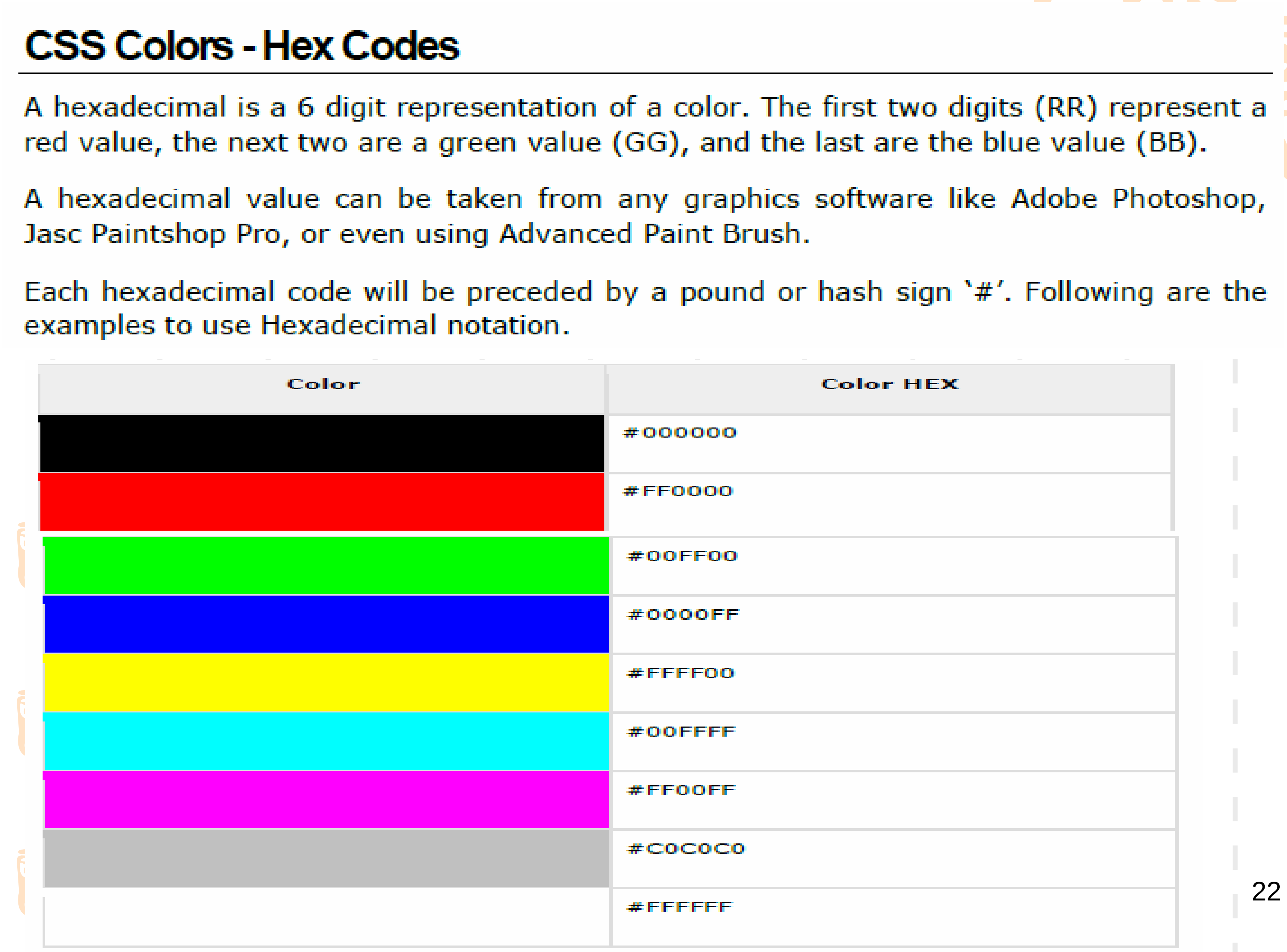
CSS — Colors



You can specify your color values in various formats. Following table lists all the possible formats:

Format	Syntax	Example
Hex Code	#RRGGBB	p{color:#FF0000;}
Short Hex Code	#RGB	p{color:#6A7;}
RGB %	rgb(rrr%,ggg%,bbb%)	p{color:rgb(50%,50%,50%);}
RGB Absolute	rgb(rrr,ggg,bbb)	p{color:rgb(0,0,255);}
keyword	aqua, black, etc.	p{color:teal;}

CSS — Colors



CSS Colors - Hex Codes

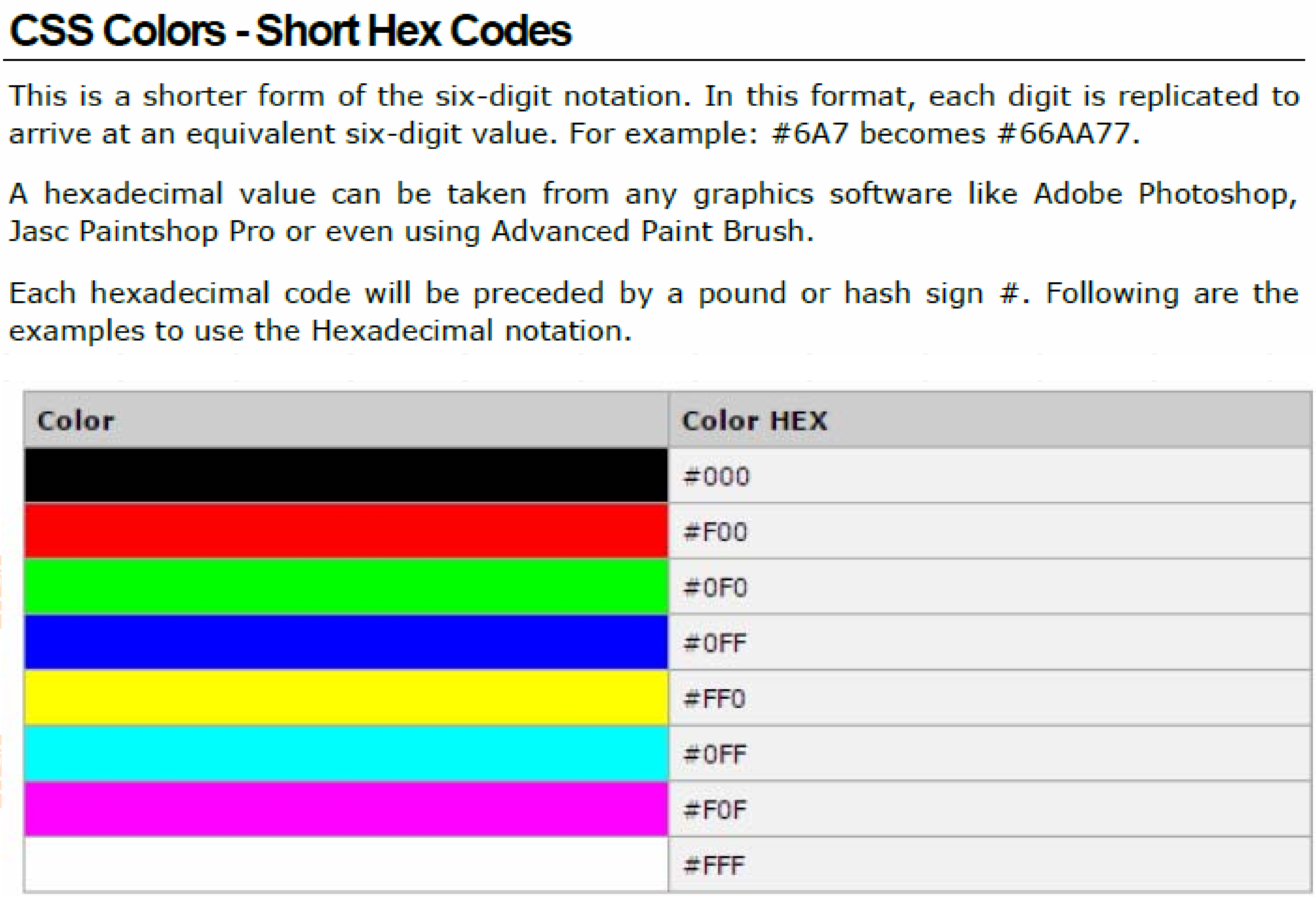
A hexadecimal is a 6 digit representation of a color. The first two digits (RR) represent a red value, the next two are a green value (GG), and the last are the blue value (BB).

A hexadecimal value can be taken from any graphics software like Adobe Photoshop, Jasc Paintshop Pro, or even using Advanced Paint Brush.

Each hexadecimal code will be preceded by a pound or hash sign '#'. Following are the examples to use Hexadecimal notation.

Color	Color HEX
⬛	#000000
🟥	#FF0000
🟩	#00FF00
🟦	#0000FF
🟨	#FFFF00
■	#00FFFF
🟪	#FF00FF
▨	#C0C0C0
⬜	#FFFFFF

CSS — Colors



CSS Colors - Short Hex Codes

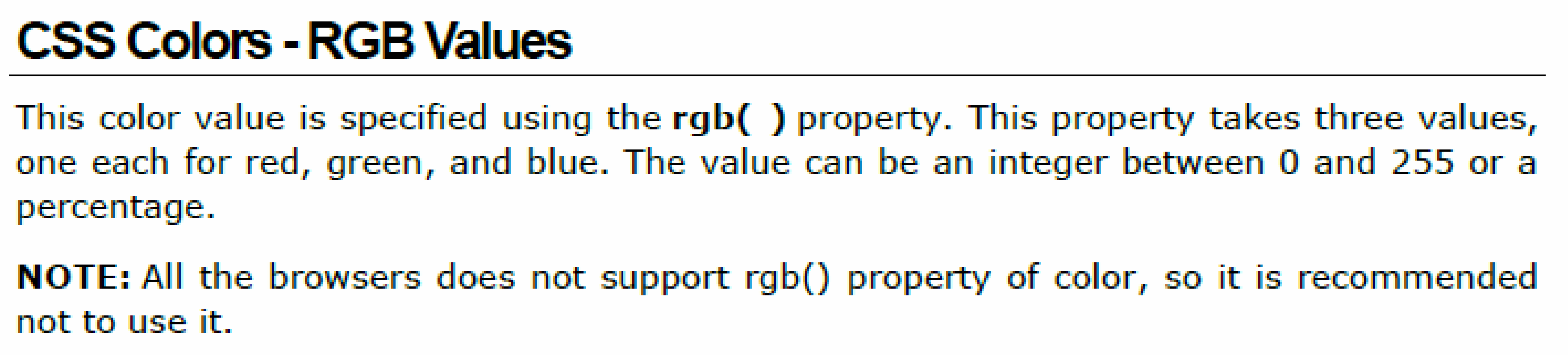
This is a shorter form of the six-digit notation. In this format, each digit is replicated to arrive at an equivalent six-digit value. For example: #6A7 becomes #66AA77.

A hexadecimal value can be taken from any graphics software like Adobe Photoshop, Jasc Paintshop Pro or even using Advanced Paint Brush.

Each hexadecimal code will be preceded by a pound or hash sign #. Following are the examples to use the Hexadecimal notation.

Color	Color HEX
⬛	#000
🟥	#F00
🟩	#0F0
🟦	#0FF
🟨	#FF0
🟦	#0FF
🟪	#F0F
⬜	#FFF

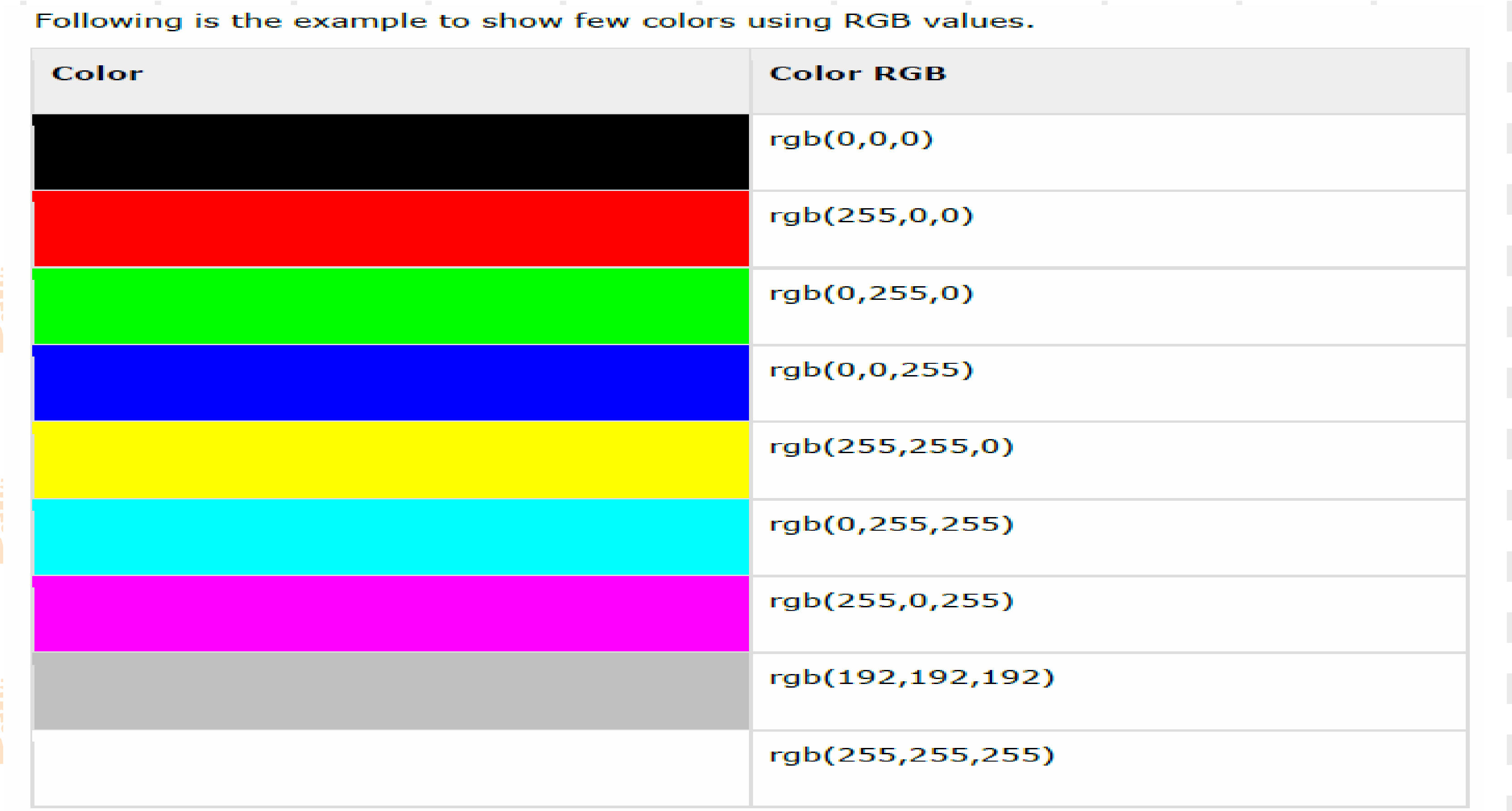
CSS — Colors



CSS Colors - RGB Values

This color value is specified using the **rgb( )** property. This property takes three values, one each for red, green, and blue. The value can be an integer between 0 and 255 or a percentage.

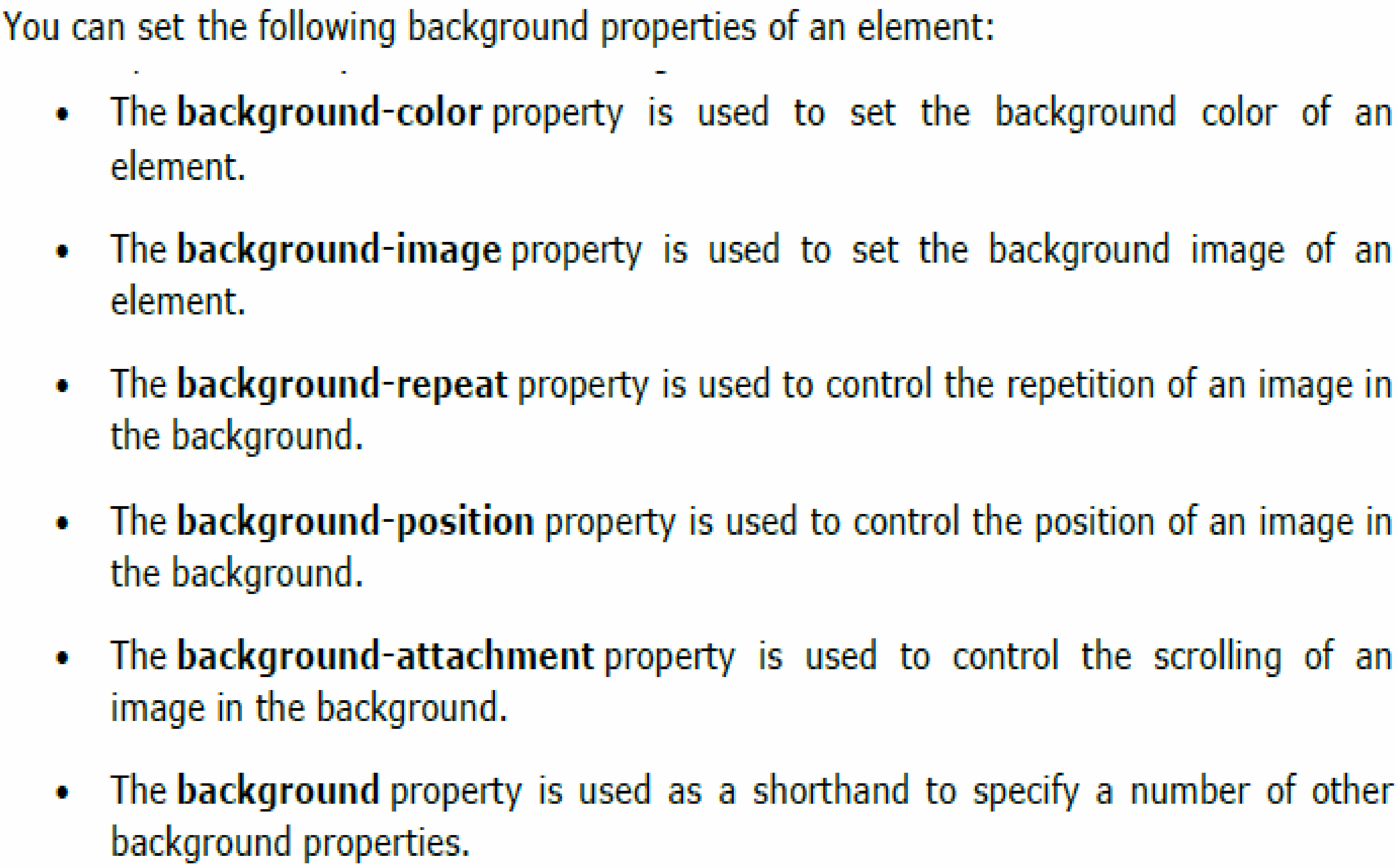
NOTE: All the browsers does not support rgb() property of color, so it is recommended not to use it.



Following is the example to show few colors using RGB values.

Color	Color RGB
⬛	rgb(0,0,0)
🟥	rgb(255,0,0)
🟩	rgb(0,255,0)
🟦	rgb(0,0,255)
🟨	rgb(255,255,0)
🟦	rgb(0,255,255)
🟪	rgb(255,0,255)
▨	rgb(192,192,192)
⬜	rgb(255,255,255)

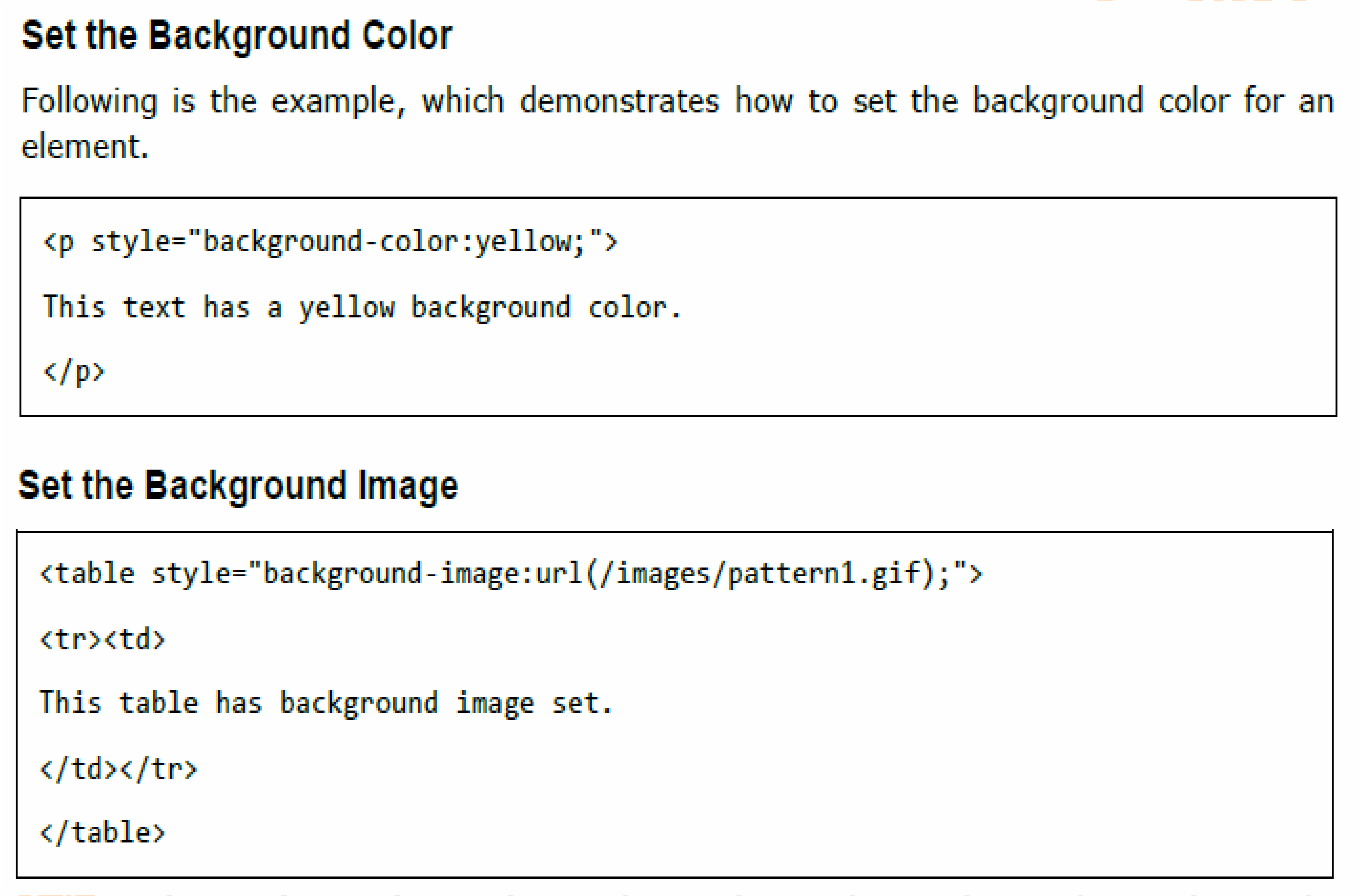
CSS — Background



You can set the following background properties of an element:

- The **background-color** property is used to set the background color of an element.
- The **background-image** property is used to set the background image of an element.
- The **background-repeat** property is used to control the repetition of an image in the background.
- The **background-position** property is used to control the position of an image in the background.
- The **background-attachment** property is used to control the scrolling of an image in the background.
- The **background** property is used as a shorthand to specify a number of other background properties.

CSS — Background



Set the Background Color

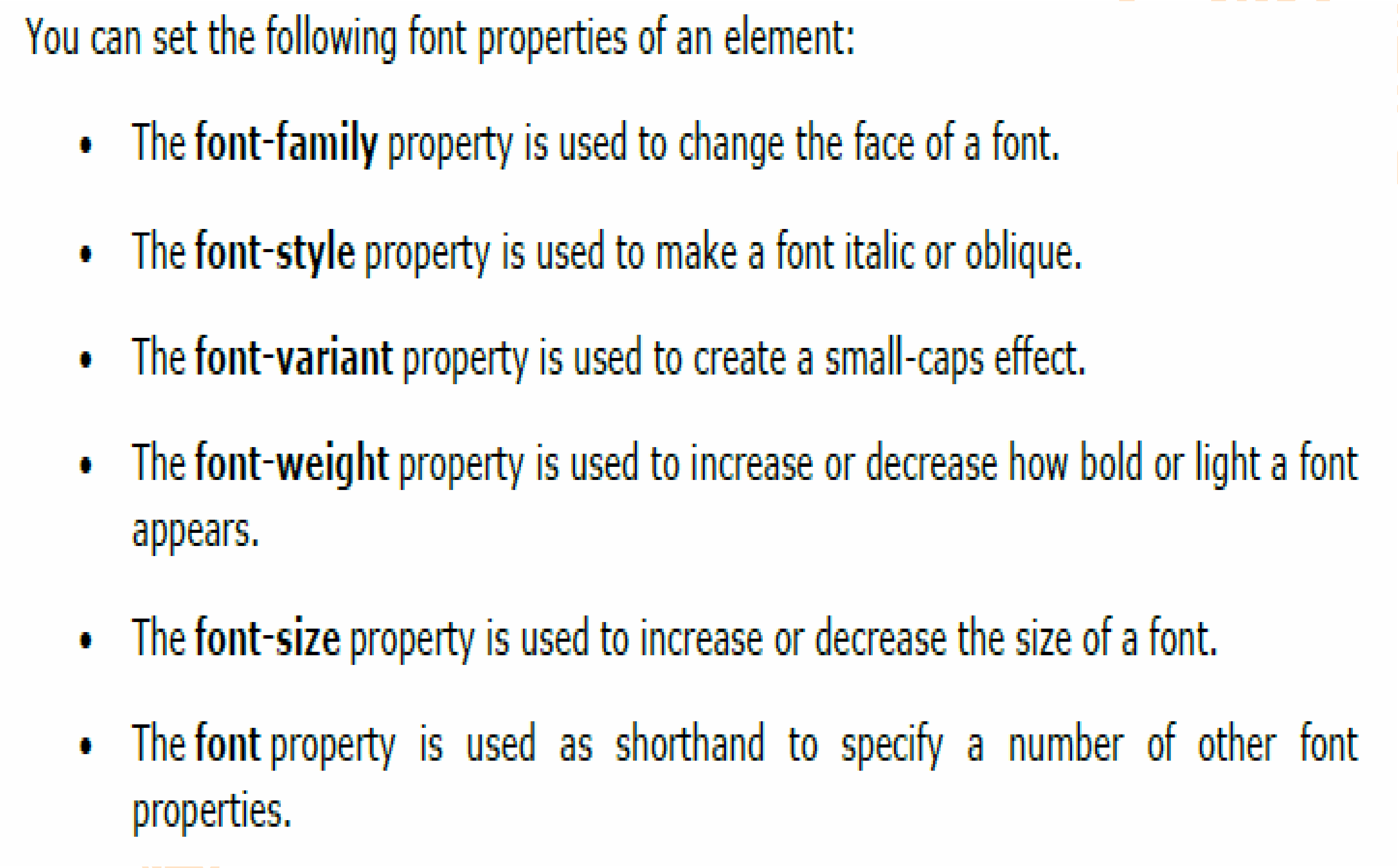
Following is the example, which demonstrates how to set the background color for an element.

```
<p style="background-color:yellow;">

This text has a yellow background color.

</p>
```

Set the Background Image

```
<table style="background-image:url(/images/pattern1.gif);">

<tr><td>

This table has background image set.

</td></tr>

</table>
```

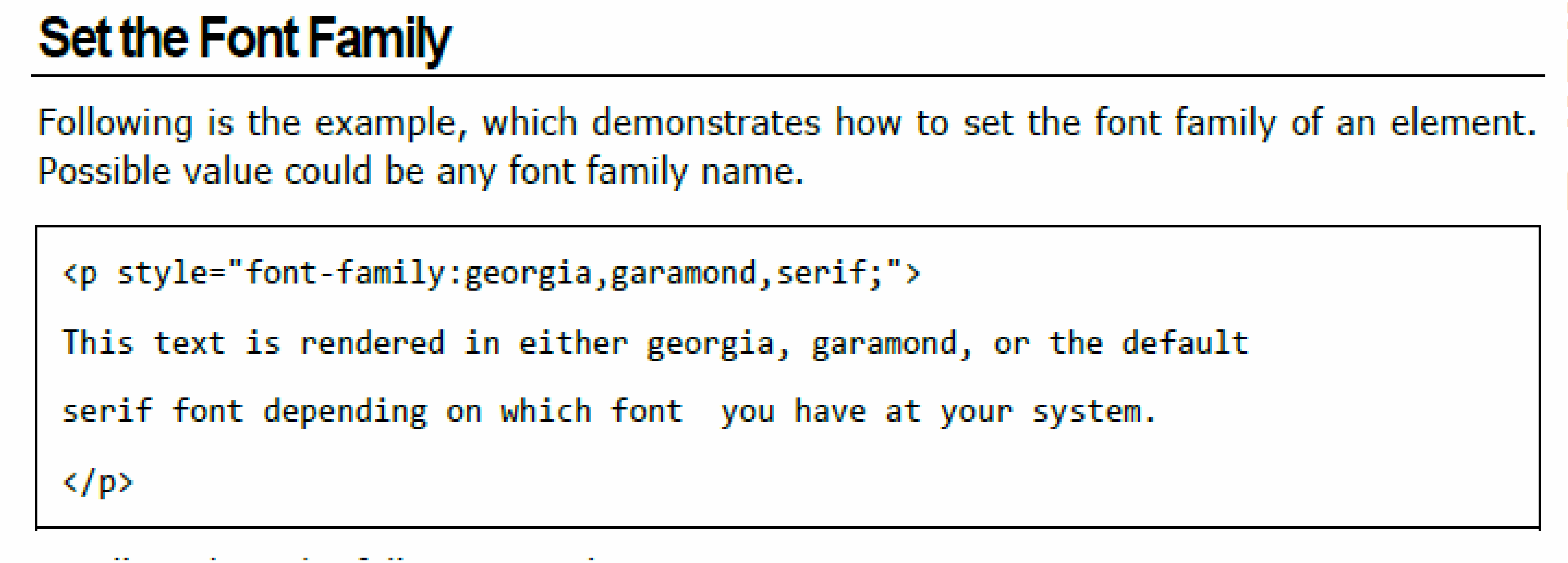
CSS — Fonts



You can set the following font properties of an element:

- The **font-family** property is used to change the face of a font.
- The **font-style** property is used to make a font italic or oblique.
- The **font-variant** property is used to create a small-caps effect.
- The **font-weight** property is used to increase or decrease how bold or light a font appears.
- The **font-size** property is used to increase or decrease the size of a font.
- The **font** property is used as shorthand to specify a number of other font properties.

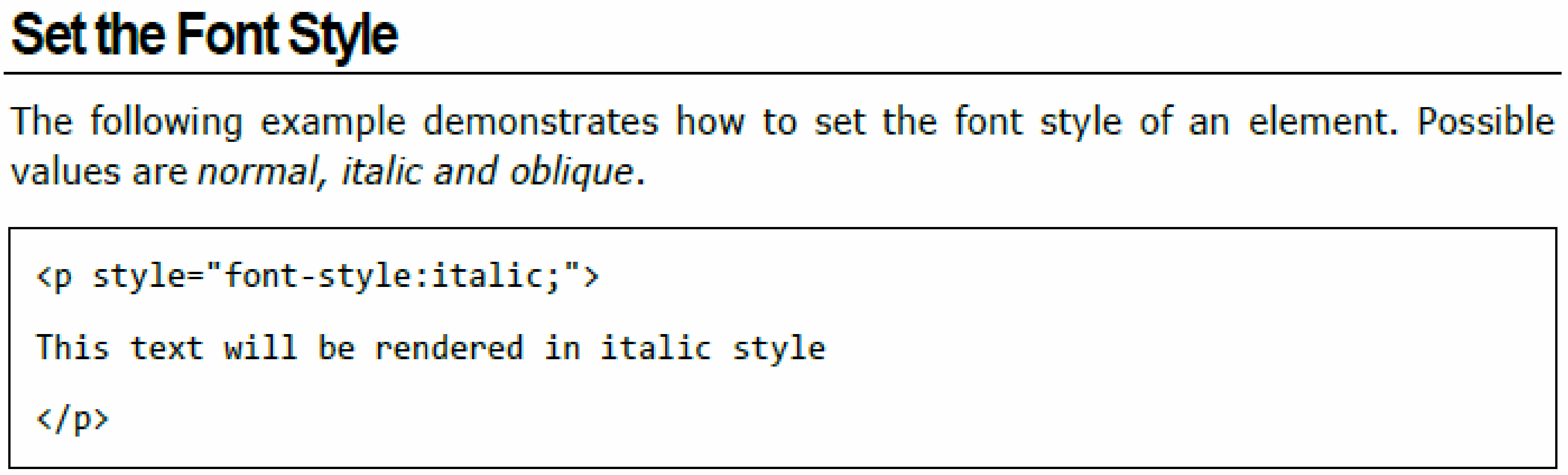
CSS — Fonts



Set the Font Family

Following is the example, which demonstrates how to set the font family of an element. Possible value could be any font family name.

```
<p style="font-family:georgia,garamond,serif;">

This text is rendered in either georgia, garamond, or the default

serif font depending on which font  you have at your system.

</p>
```

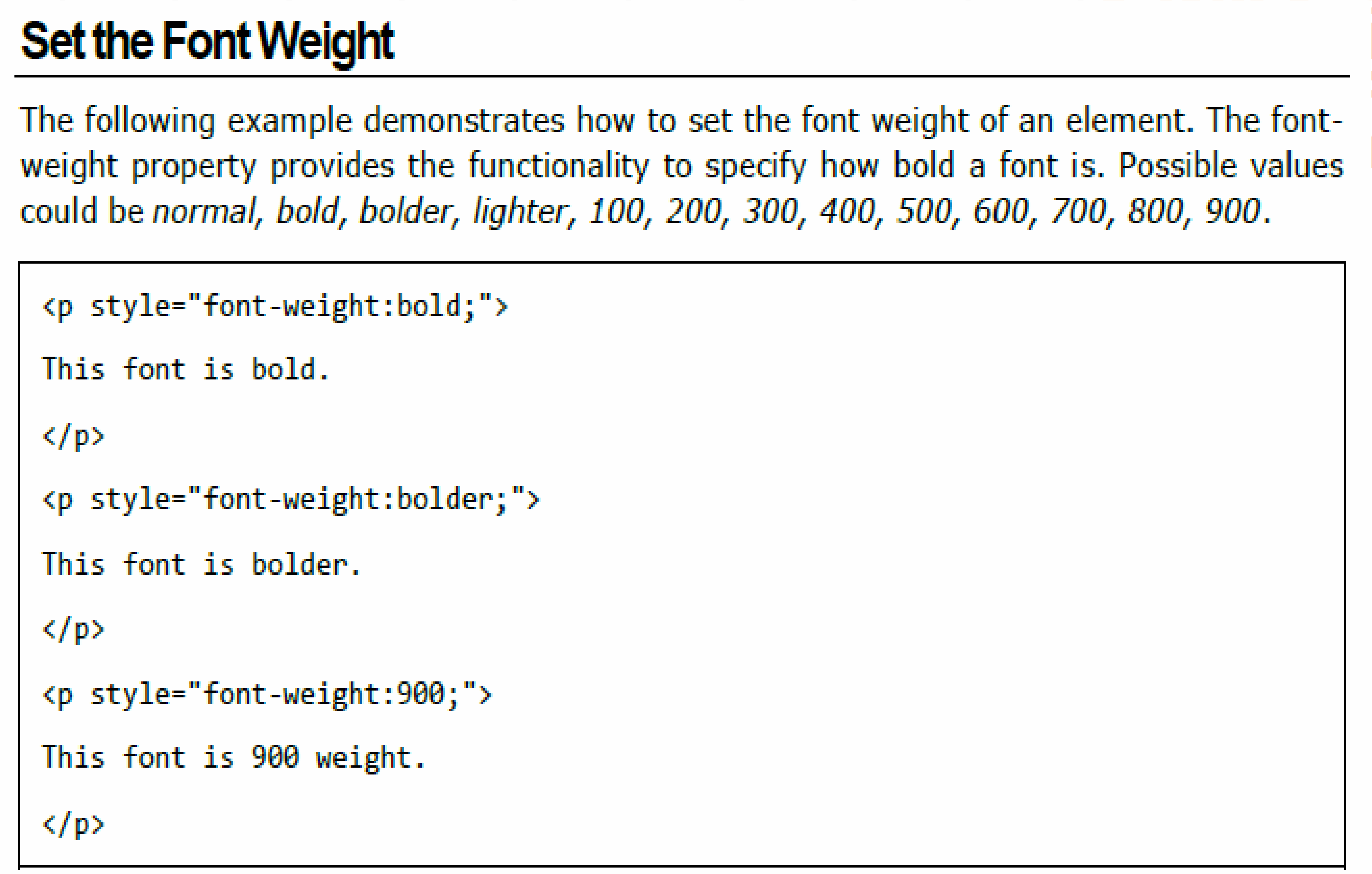


Set the Font Style

The following example demonstrates how to set the font style of an element. Possible values are *normal, italic and oblique*.

```
<p style="font-style:italic;">

This text will be rendered in italic style

</p>
```

CSS — Fonts

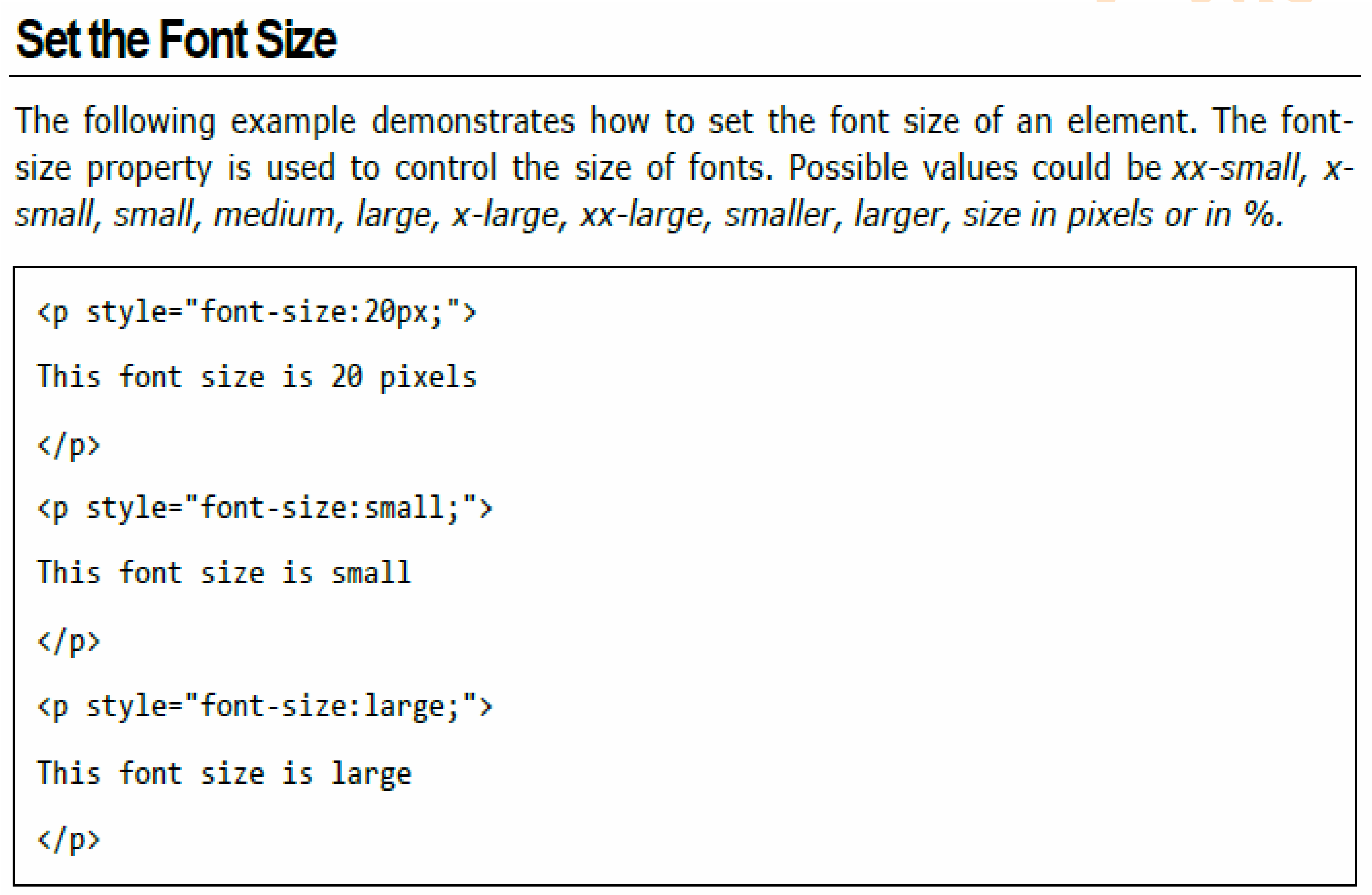


Set the Font Weight

The following example demonstrates how to set the font weight of an element. The font-weight property provides the functionality to specify how bold a font is. Possible values could be *normal, bold, bolder, lighter, 100, 200, 300, 400, 500, 600, 700, 800, 900*.

```
<p style="font-weight:bold;">

This font is bold.

</p>

<p style="font-weight:bolder;">

This font is bolder.

</p>

<p style="font-weight:900;">

This font is 900 weight.

</p>
```

CSS — Fonts

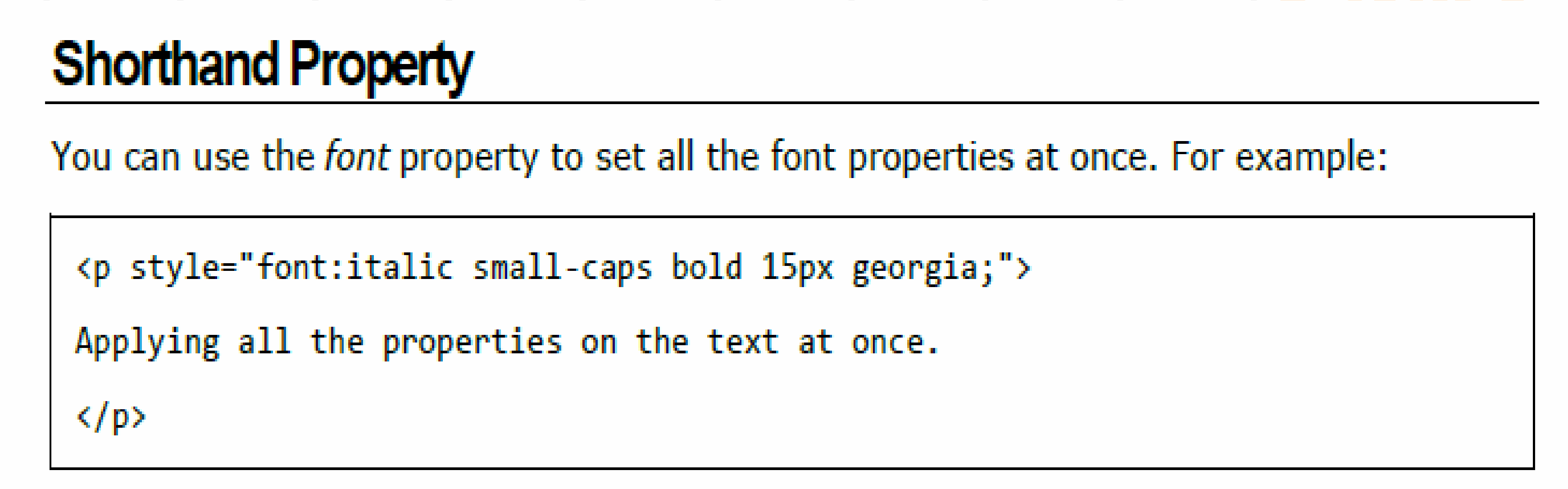


Set the Font Size

The following example demonstrates how to set the font size of an element. The font-size property is used to control the size of fonts. Possible values could be *xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger, size in pixels or in %.*

```
<p style="font-size:20px;">

This font size is 20 pixels

</p>

<p style="font-size:small;">

This font size is small

</p>

<p style="font-size:large;">

This font size is large

</p>
```

CSS — Fonts

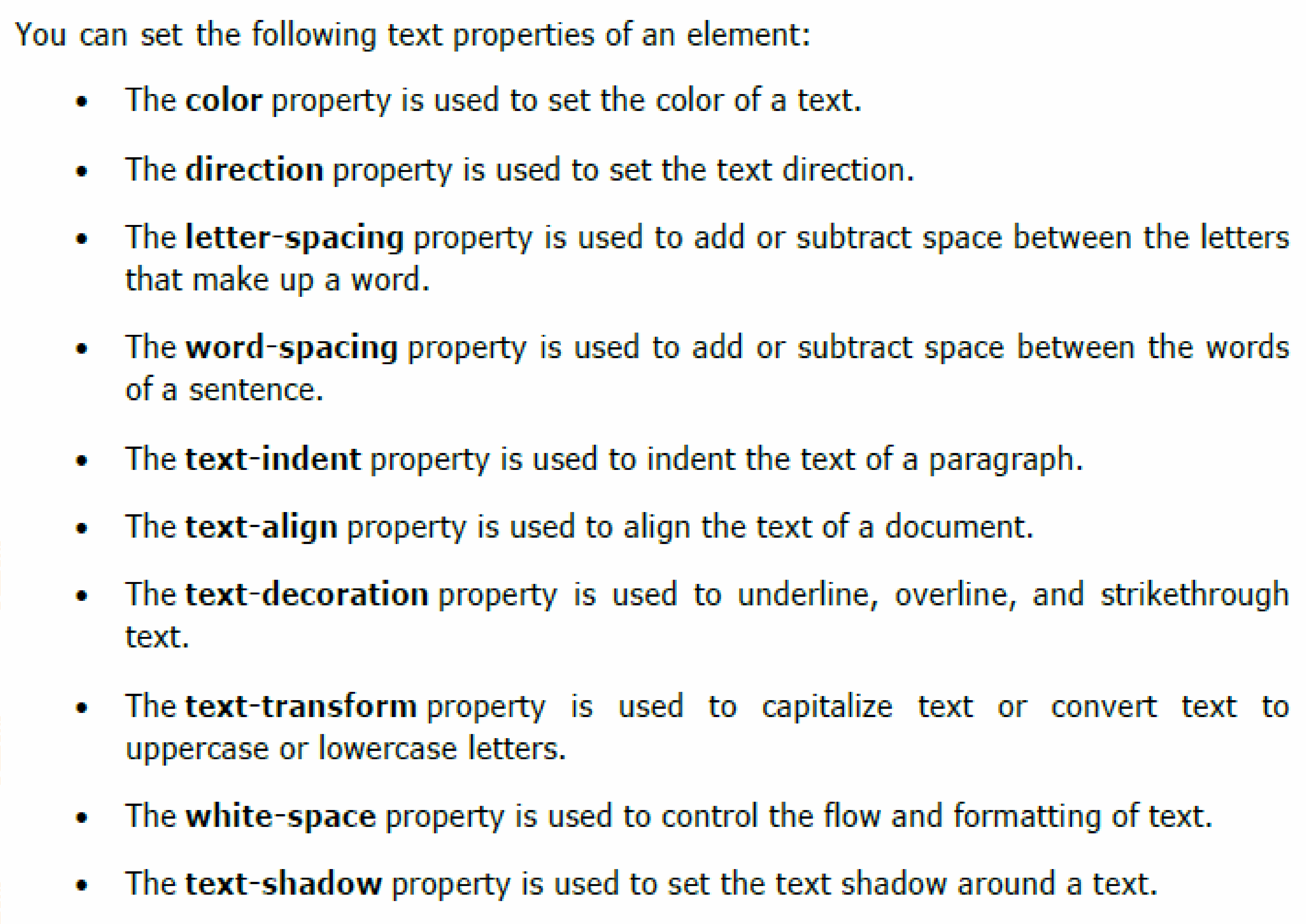


Shorthand Property

You can use the *font* property to set all the font properties at once. For example:

```
<p style="font:italic small-caps bold 15px georgia;">

Applying all the properties on the text at once.

</p>
```

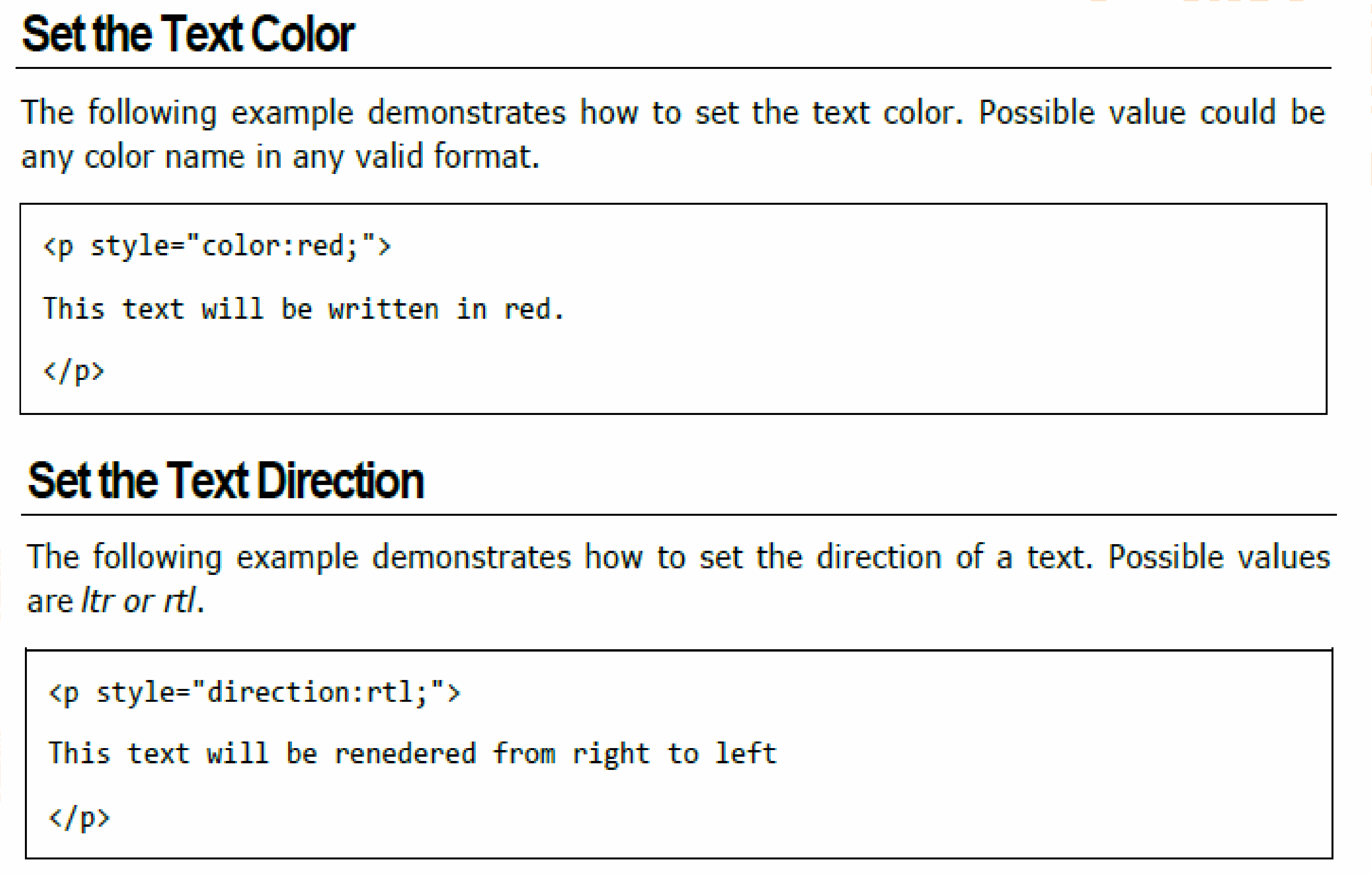
CSS — Text



You can set the following text properties of an element:

- The **color** property is used to set the color of a text.
- The **direction** property is used to set the text direction.
- The **letter-spacing** property is used to add or subtract space between the letters that make up a word.
- The **word-spacing** property is used to add or subtract space between the words of a sentence.
- The **text-indent** property is used to indent the text of a paragraph.
- The **text-align** property is used to align the text of a document.
- The **text-decoration** property is used to underline, overline, and strikethrough text.
- The **text-transform** property is used to capitalize text or convert text to uppercase or lowercase letters.
- The **white-space** property is used to control the flow and formatting of text.
- The **text-shadow** property is used to set the text shadow around a text.

CSS — Text



Set the Text Color

The following example demonstrates how to set the text color. Possible value could be any color name in any valid format.

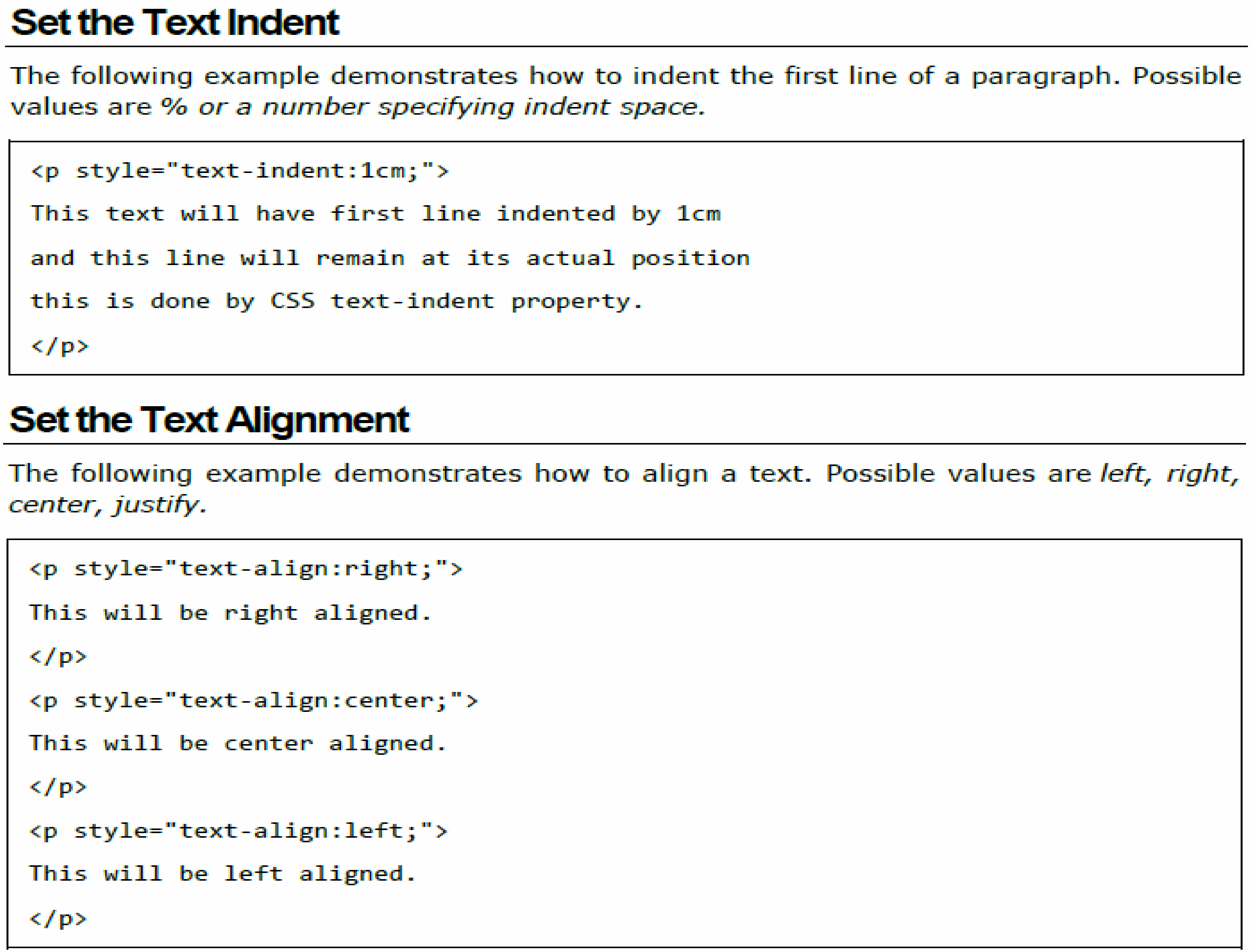
```
<p style="color:red;">

This text will be written in red.

</p>
```

Set the Text Direction

The following example demonstrates how to set the direction of a text. Possible values are *ltr or rtl*.

```
<p style="direction:rtl;">

This text will be renedered from right to left

</p>
```

CSS — Text



Set the Text Indent

The following example demonstrates how to indent the first line of a paragraph. Possible values are *% or a number specifying indent space.*

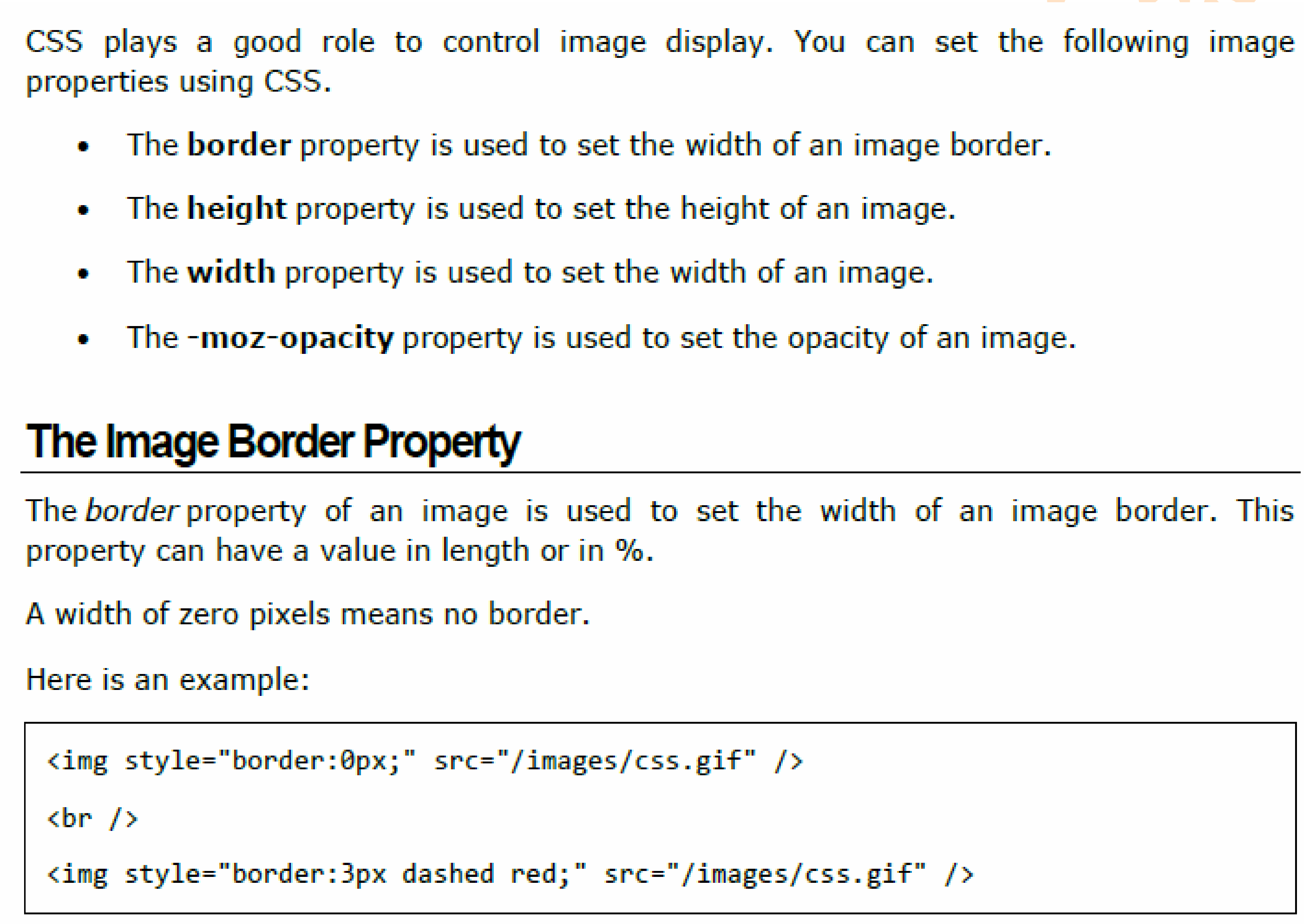
```
<p style="text-indent:1cm;">
This text will have first line indented by 1cm
and this line will remain at its actual position
this is done by CSS text-indent property.
</p>
```

Set the Text Alignment

The following example demonstrates how to align a text. Possible values are *left, right, center, justify.*

```
<p style="text-align:right;">
This will be right aligned.
</p>
<p style="text-align:center;">
This will be center aligned.
</p>
<p style="text-align:left;">
This will be left aligned.
</p>
```

CSS — Images



CSS plays a good role to control image display. You can set the following image properties using CSS.

- The **border** property is used to set the width of an image border.
- The **height** property is used to set the height of an image.
- The **width** property is used to set the width of an image.
- The **-moz-opacity** property is used to set the opacity of an image.

The Image Border Property

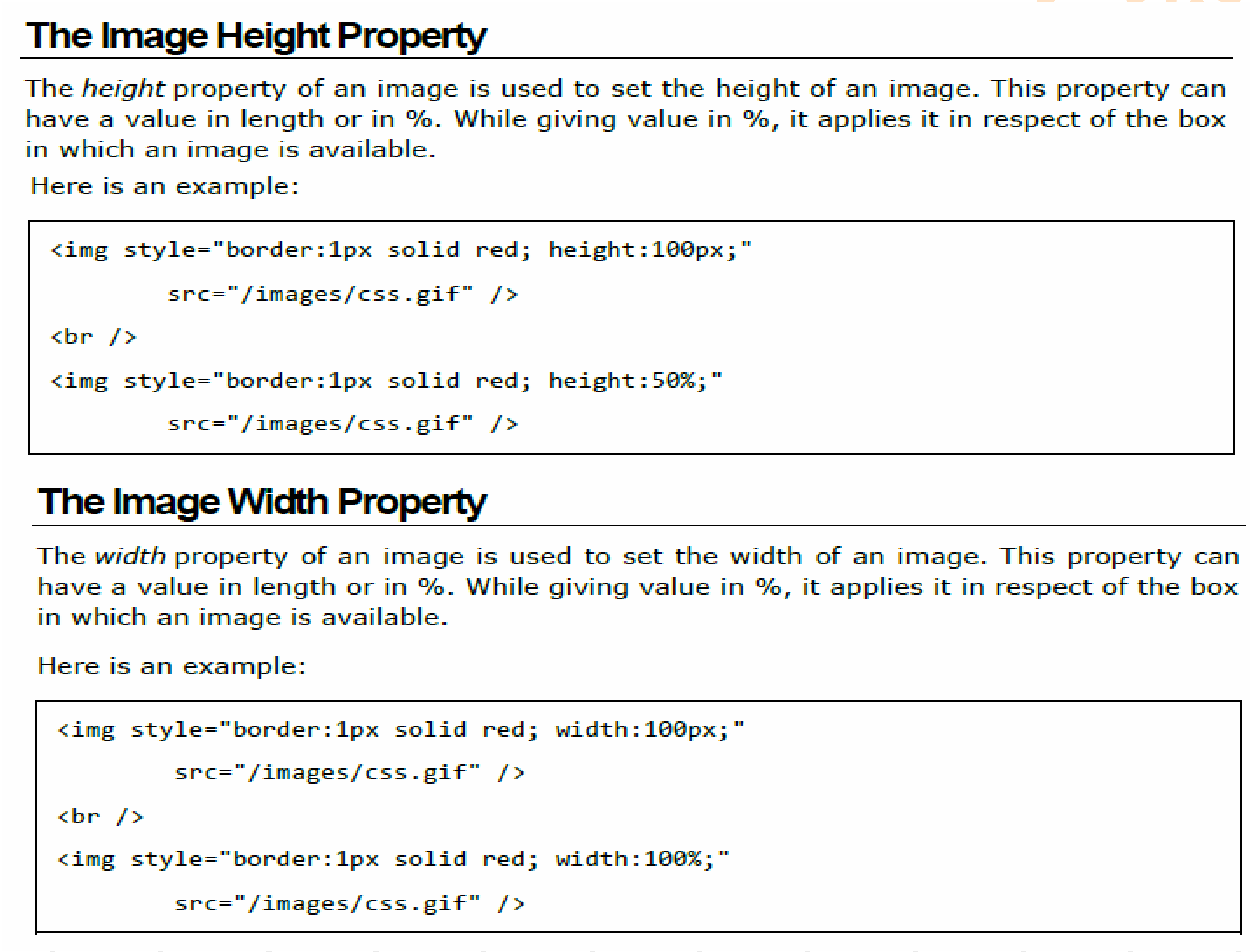
The *border* property of an image is used to set the width of an image border. This property can have a value in length or in %.

A width of zero pixels means no border.

Here is an example:

```
<img style="border:0px;" src="/images/css.gif" />

<br />

<img style="border:3px dashed red;" src="/images/css.gif" />
```

CSS — Images



The Image Height Property

The *height* property of an image is used to set the height of an image. This property can have a value in length or in %. While giving value in %, it applies it in respect of the box in which an image is available.

Here is an example:

```
<img style="border:1px solid red; height:100px;"
        src="/images/css.gif" />
<br />
<img style="border:1px solid red; height:50%;"
        src="/images/css.gif" />
```

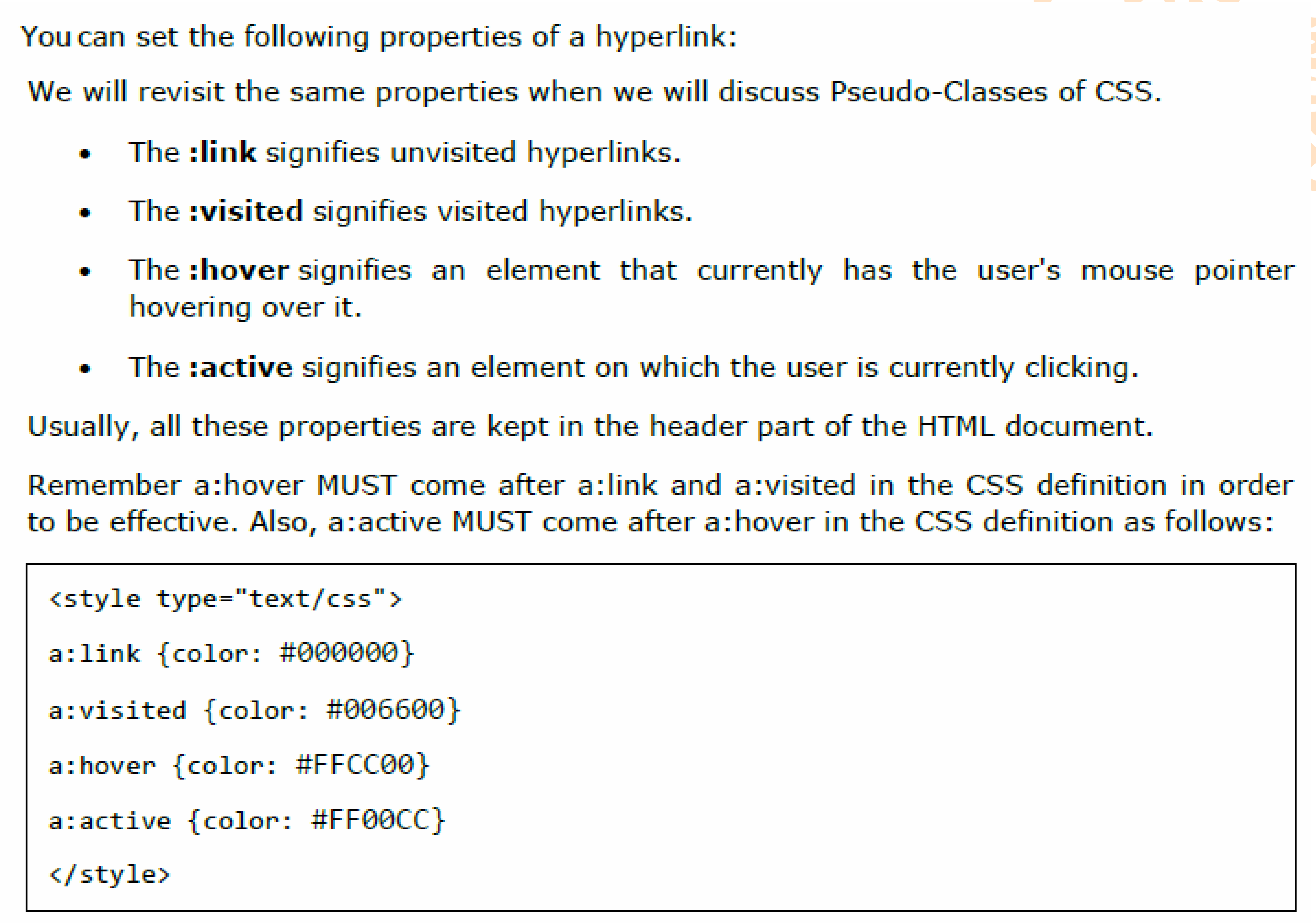
The Image Width Property

The *width* property of an image is used to set the width of an image. This property can have a value in length or in %. While giving value in %, it applies it in respect of the box in which an image is available.

Here is an example:

```
<img style="border:1px solid red; width:100px;"
        src="/images/css.gif" />
<br />
<img style="border:1px solid red; width:100%;"
        src="/images/css.gif" />
```

CSS — Links



You can set the following properties of a hyperlink:

We will revisit the same properties when we will discuss Pseudo-Classes of CSS.

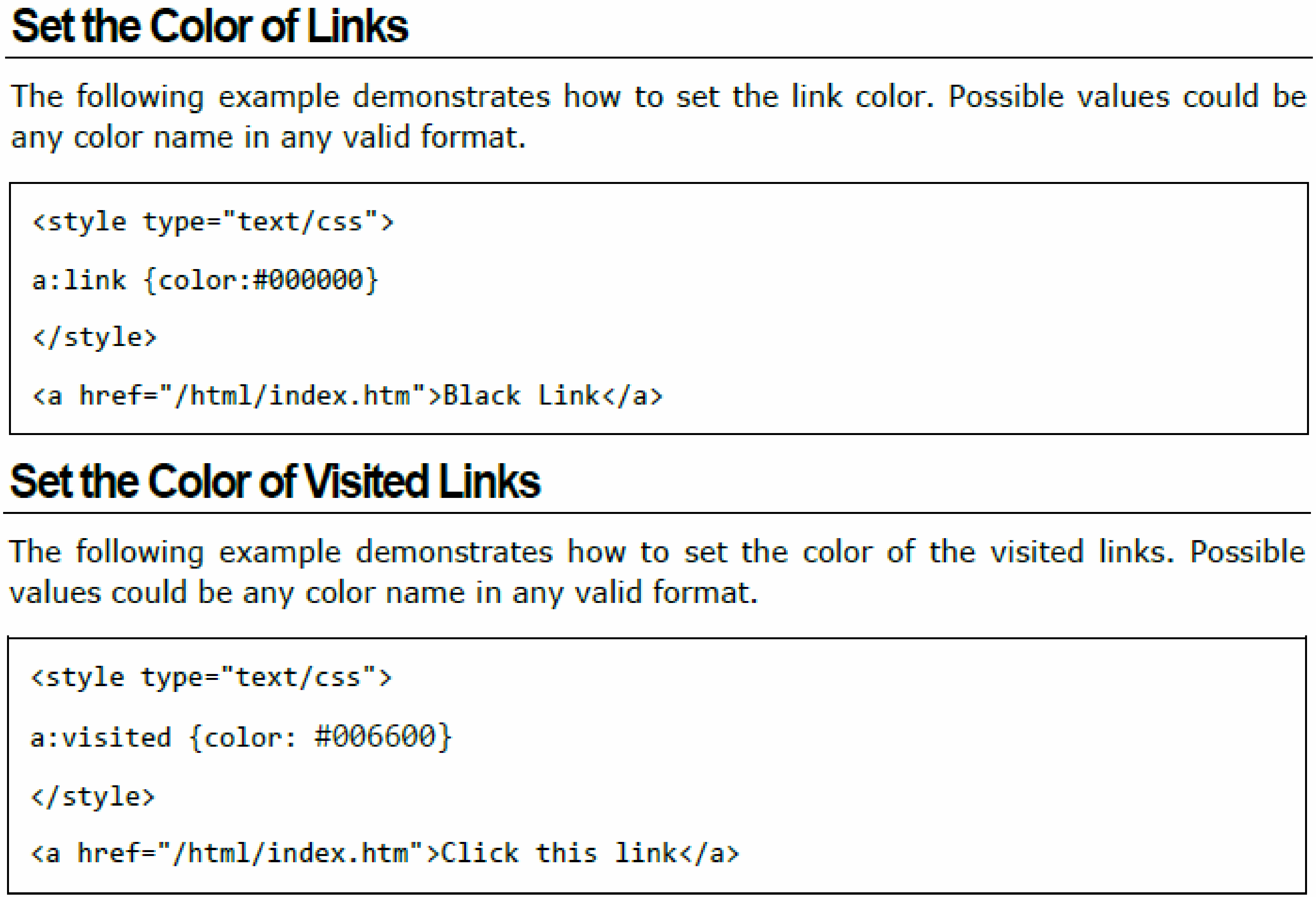
- The **:link** signifies unvisited hyperlinks.
- The **:visited** signifies visited hyperlinks.
- The **:hover** signifies an element that currently has the user's mouse pointer hovering over it.
- The **:active** signifies an element on which the user is currently clicking.

Usually, all these properties are kept in the header part of the HTML document.

Remember a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective. Also, a:active MUST come after a:hover in the CSS definition as follows:

```
<style type="text/css">

a:link {color: #000000}

a:visited {color: #006600}

a:hover {color: #FFCC00}

a:active {color: #FF00CC}

</style>
```

CSS — Links



Set the Color of Links

The following example demonstrates how to set the link color. Possible values could be any color name in any valid format.

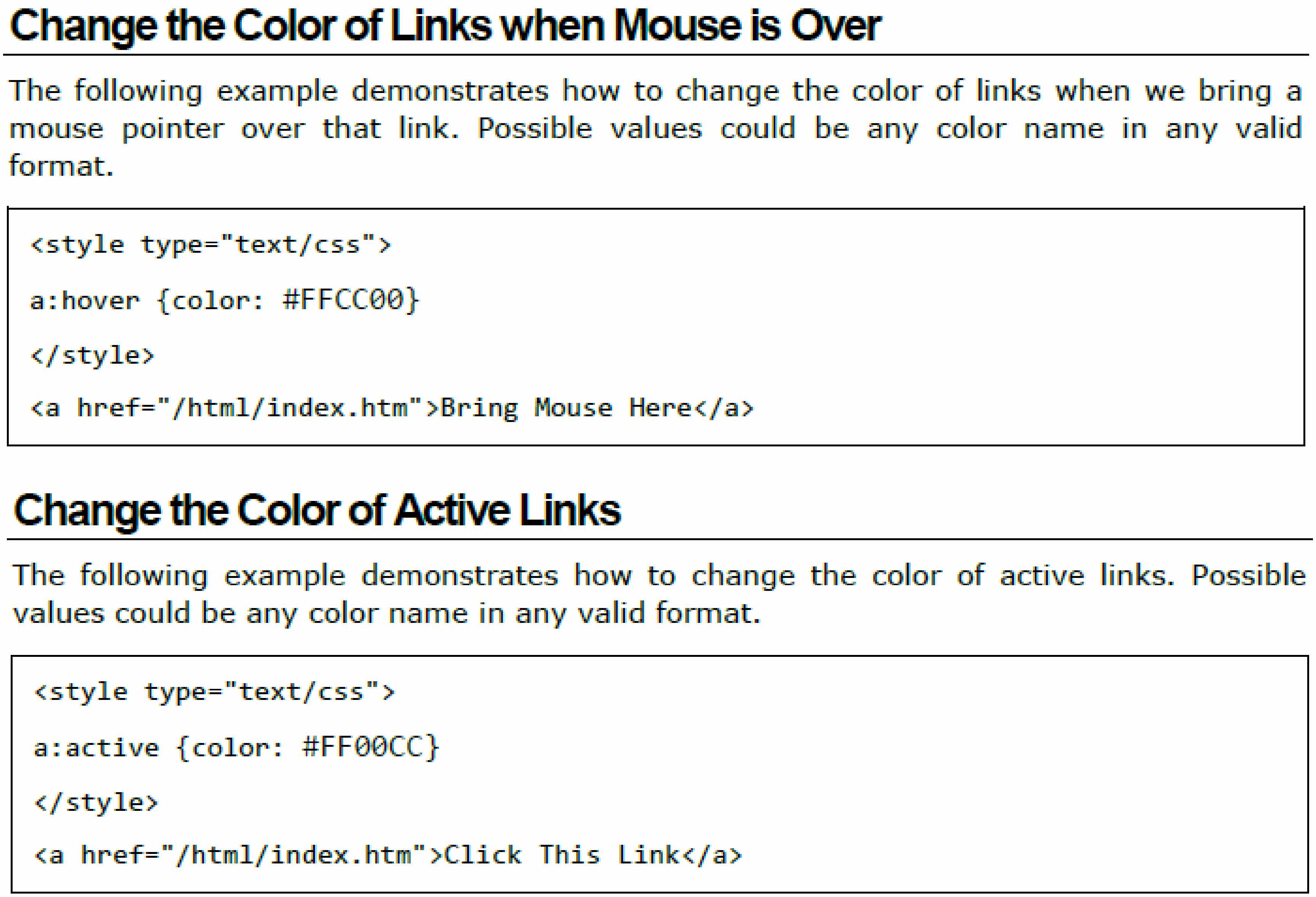
```
<style type="text/css">

a:link {color:#000000}

</style>

<a href="/html/index.htm">Black Link</a>
```

Set the Color of Visited Links

The following example demonstrates how to set the color of the visited links. Possible values could be any color name in any valid format.

```
<style type="text/css">

a:visited {color: #006600}

</style>

<a href="/html/index.htm">Click this link</a>
```

CSS — Links



Change the Color of Links when Mouse is Over

The following example demonstrates how to change the color of links when we bring a mouse pointer over that link. Possible values could be any color name in any valid format.

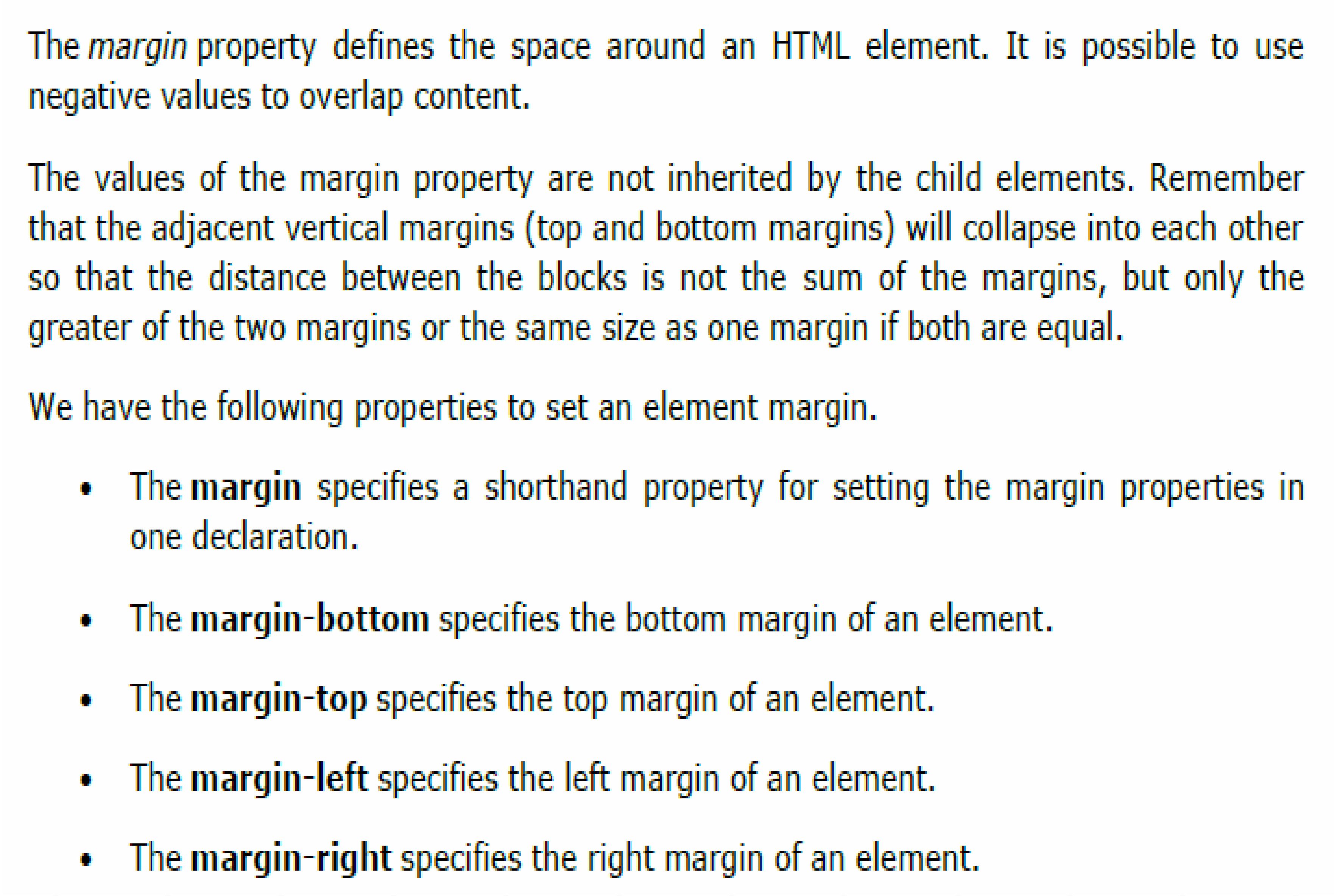
```
<style type="text/css">
a:hover {color: #FFCC00}
</style>
<a href="/html/index.htm">Bring Mouse Here</a>
```

Change the Color of Active Links

The following example demonstrates how to change the color of active links. Possible values could be any color name in any valid format.

```
<style type="text/css">
a:active {color: #FF00CC}
</style>
<a href="/html/index.htm">Click This Link</a>
```

CSS — Margins



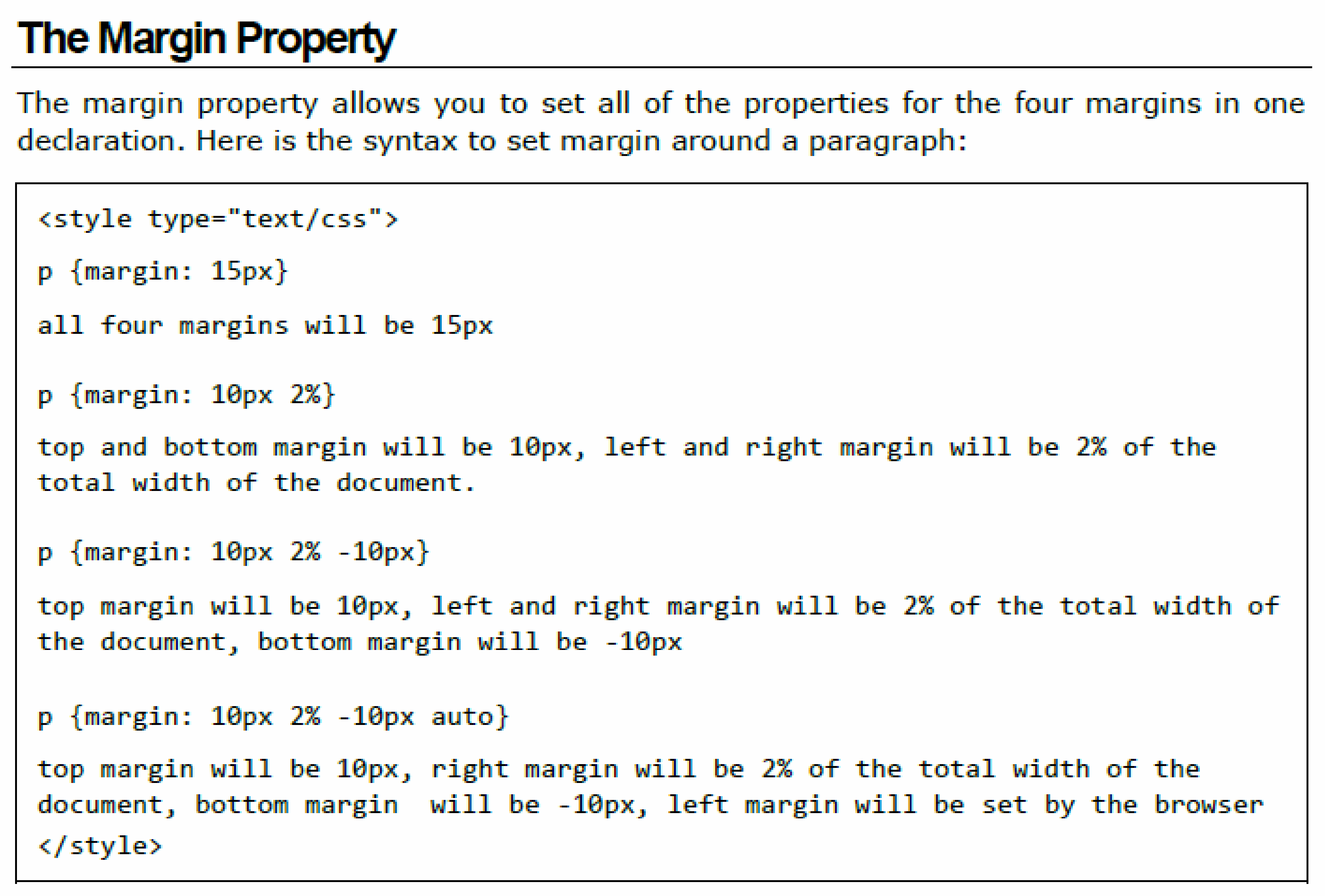
The *margin* property defines the space around an HTML element. It is possible to use negative values to overlap content.

The values of the margin property are not inherited by the child elements. Remember that the adjacent vertical margins (top and bottom margins) will collapse into each other so that the distance between the blocks is not the sum of the margins, but only the greater of the two margins or the same size as one margin if both are equal.

We have the following properties to set an element margin.

- The **margin** specifies a shorthand property for setting the margin properties in one declaration.
- The **margin-bottom** specifies the bottom margin of an element.
- The **margin-top** specifies the top margin of an element.
- The **margin-left** specifies the left margin of an element.
- The **margin-right** specifies the right margin of an element.

CSS — Margins

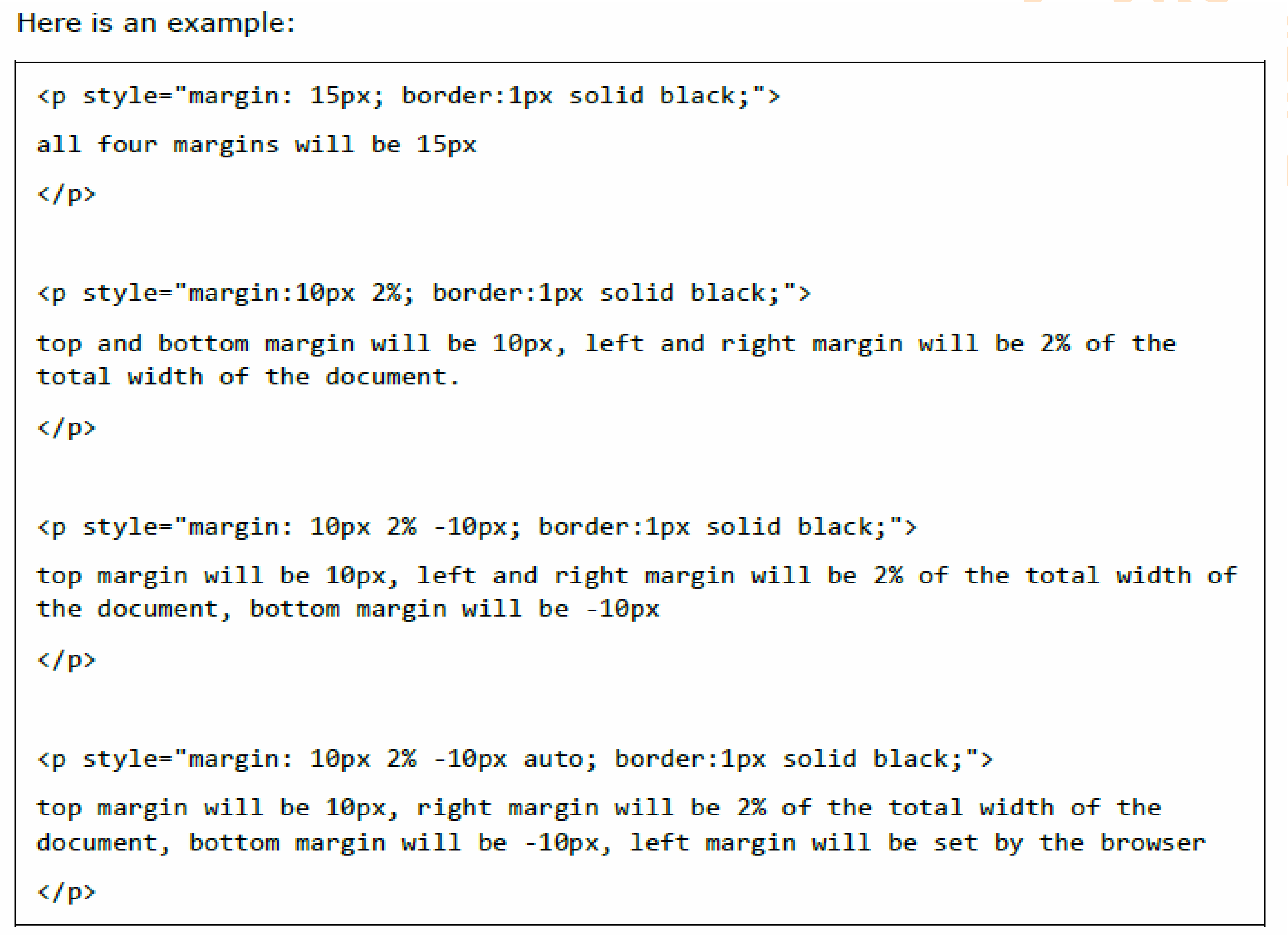


The Margin Property

The margin property allows you to set all of the properties for the four margins in one declaration. Here is the syntax to set margin around a paragraph:

```
<style type="text/css">

p {margin: 15px}

all four margins will be 15px


p {margin: 10px 2%}

top and bottom margin will be 10px, left and right margin will be 2% of the
total width of the document.


p {margin: 10px 2% -10px}

top margin will be 10px, left and right margin will be 2% of the total width of
the document, bottom margin will be -10px


p {margin: 10px 2% -10px auto}

top margin will be 10px, right margin will be 2% of the total width of the
document, bottom margin  will be -10px, left margin will be set by the browser
</style>
```

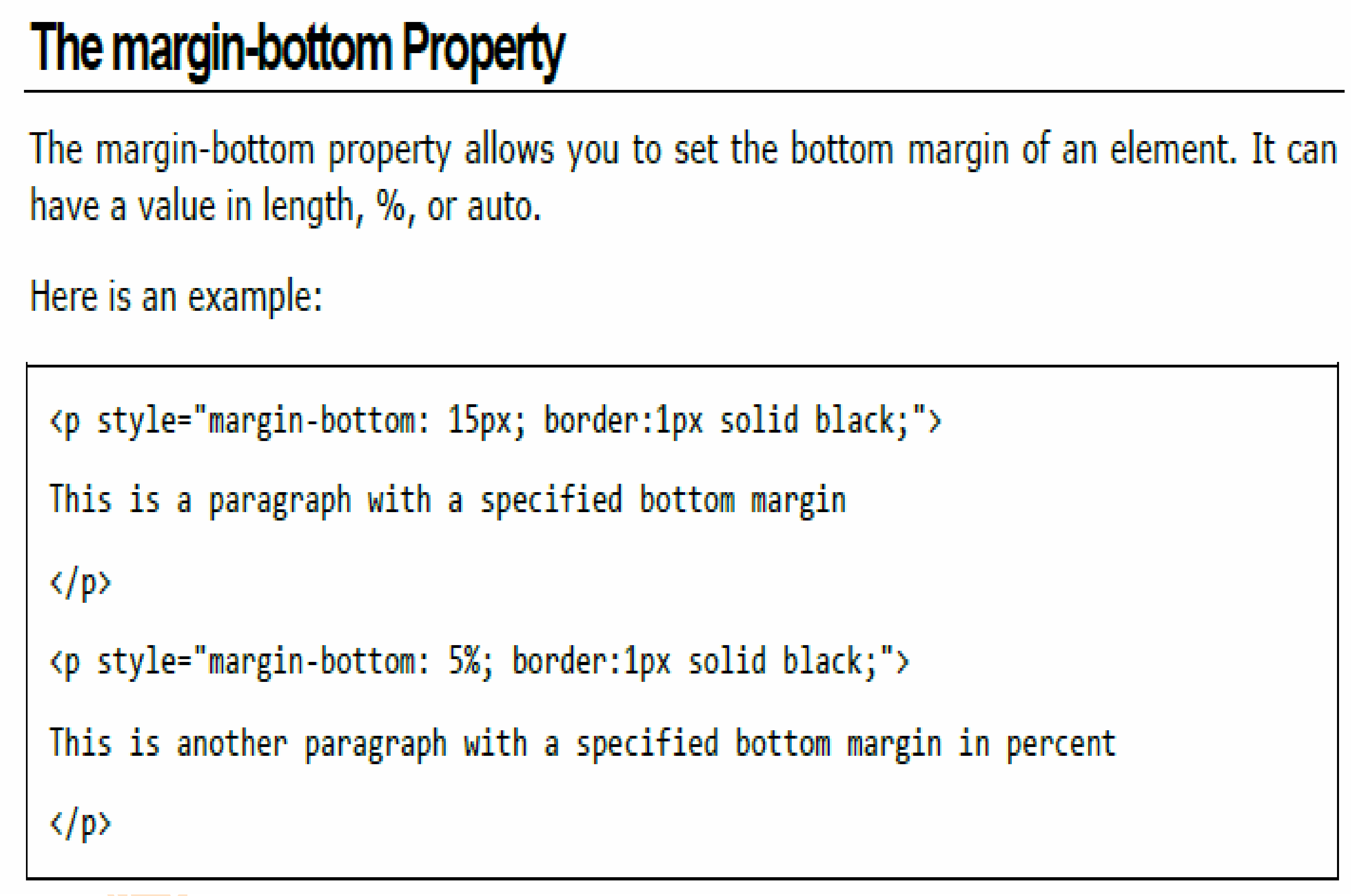
CSS — Margins



Here is an example:

```
<p style="margin: 15px; border:1px solid black;">

all four margins will be 15px

</p>


<p style="margin:10px 2%; border:1px solid black;">

top and bottom margin will be 10px, left and right margin will be 2% of the
total width of the document.

</p>


<p style="margin: 10px 2% -10px; border:1px solid black;">

top margin will be 10px, left and right margin will be 2% of the total width of
the document, bottom margin will be -10px

</p>


<p style="margin: 10px 2% -10px auto; border:1px solid black;">

top margin will be 10px, right margin will be 2% of the total width of the
document, bottom margin will be -10px, left margin will be set by the browser

</p>
```

CSS — Margins



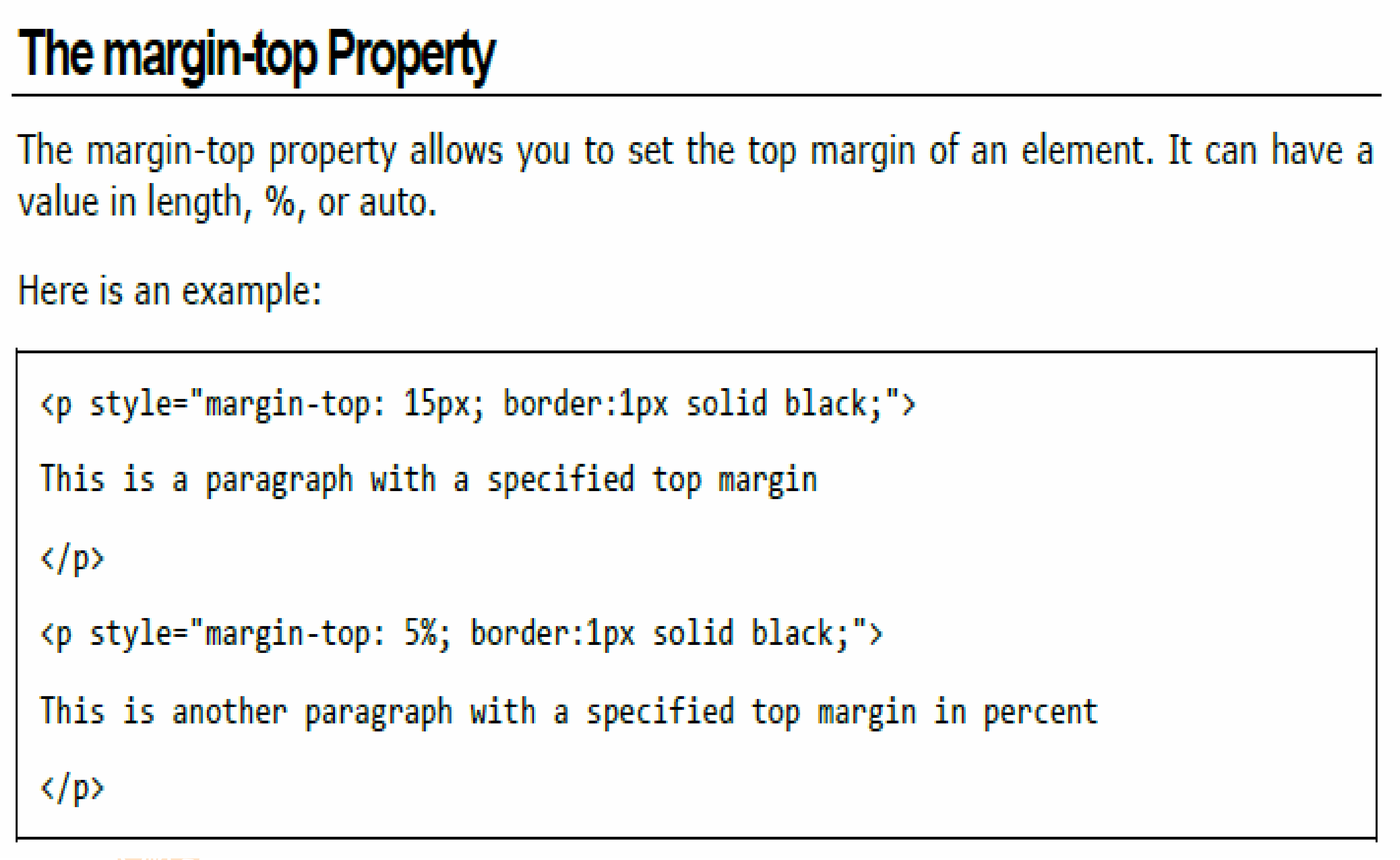
The margin-bottom Property

The margin-bottom property allows you to set the bottom margin of an element. It can have a value in length, %, or auto.

Here is an example:

```
<p style="margin-bottom: 15px; border:1px solid black;">

This is a paragraph with a specified bottom margin

</p>

<p style="margin-bottom: 5%; border:1px solid black;">

This is another paragraph with a specified bottom margin in percent

</p>
```

CSS — Margins



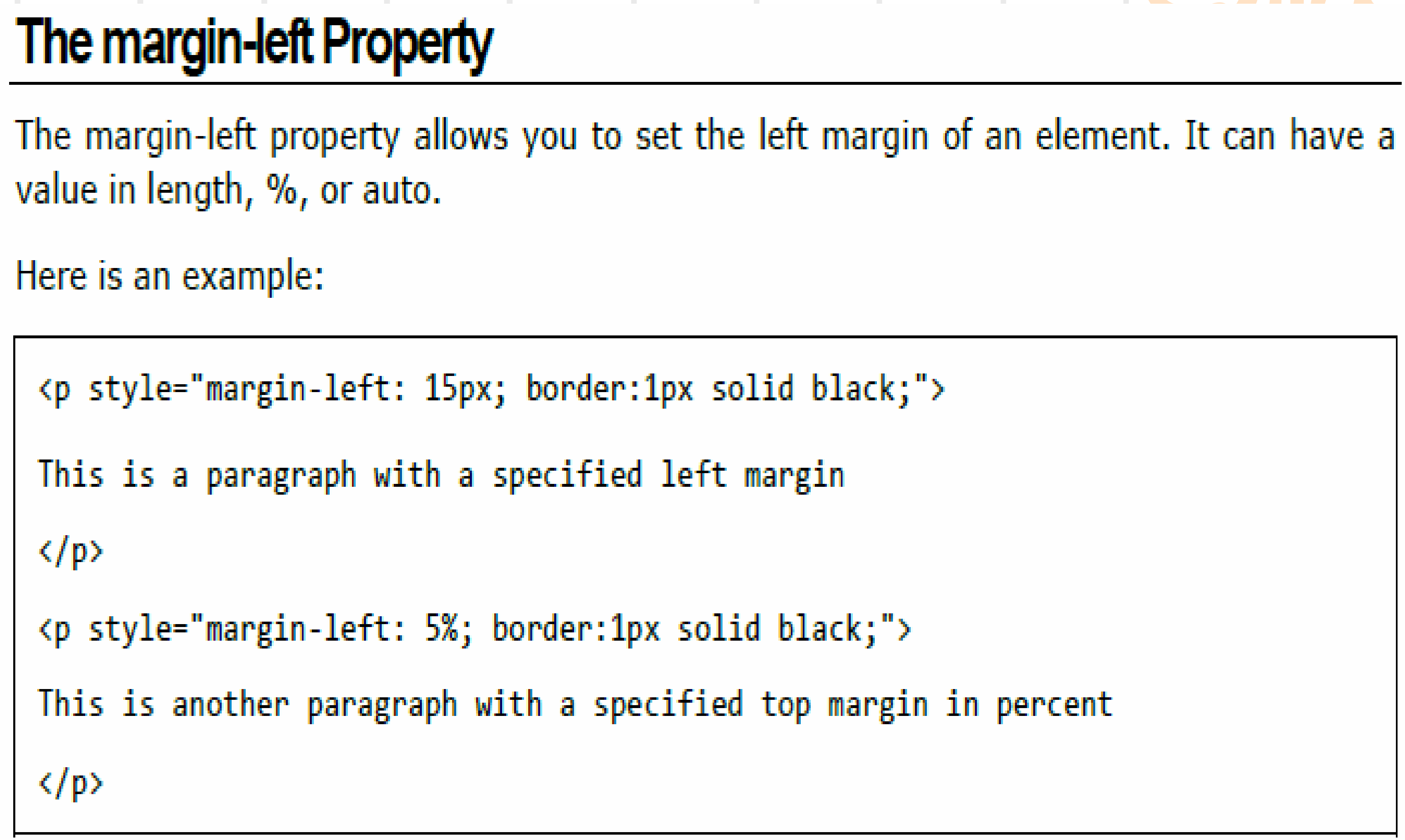
The margin-top Property

The margin-top property allows you to set the top margin of an element. It can have a value in length, %, or auto.

Here is an example:

```
<p style="margin-top: 15px; border:1px solid black;">

This is a paragraph with a specified top margin

</p>

<p style="margin-top: 5%; border:1px solid black;">

This is another paragraph with a specified top margin in percent

</p>
```

CSS — Margins



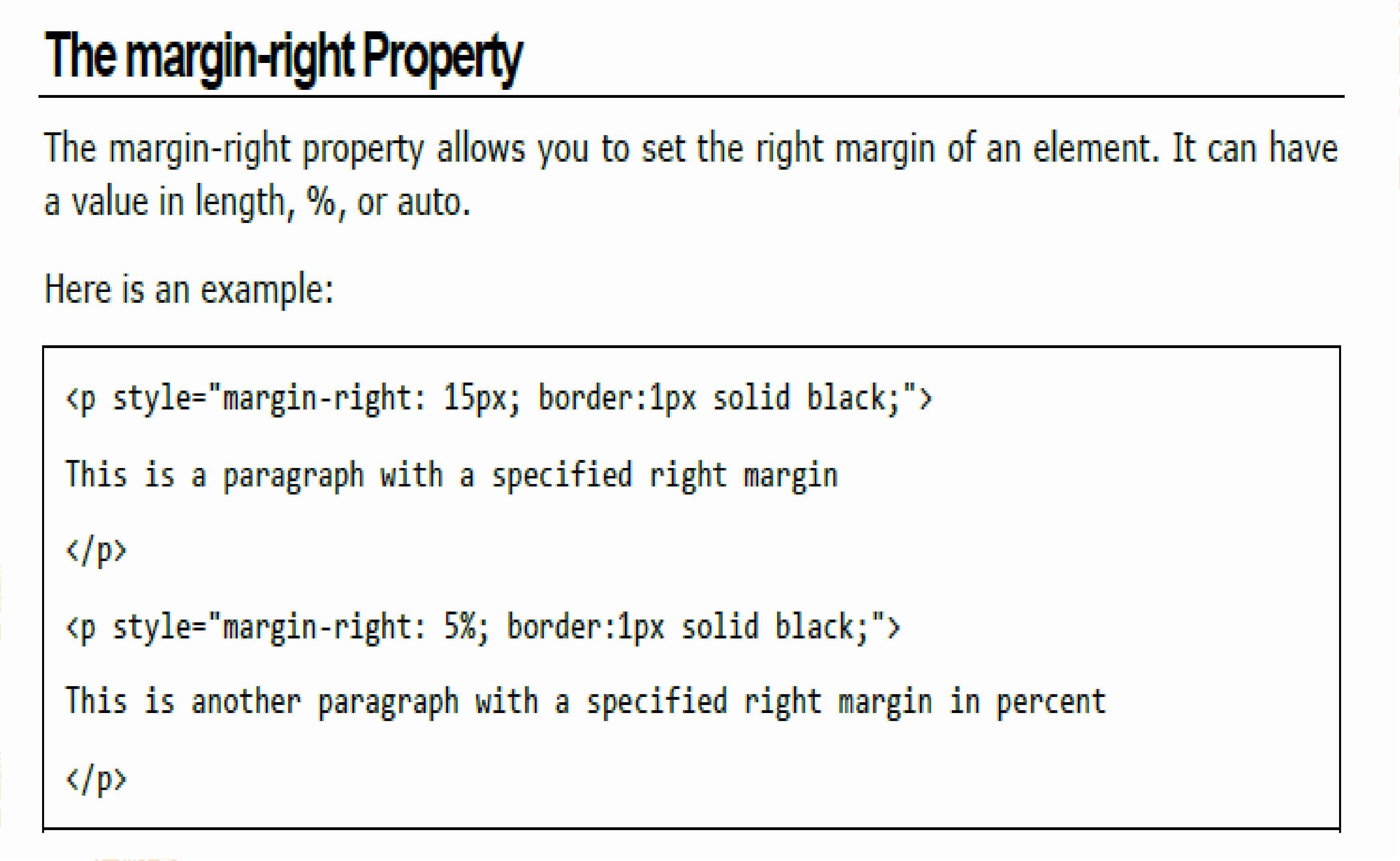
The margin-left Property

The margin-left property allows you to set the left margin of an element. It can have a value in length, %, or auto.

Here is an example:

```
<p style="margin-left: 15px; border:1px solid black;">

This is a paragraph with a specified left margin

</p>

<p style="margin-left: 5%; border:1px solid black;">

This is another paragraph with a specified top margin in percent

</p>
```

CSS — Margins



The margin-right Property

The margin-right property allows you to set the right margin of an element. It can have a value in length, %, or auto.

Here is an example:

```
<p style="margin-right: 15px; border:1px solid black;">

This is a paragraph with a specified right margin

</p>

<p style="margin-right: 5%; border:1px solid black;">

This is another paragraph with a specified right margin in percent

</p>
```