

CSS height and width Values

The `height` and `width` properties may have the following values:

- `auto` - This is default. The browser calculates the height and width
- `length` - Defines the height/width in px, cm, etc.
- `%` - Defines the height/width in percent of the containing block
- `initial` - Sets the height/width to its default value

CSS height and width Examples

This element has a height of 200 pixels and a width of 50%

Example

Set the height and width of a `<div>` element:

```
div {  
  height: 200px;  
  width: 50%;  
  background-color: powderblue;  
}
```

This element has a height of 100 pixels and a width of 500 pixels.

Example

Set the height and width of another `<div>` element:

```
div {  
  height: 100px;  
  width: 500px;  
  background-color: powderblue;  
}
```

Note: Remember that the `height` and `width` properties do not include padding, borders, or margins! They set the height/width of the area inside the padding, border, and margin of the element!

Width and Height of an Element

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

Important: When you set the width and height properties of an element with CSS, you just set the width and height of the **content area**. To calculate the total width and height of an element, you must also include the padding and borders.

Example

This `<div>` element will have a total width of 350px and a total height of 80px:

```
div {  
  width: 320px;  
  height: 50px;  
  padding: 10px;  
  border: 5px solid gray;  
  margin: 0;  
}
```

Here is the calculation:

320px (width of content area)
+ 20px (left padding + right padding)
+ 10px (left border + right border)
= 350px (total width)

50px (height of content area)
+ 20px (top padding + bottom padding)
+ 10px (top border + bottom border)
= 80px (total height)

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border

Note: The margin property also affects the total space that the box will take up on the page, but the margin is not included in the actual size of the box. The box's total width and height stops at the border.

CSS Text

Text Color

The `color` property is used to set the color of the text. The color is specified by:

- a color name - like "red"
- a HEX value - like "#ff0000"
- an RGB value - like "rgb(255,0,0)"

Look at [CSS Color Values](#) for a complete list of possible color values.

The default text color for a page is defined in the body selector.

Example

```
body {  
  color: blue;  
}  
  
h1 {  
  color: green;  
}
```

Text Color and Background Color

In this example, we define both the `background-color` property and the `color` property:

Example

```
body {  
  background-color: lightgrey;  
  color: blue;  
}
```

```
h1 {  
  background-color: black;  
  color: white;  
}
```

```
div {  
  background-color: blue;  
  color: white;  
}
```

Important: High contrast is very important for people with vision problems. So, always ensure that the contrast between the text color and the background color (or background image) is good!

Text Alignment and Text Direction

In this chapter you will learn about the following properties:

- `text-align`
- `text-align-last`
- `direction`
- `unicode-bidi`
- `vertical-align`

Text Alignment

The `text-align` property is used to set the horizontal alignment of a text.

A text can be left or right aligned, centered, or justified.

The following example shows center aligned, and left and right aligned text (left alignment is default if text direction is left-to-right, and right alignment is default if text direction is right-to-left):

Example

```
h1 {  
  text-align: center;  
}  
  
h2 {  
  text-align: left;  
}  
  
h3 {  
  text-align: right;  
}
```

When the `text-align` property is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight (like in magazines and newspapers):

Example

```
div {  
  text-align: justify;  
}
```

Text Align Last

The `text-align-last` property specifies how to align the last line of a text.

Example

Align the last line of text in three <p> elements:

```
p.a {  
  text-align-last: right;  
}  
  
p.b {  
  text-align-last: center;  
}  
  
p.c {  
  text-align-last: justify;  
}
```

Text Direction

The `direction` and `unicode-bidi` properties can be used to change the text direction of an element:

Example

```
p {  
  direction: rtl;  
  unicode-bidi: bidi-override;  
}
```

Text Decoration

In this chapter you will learn about the following properties:

- `text-decoration-line`
- `text-decoration-color`

- `text-decoration-style`
- `text-decoration-thickness`
- `text-decoration`

Add a Decoration Line to Text

The `text-decoration-line` property is used to add a decoration line to text.

Tip: You can combine more than one value, like `overline` and `underline` to display lines both over and under a text.

Example

```
h1 {
  text-decoration-line: overline;
}

h2 {
  text-decoration-line: line-through;
}

h3 {
  text-decoration-line: underline;
}

p {
  text-decoration-line: overline underline;
}
```

Note: It is not recommended to underline text that is not a link, as this often confuses the reader.

Specify a Color for the Decoration Line

The `text-decoration-color` property is used to set the color of the decoration line.

Example

```
h1 {
  text-decoration-line: overline;
  text-decoration-color: red;
}

h2 {
  text-decoration-line: line-through;
  text-decoration-color: blue;
}

h3 {
  text-decoration-line: underline;
  text-decoration-color: green;
}

p {
  text-decoration-line: overline underline;
  text-decoration-color: purple;
}
```

Specify a Style for the Decoration Line

The `text-decoration-style` property is used to set the style of the decoration line.

Example

```
h1 {
  text-decoration-line: underline;
```



```
text-decoration-style: solid;
}

h2 {
text-decoration-line: underline;
text-decoration-style: double;
}

h3 {
text-decoration-line: underline;
text-decoration-style: dotted;
}
```

Specify the Thickness for the Decoration Line

The `text-decoration-thickness` property is used to set the thickness of the decoration line.

Example

```
h1 {
text-decoration-line: underline;
text-decoration-thickness: auto;
}

h2 {
text-decoration-line: underline;
text-decoration-thickness: 5px;
}

h3 {
text-decoration-line: underline;
text-decoration-thickness: 25%;
}
```

```
}  
  
p {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: double;  
  text-decoration-thickness: 5px;  
}
```

The Shorthand Property

The `text-decoration` property is a shorthand property for:

- `text-decoration-line` (required)
- `text-decoration-color` (optional)
- `text-decoration-style` (optional)
- `text-decoration-thickness` (optional)

Example

```
h1 {  
  text-decoration: underline;  
}  
  
h2 {  
  text-decoration: underline red;  
}  
  
h3 {  
  text-decoration: underline red double;  
}  
  
p {  
  text-decoration: underline red double 5px;  
}
```

A Small Tip

All links in HTML are underlined by default. Sometimes you see that links are styled with no underline. The `text-decoration: none;` is used to remove the underline from links, like this:

Example

```
a {  
  text-decoration: none;  
}
```

Text Spacing

In this chapter you will learn about the following properties:

- `text-indent`
- `letter-spacing`
- `line-height`
- `word-spacing`
- `white-space`

Text Indentation

The `text-indent` property is used to specify the indentation of the first line of a text:

Example

```
p {  
  text-indent: 50px;  
}
```

Letter Spacing

The `letter-spacing` property is used to specify the space between the characters in a text.

The following example demonstrates how to increase or decrease the space between characters:

Example

```
h1 {  
  letter-spacing: 5px;  
}  
  
h2 {  
  letter-spacing: -2px;  
}
```

Line Height

The `line-height` property is used to specify the space between lines:

Example

```
H1 {  
  line-height: 0.8;  
}
```

Word Spacing

The `word-spacing` property is used to specify the space between the words in a text.

The following example demonstrates how to increase or decrease the space between words:

Example

```
H1{  
  word-spacing: 10px;  
}  
  
p{  
  word-spacing: -2px;  
}
```

White Space

The `white-space` property specifies how white-space inside an element is handled.

This example demonstrates how to disable text wrapping inside an element:

Example

```
p {  
  white-space: nowrap;  
}
```

Text Shadow

The `text-shadow` property adds shadow to text.

In its simplest use, you only specify the horizontal shadow (2px) and the vertical shadow (2px):

Text shadow effect!

Example

```
h1 {  
  text-shadow: 2px 2px;  
}
```

Next, add a color (red) to the shadow:

Text shadow effect!

Example

```
h1 {  
  text-shadow: 2px 2px red;  
}
```

Then, add a blur effect (5px) to the shadow:

Text shadow effect!

Example

```
h1 {  
  text-shadow: 2px 2px 5px red;  
}
```

More Text Shadow Examples

Example 1

Text-shadow on a white text:

```
h1 {  
  color: white;  
  text-shadow: 2px 2px 4px #000000;  
}
```

Example 2

Text-shadow with red neon glow:

```
h1 {  
  text-shadow: 0 0 3px #ff0000;  
}
```

Example 3

Text-shadow with red and blue neon glow:

```
h1 {  
  text-shadow: 0 0 3px #ff0000, 0 0 5px #0000ff;  
}
```

Example 4

```
h1 {  
  color: white;  
  text-shadow: 1px 1px 2px black, 0 0 25px blue, 0 0 5px darkblue;  
}
```

The CSS font-family Property

Example

Specify some different fonts for three paragraphs:

```
p1 {  
  font-family: "Times New Roman", Times, serif;  
}  
  
p2 {  
  font-family: Arial, Helvetica, sans-serif;
```

```
}
```

```
p3 {  
  font-family: "Lucida Console", "Courier New", monospace;  
}
```

CSS Links

```
/* unvisited link */  
a:link {  
  color: red;  
}
```

```
/* visited link */  
a:visited {  
  color: green;  
}
```

```
/* mouse over link */  
a:hover {  
  color: hotpink;  
}
```

```
/* selected link */  
a:active {  
  color: blue;  
}
```

Text Decoration

```
a:link {  
  text-decoration: none;  
}
```

```
a:visited {  
  text-decoration: none;  
}
```



```
a:hover {  
  text-decoration: underline;  
}
```

```
a:active {  
  text-decoration: underline;  
}
```

Background Color

```
a:link {  
  background-color: yellow;  
}
```

```
a:visited {  
  background-color: cyan;  
}
```

```
a:hover {  
  background-color: lightgreen;  
}
```

```
a:active {  
  background-color: hotpink;  
}
```

Link Buttons

This example demonstrates a more advanced example where we combine several CSS properties to display links as boxes/buttons:

Example

```
a:link, a:visited {  
  background-color: #f44336;  
  color: white;  
  padding: 14px 25px;
```

```
text-align: center;  
text-decoration: none;  
display: inline-block;  
}
```

```
a:hover, a:active {  
background-color: red;  
}
```