How to add CSS

CSS is added to HTML pages to format the document according to information in the style sheet. There are three ways to insert CSS in HTML documents.

1. Inline CSS
2. Internal CSS
3. External CSS

1) Inline CSS

Inline CSS is used to apply CSS on a single line or element.

For example:

1. **<p** style="color:blue"**>**Hello CSS**</p>**

For more visit here: [Inline CSS](https://www.javatpoint.com/inline-css)

2) Internal CSS

Internal CSS is used to apply CSS on a single document or page. It can affect all the elements of the page. It is written inside the style tag within head section of html.

For example:

1. **<style>**
2. p{color:blue}
3. **</style>**

For more visit here: [Internal CSS](https://www.javatpoint.com/internal-css)

3) External CSS

External CSS is used to apply CSS on multiple pages or all pages. Here, we write all the CSS code in a css file. Its extension must be .css for example style.css.

For example:

1. p{color:blue}

You need to link this style.css file to your html pages like this:

1. **<link** rel="stylesheet" type="text/css" href="style.css"**>**

The link tag must be used inside head section of html.

CSS Example with CSS Editor

In this tutorial, you will get a lot of examples, you can edit and run these examples, with our online CSS editor.

1. <!DOCTYPE**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h1{
6. color:white;
7. background-color:red;
8. padding:5px;
9. }
10. p{
11. color:blue;
12. }
13. **</style>**
14. **</head>**
15. **<body>**
16. **<h1>**Write Your First CSS Example**</h1>**
17. **<p>**This is Paragraph.**</p>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csscss1)

Output:

**Write Your First CSS Example**

This is Paragraph.

# Inline CSS

We can apply CSS in a single element by inline CSS technique.

The inline CSS is also a method to insert style sheets in HTML document. This method mitigates some advantages of style sheets so it is advised to use this method sparingly.

If you want to use inline CSS, you should use the style attribute to the relevant tag.

Syntax:

1. **<htmltag** style="cssproperty1:value; cssproperty2:value;"**>** **</htmltag>**

Example:

1. **<h2** style="color:red;margin-left:40px;"**>**Inline CSS is applied on this heading.**</h2>**
2. **<p>**This paragraph is not affected.**</p>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssinline1)

Output:

**Inline CSS is applied on this heading.**

This paragraph is not affected.

Disadvantages of Inline CSS

* You cannot use quotations within inline CSS. If you use quotations the browser will interpret this as an end of your style value.
* These styles cannot be reused anywhere else.
* These styles are tough to be edited because they are not stored at a single place.
* It is not possible to style pseudo-codes and pseudo-classes with inline CSS.
* Inline CSS does not provide browser cache advantages.

Internal CSS

The internal style sheet is used to add a unique style for a single document. It is defined in <head> section of the HTML page inside the <style> tag.

Example:

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. background-color: linen;
7. }
8. h1 {
9. color: red;
10. margin-left: 80px;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h1>**The internal style sheet is applied on this heading.**</h1>**
16. **<p>**This paragraph will not be affected.**</p>**
17. **</body>**
18. **</html>**

External CSS

The external style sheet is generally used when you want to make changes on multiple pages. It is ideal for this condition because it facilitates you to change the look of the entire web site by changing just one file.

It uses the <link> tag on every pages and the <link> tag should be put inside the head section.

Example:

1. **<head>**
2. **<link** rel="stylesheet" type="text/css" href="mystyle.css"**>**
3. **</head>**

The external style sheet may be written in any text editor but must be saved with a .css extension. This file should not contain HTML elements.

Let's take an example of a style sheet file named "mystyle.css".

*File: mystyle.css*

1. body {
2. background-color: lightblue;
3. }
4. h1 {
5. color: navy;
6. margin-left: 20px;
7. }

Note: You should not use a space between the property value and the unit. For example: It should be margin-left:20px not margin-left:20 px.

CSS Comments

CSS comments are generally written to explain your code. It is very helpful for the users who reads your code so that they can easily understand the code.

Comments are ignored by browsers.

Comments are single or multiple lines statement and written within /\*............\*/ .

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. color: blue;
7. /\* This is a single-line comment \*/
8. text-align: center;
9. }
10. /\* This is
11. a multi-line
12. comment \*/
13. **</style>**
14. **</head>**
15. **<body>**
16. **<p>**Hello Javatpoint.com**</p>**
17. **<p>**This statement is styled with CSS.**</p>**
18. **<p>**CSS comments are ignored by the browsers and not shown in the output.**</p>**
19. **</body>**
20. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csscomments1)

Output:

Hello Javatpoint.com

This statement is styled with CSS.

CSS comments are ignored by the browsers and not shown in the output.

CSS Background

CSS background property is used to define the background effects on element. There are 5 CSS background properties that affects the HTML elements:

1. background-color
2. background-image
3. background-repeat
4. background-attachment
5. background-position

1) CSS background-color

The background-color property is used to specify the background color of the element.

You can set the background color like this:

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h2,p{
6. background-color: #b0d4de;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<h2>**My first CSS page.**</h2>**
12. **<p>**Hello Javatpoint. This is an example of CSS background-color.**</p>**
13. **</body>**
14. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssbackground1)

Output:

**My first CSS page.**

Hello Javatpoint. This is an example of CSS background-color.

2) CSS background-image

The background-image property is used to set an image as a background of an element. By default the image covers the entire element. You can set the background image for a page like this.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. background-image: url("paper1.gif");
7. margin-left:100px;
8. }
9. **</style>**
10. **</head>**
11. **<body>**
12. **<h1>**Hello Javatpoint.com**</h1>**
13. **</body>**
14. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssbackground2)

Note: The background image should be chosen according to text color. The bad combination of text and background image may be a cause of poor designed and not readable webpage.

3) CSS background-repeat

By default, the background-image property repeats the background image horizontally and vertically. Some images are repeated only horizontally or vertically.

The background looks better if the image repeated horizontally only.

**background-repeat: repeat-x;**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. background-image: url("gradient\_bg.png");
7. background-repeat: repeat-x;
8. }
9. **</style>**
10. **</head>**
11. **<body>**
12. **<h1>**Hello Javatpoint.com**</h1>**
13. **</body>**
14. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssbackground3)

**background-repeat: repeat-y;**

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. background-image: url("gradient\_bg.png");
7. background-repeat: repeat-y;
8. }
9. **</style>**
10. **</head>**
11. **<body>**
12. **<h1>**Hello Javatpoint.com**</h1>**
13. **</body>**
14. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssbackground3y)

4) CSS background-attachment

The background-attachment property is used to specify if the background image is fixed or scroll with the rest of the page in browser window. If you set fixed the background image then the image will not move during scrolling in the browser. Let?s take an example with fixed background image.

1. background: white url('bbb.gif');
2. background-repeat: no-repeat;
3. background-attachment: fixed;

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssbackground4)

5) CSS background-position

The background-position property is used to define the initial position of the background image. By default, the background image is placed on the top-left of the webpage.

You can set the following positions:

1. center
2. top
3. bottom
4. left
5. right
6. background: white url('good-morning.jpg');
7. background-repeat: no-repeat;
8. background-attachment: fixed;
9. background-position: center;

# CSS Border

The CSS border is a shorthand property used to set the border on an element.

The CSS border properties are use to specify the style, color and size of the border of an element. The CSS border properties are given below

* border-style
* border-color
* border-width
* border-radius

## 1) CSS border-style

The Border style property is used to specify the border type which you want to display on the web page.

There are some border style values which are used with border-style property to define a border.

|  |  |
| --- | --- |
| **Value** | **Description** |
| none | It doesn't define any border. |
| dotted | It is used to define a dotted border. |
| dashed | It is used to define a dashed border. |
| solid | It is used to define a solid border. |
| double | It defines two borders wIth the same border-width value. |
| groove | It defines a 3d grooved border. effect is generated according to border-color value. |
| ridge | It defines a 3d ridged border. effect is generated according to border-color value. |
| inset | It defines a 3d inset border. effect is generated according to border-color value. |
| outset | It defines a 3d outset border. effect is generated according to border-color value. |

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p.none {border-style: none;}
6. p.dotted {border-style: dotted;}
7. p.dashed {border-style: dashed;}
8. p.solid {border-style: solid;}
9. p.double {border-style: double;}
10. p.groove {border-style: groove;}
11. p.ridge {border-style: ridge;}
12. p.inset {border-style: inset;}
13. p.outset {border-style: outset;}
14. p.hidden {border-style: hidden;}
15. **</style>**
16. **</head>**
17. **<body>**
18. **<p** class="none"**>**No border.**</p>**
19. **<p** class="dotted"**>**A dotted border.**</p>**
20. **<p** class="dashed"**>**A dashed border.**</p>**
21. **<p** class="solid"**>**A solid border.**</p>**
22. **<p** class="double"**>**A double border.**</p>**
23. **<p** class="groove"**>**A groove border.**</p>**
24. **<p** class="ridge"**>**A ridge border.**</p>**
25. **<p** class="inset"**>**An inset border.**</p>**
26. **<p** class="outset"**>**An outset border.**</p>**
27. **<p** class="hidden"**>**A hidden border.**</p>**
28. **</body>**
29. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssborder1)

Output:

No border.

A dotted border.

A dashed border.

A solid border.

A double border.

A groove border.

A ridge border.

An inset border.

An outset border.

A hidden border.

## 2) CSS border-width

The border-width property is used to set the border's width. It is set in pixels. You can also use the one of the three pre-defined values, thin, medium or thick to set the width of the border.

#### Note: The border-width property is not used alone. It is always used with other border properties like "border-style" property to set the border first otherwise it will not work.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p.one {
6. border-style: solid;
7. border-width: 5px;
8. }
9. p.two {
10. border-style: solid;
11. border-width: medium;
12. }
13. p.three {
14. border-style: solid;
15. border-width: 1px;
16. }
17. **</style>**
18. **</head>**
19. **<body>**
20. **<p** class="one"**>**Write your text here.**</p>**
21. **<p** class="two"**>**Write your text here.**</p>**
22. **<p** class="three"**>**Write your text here.**</p>**
23. **</body>**
24. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssborder2)

## 3) CSS border-color

There are three methods to set the color of the border.

* Name: It specifies the color name. For example: "red".
* RGB: It specifies the RGB value of the color. For example: "rgb(255,0,0)".
* Hex: It specifies the hex value of the color. For example: "#ff0000".

There is also a border color named "transparent". If the border color is not set it is inherited from the color property of the element.

#### Note: The border-color property is not used alone. It is always used with other border properties like "border-style" property to set the border first otherwise it will not work.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p.one {
6. border-style: solid;
7. border-color: red;
8. }
9. p.two {
10. border-style: solid;
11. border-color: #98bf21;
12. }
13. **</style>**
14. **</head>**
15. **<body>**
16. **<p** class="one"**>**This is a solid red border**</p>**
17. **<p** class="two"**>**This is a solid green border**</p>**
18. **</body>**
19. **</html>**

CSS Display

CSS display is the most important property of CSS which is used to control the layout of the element. It specifies how the element is displayed.

Every element has a default display value according to its nature. Every element on the webpage is a rectangular box and the CSS property defines the behavior of that rectangular box.

CSS Display default properties

|  |  |
| --- | --- |
| default value | inline |
| inherited | no |
| animation supporting | no |
| version | css1 |
| javascript syntax | object.style.display="none" |

Syntax

1. display:value;

CSS display values

There are following CSS display values which are commonly used.

1. display: inline;
2. display: inline-block;
3. display: block;
4. display: run-in;
5. display: none;

1) CSS display inline

The inline element takes the required width only. It doesn't force the line break so the flow of text doesn't break in inline example.

The inline elements are:

* <span>
* <a>
* <em>
* <b> etc.

Let's see an example of CSS display inline.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. display: inline;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<p>**Hello Javatpoint.com**</p>**
12. **<p>**Java Tutorial.**</p>**
13. **<p>**SQL Tutorial.**</p>**
14. **<p>**HTML Tutorial.**</p>**
15. **<p>**CSS Tutorial.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssdisplay1)

Output:

Hello Javatpoint.com Java Tutorial. SQL Tutorial. HTML Tutorial. CSS Tutorial.

2) CSS display inline-block

The CSS display inline-block element is very similar to inline element but the difference is that you are able to set the width and height.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. display: inline-block;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<p>**Hello Javatpoint.com**</p>**
12. **<p>**Java Tutorial.**</p>**
13. **<p>**SQL Tutorial.**</p>**
14. **<p>**HTML Tutorial.**</p>**
15. **<p>**CSS Tutorial.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssdisplay2)

Output:

Hello Javatpoint.com Java Tutorial. SQL Tutorial. HTML Tutorial. CSS Tutorial.

3) CSS display block

The CSS display block element takes as much as horizontal space as they can. Means the block element takes the full available width. They make a line break before and after them.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. display: block;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<p>**Hello Javatpoint.com**</p>**
12. **<p>**Java Tutorial.**</p>**
13. **<p>**SQL Tutorial.**</p>**
14. **<p>**HTML Tutorial.**</p>**
15. **<p>**CSS Tutorial.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssdisplay3)

Output:

Hello Javatpoint.com

Java Tutorial.

SQL Tutorial.

HTML Tutorial.

CSS Tutorial.

4) CSS display run-in

This property doesn?t work in Mozilla Firefox. These elements don't produce a specific box by themselves.

* If the run-in box contains a bock box, it will be same as block.
* If the block box follows the run-in box, the run-in box becomes the first inline box of the block box.
* If the inline box follows the run-in box, the run-in box becomes a block box.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. display: run-in;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<p>**Hello Javatpoint.com**</p>**
12. **<p>**Java Tutorial.**</p>**
13. **<p>**SQL Tutorial.**</p>**
14. **<p>**HTML Tutorial.**</p>**
15. **<p>**CSS Tutorial.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssdisplay4)

Output:

Hello Javatpoint.com

Java Tutorial.

SQL Tutorial.

HTML Tutorial.

CSS Tutorial.

5) CSS display none

The "none" value totally removes the element from the page. It will not take any space.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h1.hidden {
6. display: none;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<h1>**This heading is visible.**</h1>**
12. **<h1** class="hidden"**>**This is not visible.**</h1>**
13. **<p>**You can see that the hidden heading does not contain any space.**</p>**
14. **</body>**
15. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssdisplay5)

Output:

**This heading is visible.**

You can see that the hidden heading does not contain any space.

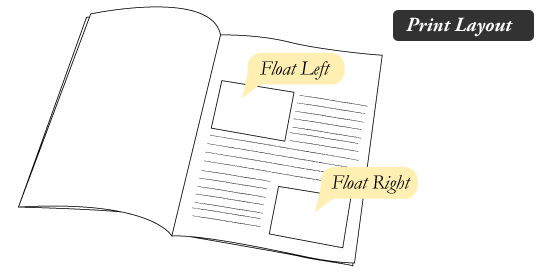
Other CSS display values

|  |  |
| --- | --- |
| **Property-value** | **Description** |
| flex | It is used to display an element as an block-level flex container. It is new in css3. |
| inline-flex | It is used to display an element as an inline-level flex container. It is new in css3. |
| inline-table | It displays an element as an inline-level table. |
| list-Item | It makes the element behave like a <li> element. |
| table | It makes the element behave like a <table> element. |
| table-caption | It makes the element behave like a <caption> element. |
| table-column-group | It makes the element behave like a <colgroup> element. |
| table-header-group | It makes the element behave like a <thead> element. |
| table-footer-group | It makes the element behave like a <tfoot> element. |
| table-row-group | It makes the element behave like a <tbody> element. |
| table-cell | It makes the element behave like a <td> element. |
| table-row | It makes the element behave like a <tr> element. |
| table-column | It makes the element behave like a <col> element. |

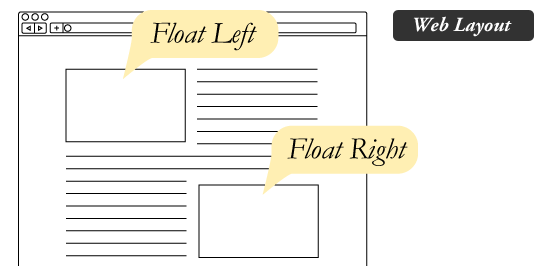
CSS Float

The **CSS float property** is *a positioning property*. It is used *to push an element to the left or right*, allowing other element to wrap around it. It is generally used with images and layouts.

To understand its purpose and origin, let's take a look to its print display. In the print display, image is set into the page such that text wraps around it as needed.



Its web layout is also just similar to print layout.



How it works

Elements are floated only horizontally. So it is possible only to float elements left or right, not up or down.

1. A floated element may be moved as far to the left or the right as possible. Simply, it means that a floated element can display at extreme left or extreme right.
2. The elements after the floating element will flow around it.
3. The elements before the floating element will not be affected.
4. If the image floated to the right, the texts flow around it, to the left and if the image floated to the left, the text flows around it, to the right.

CSS Float Properties

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| clear | The clear property is used to avoid elements after the floating elements which flow around it. | left, right, both, none, inherit |
| float | It specifies whether the box should float or not. | left, right, none, inherit |

CSS Float Property Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| none | It specifies that the element is not floated, and will be displayed just where it occurs in the text. this is a default value. |
| left | It is used to float the element to the left. |
| right | It is used to float the element to the right. |
| initial | It sets the property to its initial value. |
| inherit | It is used to inherit this property from its parent element. |

CSS Float Property Example

Let's see a simple example to understand the CSS float property.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. img {
6. float: right;
7. }
8. **</style>**
9. **</head>**
10. **<body>**
11. **<p>**The following paragraph contains an image with style
12. **<b>**float:right**</b>**. The result is that the image will float to the right in the paragraph.**</p>**
13. **<img** src="good-morning.jpg" alt="Good Morning Friends"**/>**
14. This is some text. This is some text. This is some text.
15. This is some text. This is some text. This is some text.
16. This is some text. This is some text. This is some text.
17. This is some text. This is some text. This is some text.
18. This is some text. This is some text. This is some text.
19. This is some text. This is some text. This is some text.
20. This is some text. This is some text. This is some text.
21. This is some text. This is some text. This is some text.
22. This is some text. This is some text. This is some text.
23. This is some text. This is some text. This is some text.
24. This is some text. This is some text. This is some text.
25. This is some text. This is some text. This is some text.
26. This is some text. This is some text. This is some text.
27. **</p>**
28. **</body>**
29. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfloat1)

# CSS Font

CSS Font property is used to control the look of texts. By the use of CSS font property you can change the text size, color, style and more. You have already studied how to make text bold or underlined. Here, you will also know how to resize your font using percentage.

These are some important font attributes:

1. **CSS Font color**: This property is used to change the color of the text. (standalone attribute)
2. **CSS Font family**: This property is used to change the face of the font.
3. **CSS Font size**: This property is used to increase or decrease the size of the font.
4. **CSS Font style**: This property is used to make the font bold, italic or oblique.
5. **CSS Font variant**: This property creates a small-caps effect.
6. **CSS Font weight**: This property is used to increase or decrease the boldness and lightness of the font.

## 1) CSS Font Color

CSS font color is a standalone attribute in CSS although it seems that it is a part of CSS fonts. It is used to change the color of the text.

There are three different formats to define a color:

* By a color name
* By hexadecimal value
* By RGB

In the above example, we have defined all these formats.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. font-size: 100%;
7. }
8. h1 { color: red; }
9. h2 { color: #9000A1; }
10. p { color:rgb(0, 220, 98); }
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h1>**This is heading 1**</h1>**
16. **<h2>**This is heading 2**</h2>**
17. **<p>**This is a paragraph.**</p>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont1)

Output:

## This is heading 1

### This is heading 2

This is a paragraph.

## 2) CSS Font Family

CSS font family can be divided in two types:

* Generic family: It includes Serif, Sans-serif, and Monospace.
* Font family: It specifies the font family name like Arial, New Times Roman etc.

**Serif**: Serif fonts include small lines at the end of characters. Example of serif: Times new roman, Georgia etc.

**Sans-serif**: A sans-serif font doesn't include the small lines at the end of characters. Example of Sans-serif: Arial, Verdana etc.



1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. font-size: 100%;
7. }
8. h1 { font-family: sans-serif; }
9. h2 { font-family: serif; }
10. p { font-family: monospace; }
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h1>**This heading is shown in sans-serif.**</h1>**
16. **<h2>**This heading is shown in serif.**</h2>**
17. **<p>**This paragraph is written in monospace.**</p>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont2)

Output:

## This heading is shown in sans-serif.

### This heading is shown in serif.

This paragraph is written in monospace.

## 3) CSS Font Size

CSS font size property is used to change the size of the font.

These are the possible values that can be used to set the font size:

|  |  |
| --- | --- |
| **Font Size Value** | **Description** |
| xx-small | used to display the extremely small text size. |
| x-small | used to display the extra small text size. |
| small | used to display small text size. |
| medium | used to display medium text size. |
| large | used to display large text size. |
| x-large | used to display extra large text size. |
| xx-large | used to display extremely large text size. |
| smaller | used to display comparatively smaller text size. |
| larger | used to display comparatively larger text size. |
| size in pixels or % | used to set value in percentage or in pixels. |

1. **<html>**
2. **<head>**
3. **<title>**Practice CSS font-size property**</title>**
4. **</head>**
5. **<body>**
6. **<p** style="font-size:xx-small;"**>**  This font size is extremely small.**</p>**
7. **<p** style="font-size:x-small;"**>**  This font size is extra small**</p>**
8. **<p** style="font-size:small;"**>**  This font size is small**</p>**
9. **<p** style="font-size:medium;"**>**  This font size is medium. **</p>**
10. **<p** style="font-size:large;"**>**  This font size is large. **</p>**
11. **<p** style="font-size:x-large;"**>**  This font size is extra large. **</p>**
12. **<p** style="font-size:xx-large;"**>**  This font size is extremely large. **</p>**
13. **<p** style="font-size:smaller;"**>**  This font size is smaller. **</p>**
14. **<p** style="font-size:larger;"**>**  This font size is larger. **</p>**
15. **<p** style="font-size:200%;"**>**  This font size is set on 200%. **</p>**
16. **<p** style="font-size:20px;"**>**  This font size is 20 pixels.  **</p>**
17. **</body>**
18. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont3)

Output:

This font size is extremely small.

This font size is extra small

This font size is small

This font size is medium.

This font size is large.

This font size is extra large.

This font size is extremely large.

This font size is smaller.

This font size is larger.

This font size is set on 200%.

This font size is 20 pixels.

## 4) CSS Font Style

CSS Font style property defines what type of font you want to display. It may be italic, oblique, or normal.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. body {
6. font-size: 100%;
7. }
8. h2 { font-style: italic; }
9. h3 { font-style: oblique; }
10. h4 { font-style: normal; }
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h2>**This heading is shown in italic font.**</h2>**
16. **<h3>**This heading is shown in oblique font.**</h3>**
17. **<h4>**This heading is shown in normal font.**</h4>**
18. **</body>**
19. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont4)

Output:

## *This heading is shown in italic font.*

### *This heading is shown in oblique font.*

#### This heading is shown in normal font.

## 5) CSS Font Variant

CSS font variant property specifies how to set font variant of an element. It may be normal and small-caps.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p { font-variant: small-caps; }
6. h3 { font-variant: normal; }
7. **</style>**
8. **</head>**
9. **<body>**
10. **<h3>**This heading is shown in normal font.**</h3>**
11. **<p>**This paragraph is shown in small font.**</p>**
12. **</body>**
13. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont5)

Output:

### This heading is shown in normal font.

This paragraph is shown in small font.

## 6) CSS Font Weight

CSS font weight property defines the weight of the font and specify that how bold a font is. The possible values of font weight may be normal, bold, bolder, lighter or number (100, 200..... upto 900).

1. <!DOCTYPE html**>**
2. **<html>**
3. **<body>**
4. **<p** style="font-weight:bold;"**>**This font is bold.**</p>**
5. **<p** style="font-weight:bolder;"**>**This font is bolder.**</p>**
6. **<p** style="font-weight:lighter;"**>**This font is lighter.**</p>**
7. **<p** style="font-weight:100;"**>**This font is 100 weight.**</p>**
8. **<p** style="font-weight:200;"**>**This font is 200 weight.**</p>**
9. **<p** style="font-weight:300;"**>**This font is 300 weight.**</p>**
10. **<p** style="font-weight:400;"**>**This font is 400 weight.**</p>**
11. **<p** style="font-weight:500;"**>**This font is 500 weight.**</p>**
12. **<p** style="font-weight:600;"**>**This font is 600 weight.**</p>**
13. **<p** style="font-weight:700;"**>**This font is 700 weight.**</p>**
14. **<p** style="font-weight:800;"**>**This font is 800 weight.**</p>**
15. **<p** style="font-weight:900;"**>**This font is 900 weight.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssfont6)

Output:

**This font is bold.**

**This font is bolder.**

This font is lighter.

This font is 100 weight.

This font is 200 weight.

This font is 300 weight.

This font is 400 weight.

This font is 500 weight.

**This font is 600 weight.**

**This font is 700 weight.**

**This font is 800 weight.**

**This font is 900 weight.**

CSS Line Height

The **CSS line height property** is used *to define the minimal height of line boxes within the element*. It sets the differences between two lines of your content.

It defines the amount of space above and below inline elements. It allows you to set the height of a line of independently from the font size.

CSS line-height values

There are some property values which are used with CSS line-height property.

|  |  |
| --- | --- |
| **value** | **description** |
| normal | This is a default value. it specifies a normal line height. |
| number | It specifies a number that is multiplied with the current font size to set the line height. |
| length | It is used to set the line height in px, pt,cm,etc. |
| % | It specifies the line height in percent of the current font. |
| initial | It sets this property to its default value. |
| inherit | It inherits this property from its parent element. |

CSS line-height example

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h3.small {
6. line-height: 70%;
7. }
8. h3.big {
9. line-height: 200%;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<h3>**
15. This is a heading with a standard line-height.**<br>**
16. This is a heading with a standard line-height.**<br>**
17. The default line height in most browsers is about 110% to 120%.**<br>**
18. **</h3>**
19. **<h3** class="small"**>**
20. This is a heading with a smaller line-height.**<br>**
21. This is a heading with a smaller line-height.**<br>**
22. This is a heading with a smaller line-height.**<br>**
23. This is a heading with a smaller line-height.**<br>**
24. **</h3>**
25. **<h3** class="big"**>**
26. This is a heading with a bigger line-height.**<br>**
27. This is a heading with a bigger line-height.**<br>**
28. This is a heading with a bigger line-height.**<br>**
29. This is a heading with a bigger line-height.**<br>**
30. **</h3>**
31. **</body>**
32. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csslineheight1)

# CSS Margin

CSS Margin property is used to define the space around elements. It is completely transparent and doesn't have any background color. It clears an area around the element.

Top, bottom, left and right margin can be changed independently using separate properties. You can also change all properties at once by using shorthand margin property.

There are following CSS margin properties:

## CSS Margin Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| margin | This property is used to set all the properties in one declaration. |
| margin-left | it is used to set left margin of an element. |
| margin-right | It is used to set right margin of an element. |
| margin-top | It is used to set top margin of an element. |
| margin-bottom | It is used to set bottom margin of an element. |

## CSS Margin Values

These are some possible values for margin property.

|  |  |
| --- | --- |
| **Value** | **Description** |
| auto | This is used to let the browser calculate a margin. |
| length | It is used to specify a margin pt, px, cm, etc. its default value is 0px. |
| % | It is used to define a margin in percent of the width of containing element. |
| inherit | It is used to inherit margin from parent element. |

#### Note: You can also use negative values to overlap content.

## CSS margin Example

You can define different margin for different sides for an element.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.ex {
9. margin-top: 50px;
10. margin-bottom: 50px;
11. margin-right: 100px;
12. margin-left: 100px;
13. }
14. **</style>**
15. **</head>**
16. **<body>**
17. **<p>**This paragraph is not displayed with specified margin. **</p>**
18. **<p** class="ex"**>**This paragraph is displayed with specified margin.**</p>**
19. **</body>**
20. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssmargin1)

Output:

This paragraph is not displayed with specified margin.

This paragraph is displayed with specified margin.

## Margin: Shorthand Property

CSS shorthand property is used to shorten the code. It specifies all the margin properties in one property.

There are four types to specify the margin property. You can use one of them.

1. margin: 50px 100px 150px 200px;
2. margin: 50px 100px 150px;
3. margin: 50px 100px;
4. margin 50px;

## 1) margin: 50px 100px 150px 200px;

It identifies that:

**top** margin value is 50px

**right** margin value is 100px

**bottom** margin value is 150px

**left** margin value is 200px

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.ex {
9. margin: 50px 100px 150px 200px;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<p>**This paragraph is not displayed with specified margin. **</p>**
15. **<p** class="ex"**>**This paragraph is displayed with specified margin.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssmargin2)

Output:

This paragraph is not displayed with specified margin.

This paragraph is displayed with specified margin.

## 2) margin: 50px 100px 150px;

It identifies that:

**top** margin value is 50px

**left and right** margin values are 100px

**bottom** margin value is 150px

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.ex {
9. margin: 50px 100px 150px;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<p>**This paragraph is not displayed with specified margin. **</p>**
15. **<p** class="ex"**>**This paragraph is displayed with specified margin.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssmargin3)

Output:

This paragraph is not displayed with specified margin.

This paragraph is displayed with specified margin.

## 3) margin: 50px 100px;

It identifies that:

**top and bottom** margin values are 50px

**left and right** margin values are 100px

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.ex {
9. margin: 50px 100px;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<p>**This paragraph is not displayed with specified margin. **</p>**
15. **<p** class="ex"**>**This paragraph is displayed with specified margin.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssmargin4)

Output:

This paragraph is not displayed with specified margin.

This paragraph is displayed with specified margin.

## 4) margin: 50px;

It identifies that:

**top right bottom and left** margin values are 50px

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.ex {
9. margin: 50px;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<p>**This paragraph is not displayed with specified margin. **</p>**
15. **<p** class="ex"**>**This paragraph is displayed with specified margin.**</p>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssmargin5)

Output:

This paragraph is not displayed with specified margin.

This paragraph is displayed with specified margin.

# CSS Opacity

The CSS opacity property is used to specify the transparency of an element. In simple word, you can say that it specifies the clarity of the image.

In technical terms, Opacity is defined as degree in which light is allowed to travel through an object.

## How to apply CSS opacity setting

Opacity setting is applied uniformly across the entire object and the opacity value is defined in term of digital value less than 1. The lesser opacity value displays the greater opacity. Opacity is not inherited.

## CSS Opacity Example: transparent image

Let's see a simple CSS opacity example of image transparency.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. img.trans {
6. opacity: 0.4;
7. filter: alpha(opacity=40); /\* For IE8 and earlier \*/
8. }
9. **</style>**
10. **</head>**
11. **<body>**
12. **<p>**Normal Image**</p>**
13. **<img** src="rose.jpg" alt="normal rose"**>**
14. **<p>**Transparent Image**</p>**
15. **<img** class="trans" src="rose.jpg" alt="transparent rose"**>**
16. **</body>**
17. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssopacity1)

Output:

Normal Image



Transparent Image



#### Note 1: Chrome, Firefox, Opera, Safari, and IE9 use the opacity property for transparency. The opacity value ranges from 0.1 to 1.0. Lower value produces the greater opacity.

#### Note 2: The older versions of IE use filter: alpha(opacity=x). Here x value varies from 0 to 100. Lower value produces the greater opacity.

CSS Overflow

The **CSS overflow property** *specifies how to handle the content when it overflows* its block level container.

We know that every single element on a page is a rectangular box and the size, positioning and behavior of these boxes are controlled via CSS.

Let's take an example: If you don't set the height of the box, it will grow as large as the content. But if you set a specific height or width of the box and the content inside cannot fit then what will happen. The CSS overflow property is used to overcome this problem. It specifies whether to clip content, render scroll bars, or just display content.

CSS Overflow property values

|  |  |
| --- | --- |
| **Value** | **Description** |
| visible | It specifies that overflow is not clipped. it renders outside the element's box.this is a default value. |
| hidden | It specifies that the overflow is clipped, and rest of the content will be invisible. |
| scroll | It specifies that the overflow is clipped, and a scroll bar is used to see the rest of the content. |
| auto | It specifies that if overflow is clipped, a scroll bar is needed to see the rest of the content. |
| inherit | It inherits the property from its parent element. |
| initial | It is used to set the property to its initial value. |

CSS Overflow Example

Let's see a simple CSS overflow property.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. div.scroll {
6. background-color: #00ffff;
7. width: 100px;
8. height: 100px;
9. overflow: scroll;
10. }
12. div.hidden {
13. background-color: #00FF00;
14. width: 100px;
15. height: 170px;
16. overflow: hidden;
17. }
18. **</style>**
19. **</head>**
20. **<body>**
21. **<p>**The overflow property specifies what to do if the content of an element exceeds the size of the element's box.**</p>**
22. **<p>**overflow:scroll**</p>**
23. **<div** class="scroll"**>**You can use the overflow property when you want to have better control of the layout.
24. The default value is visible.**</div>**
25. **<p>**overflow:hidden**</p>**
26. **<div** class="hidden"**>**You can use the overflow property when you want to have better control of the layout.
27. The default value is visible.**</div>**
28. **</body>**
29. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssoverflow1)

Output:

The overflow property specifies what to do if the content of an element exceeds the size of the element's box.

overflow:scroll

You can use the overflow property when you want to have better control of the layout. The default value is visible.

overflow:hidden

You can use the overflow property when you want to have better control of the layout. The default value

CSS Padding

**CSS Padding property** is used *to define the space between the element content and the element border*.

It is different from CSS margin in the way that CSS margin defines the space around elements. CSS padding is affected by the background colors. It clears an area around the content.

Top, bottom, left and right padding can be changed independently using separate properties. You can also change all properties at once by using shorthand padding property.

CSS Padding Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| padding | It is used to set all the padding properties in one declaration. |
| padding-left | It is used to set left padding of an element. |
| padding-right | It is used to set right padding of an element. |
| padding-top | It is used to set top padding of an element. |
| padding-bottom | It is used to set bottom padding of an element. |

CSS Padding Values

|  |  |
| --- | --- |
| **Value** | **Description** |
| length | It is used to define fixed padding in pt, px, em etc. |
| % | It defines padding in % of containing element. |

CSS Padding Example

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p {
6. background-color: pink;
7. }
8. p.padding {
9. padding-top: 50px;
10. padding-right: 100px;
11. padding-bottom: 150px;
12. padding-left: 200px;
13. }
14. **</style>**
15. **</head>**
16. **<body>**
17. **<p>**This is a paragraph with no specified padding.**</p>**
18. **<p** class="padding"**>**This is a paragraph with specified paddings.**</p>**
19. **</body>**
20. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=csspadding1)

Output:

This is a paragraph with no specified padding.

This is a paragraph with specified paddings.

CSS Position

The **CSS position property** is used *to set position for an element*. it is also used to place an element behind another and also useful for scripted animation effect.

You can position an element using the top, bottom, left and right properties. These properties can be used only after position property is set first. A position element's computed position property is relative, absolute, fixed or sticky.

Let's have a look at following CSS positioning:

1. CSS Static Positioning
2. CSS Fixed Positioning
3. CSS Relative Positioning
4. CSS Absolute Positioning

1) CSS Static Positioning

This is a by default position for HTML elements. It always positions an element according to the normal flow of the page. It is not affected by the top, bottom, left and right properties.

2) CSS Fixed Positioning

The fixed positioning property helps to put the text fixed on the browser. This fixed test is positioned relative to the browser window, and doesn't move even you scroll the window.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. p.pos\_fixed {
6. position: fixed;
7. top: 50px;
8. right: 5px;
9. color: blue;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
15. **<p>**Some text...**</p><p>**Some text...**</p><p>**Some text...**</p><p>**........**</p><p>**.... ...**</p**
16. **><p>**........**</p><p>**........**</p><p>**........**</p><p>**........**</p>**
17. **<p>**........ **</p><p>**........**</p><p>**........**</p><p>**........**</p><p>**........**</p>**
18. **<p>**........**</p><p>**........**</p><p>**Some text...**</p><p>**Some text...**</p><p>**Some text...**</p>**
19. **<p** class="pos\_fixed"**>**This is the fix positioned text.**</p>**
20. **</body>**
21. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssposition1)

3) CSS Relative Positioning

The relative positioning property is used to set the element relative to its normal position.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h2.pos\_left {
6. position: relative;
7. left: -30px;
8. }
9. h2.pos\_right {
10. position: relative;
11. left: 30px;
12. }
13. **</style>**
14. **</head>**
15. **<body>**
16. **<h2>**This is a heading with no position**</h2>**
17. **<h2** class="pos\_left"**>**This heading is positioned left according to its normal position**</h2>**
18. **<h2** class="pos\_right"**>**This heading is positioned right according to its normal position**</h2>**
19. **<p>**The style "left:-30px" subtracts 30 pixels from the element's original left position.**</p>**
20. **<p>**The style "left:30px" adds 30 pixels to the element's original left position.**</p>**
21. **</body>**
22. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssposition2)

4) CSS Absolute Positioning

The absolute positioning is used to position an element relative to the first parent element that has a position other than static. If no such element is found, the containing block is HTML.

With the absolute positioning, you can place an element anywhere on a page.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h2 {
6. position: absolute;
7. left: 150px;
8. top: 250px;
9. }
10. **</style>**
11. **</head>**
12. **<body>**
13. **<h2>**This heading has an absolute position**</h2>**
14. **<p>** The heading below is placed 150px from the left and 250px from the top of the page.**</p>**
15. **</body>**
16. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssposition3)

All CSS Position Properties

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **property** | **description** | **values** |
| 1) | bottom | It is used to set the bottom margin edge for a positioned box. | auto, length, %, inherit |
| 2) | clip | It is used to clip an absolutely positioned element. | shape, auto, inherit |
| 3) | cursor | It is used to specify the type of cursors to be displayed. | url, auto, crosshair, default, pointer, move, e-resize, ne-resize, nw-resize, n-resize, se-resize, sw-resize, s-resize, w-resize, text, wait, help |
| 4) | left | It sets a left margin edge for a positioned box. | auto, length, %, inherit |
| 5) | overflow | This property is used to define what happens if content overflow an element's box. | auto, hidden, scroll, visible, inherit |
| 6) | position | It is used to specify the type of positioning for an element. | absolute, fixed, relative, static, inherit |
| 7) | right | It is used to set a right margin edge for a positioned box. | auto, length, %, inherit |
| 8) | top | It is used to set a top margin edge for a positioned box. | auto, length, %, inherit |
| 9) | z-index | It is used to set stack order of an element. | number, auto, inherit |

CSS Vertical Align

The CSS vertical align property is used to define the vertical alignment of an inline or table-cell box. It is the one of the self-explanatory property of CSS. It is not very easy property for beginners.

What it does

1. It is applied to inline or inline-block elements.
2. It affects the alignment of the element, not its content. (except table cells)
3. When it applied to the table cells, it affect the cell contents, not the cell itself.

CSS Vertical Align Values

|  |  |
| --- | --- |
| **value** | **description** |
| baseline | It aligns the baseline of element with the baseline of parent element. This is a default value. |
| length | It is used to increase or decrease length of the element by the specified length. negative values are also allowed. |
| % | It is used to increase or decrease the element in a percent of the "line-height" property. negative values are allowed. |
| sub | It aligns the element as if it was subscript. |
| super | It aligns the element as if it was superscript. |
| top | It aligns the top of the element with the top of the tallest element on the line. |
| bottom | It aligns the bottom of the element with the lowest element on the line. |
| text-top | the top of the element is aligned with the top of the parent element's font. |
| middle | the element is placed in the middle of the parent element. |
| text-bottom | the bottom of the element is aligned with the bottom of the parent element's font. |
| initial | It sets this property to Its default value. |
| inherit | inherits this property from Its parent element. |

CSS Vertical Align Example

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. img.top {
6. vertical-align: text-top;
7. }
8. img.bottom {
9. vertical-align: text-bottom;
10. }
11. **</style>**
12. **</head>**
13. **<body>**
14. **<p><img** src="good-morning.jpg" alt="Good Morning Friends"**/>** This is an image with a default alignment.**</p>**
15. **<p><img** src="good-morning.jpg" class="top" alt="Good Morning Friends"**/>** This is an image with a text-top alignment.**</p>**
16. **<p><img** src="good-morning.jpg" class="bottom" alt="Good Morning Friends"**/>** This is an image with a text-bottom alignment.**</p>**
17. **</body>**
18. **</html>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=cssverticalalign1)

Output:

 This is an image with a default alignment.

 This is an image with a text-top alignment.

 This is an image with a text-bottom alignment.