

# 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)"

# Text Alignment and Text Direction

In this chapter you will learn about the following properties:

- `text-align`
- `text-align-last`
- `direction`

## 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):

```
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):

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

```
}
```

## Text Align Last

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

```
#box1{
  text-align-last: right;
}

#box2{
  text-align-last: center;
}

#box3{
  text-align-last: justify;
}
```

## Text Direction

The `direction` property can be used to change the text direction of an element:

Note: by default direction is left right(ltr)

We can change right to left(rlt).

```
p {
  direction: rtl;
}
```

## Text Decoration

- `text-decoration-line`
- `text-decoration-color`
- `text-decoration-style`
- `text-decoration-thickness`
- `text-decoration`

## TEXT DECORATION LINE

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.

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

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

```
h3 {  
  
  text-decoration: underline;  
  
}
```

```
p.ex {  
  
  text-decoration: overline underline;  
  
}
```

## Specify a Color for the Decoration Line

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

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

```
}
```

## Specify a Style for the Decoration Line

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

```
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;  
}
```

```
p.ex1 {  
  text-decoration-line: underline;  
  text-decoration-style: dashed;  
}
```

```
p.ex2 {  
  text-decoration-line: underline;  
  text-decoration-style: wavy;  
}
```

```
p.ex3 {  
  text-decoration-line: underline;  
  text-decoration-color: red;  
  text-decoration-style: wavy;  
}
```

## Specify the Thickness for the Decoration Line

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

```
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)

```
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:

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

## Text Transformation

The `text-transform` property is used to specify uppercase and lowercase letters in a text.

It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word:

```
p.uppercase {  
  text-transform: uppercase;  
}  
  
p.lowercase {  
  text-transform: lowercase;  
}  
  
p.capitalize {  
  text-transform: capitalize;  
}
```

## Text Spacing

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

## Text Indentation

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

```
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:

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

## Line Height

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

```
p.small {  
  line-height: 0.8;  
}  
  
p.big {  
  line-height: 1.8;  
}
```

## 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:

```
p.one {  
  word-spacing: 10px;  
}  
  
p.two {  
  word-spacing: -2px;  
}
```

# 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):

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

## Text shadow effect!

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

## Text shadow effect!

# The position Property

The `position` property specifies the type of positioning method used for an element (static, relative, fixed, absolute or sticky).

There are five different position values:

- `static`
- `relative`
- `fixed`
- `absolute`
- `sticky`

## position: static;

HTML elements are positioned static by default.



Static positioned elements are not affected by the top, bottom, left, and right properties.

An element with `position: static;` is not positioned in any special way; it is always positioned according to the normal flow of the page:

## position: relative;

An element with `position: relative;` is positioned relative to its normal position.

Setting the top, right, bottom, and left properties of a relatively-positioned element will cause it to be adjusted away from its normal position. Other content will not be adjusted to fit into any gap left by the element.

## position: fixed;

An element with `position: fixed;` is positioned relative to the viewport, which means it always stays in the same place even if the page is scrolled. The top, right, bottom, and left properties are used to position the element.

A fixed element does not leave a gap in the page where it would normally have been located.

## position: absolute;

An element with `position: absolute;` is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed).

However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling.

**Note:** Absolute positioned elements are removed from the normal flow, and can overlap elements.

## position: sticky;

An element with `position: sticky;` is positioned based on the user's scroll position.

A sticky element toggles between `relative` and `fixed`, depending on the scroll position. It is positioned relative until a given offset position is met in the viewport - then it "sticks" in place (like `position:fixed`).