

# CSS Layout

Test the CSS3 layout options like multi-columns, grid or flexbox.

Test using **Internet Explorer 10**, **Internet Explorer 9**, **Google Chrome 14**, **Firefox 7**, **Safari 5** and **Opera 11**.

## CSS Multi-column Layout Module W3C Candidate Recommendation 12 April 2011 [\[W3C\]](#)

[\(http://www.w3.org/TR/css3-multicol/\)](http://www.w3.org/TR/css3-multicol/)

### Compatibility

| CSS property      | Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari  | Opera |
|-------------------|-------------|------------|---------|---------|---------|-------|
| columns           | ATHM        | KO         | -WEBKIT | KO      | -WEBKIT | ATHM  |
| column-width      | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-count      | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-gap        | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-rule       | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-rule-width | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-rule-style | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-rule-color | OK          | KO         | -WEBKIT | -MOZ    | -WEBKIT | OK    |
| column-span       | OK          | KO         | -WEBKIT | KO      | -WEBKIT | OK    |
| column-fill       | OK          | KO         | KO      | KO      | KO      | OK    |

Both **Internet Explorer 10** and **Opera** implement everything right, but not correctly evaluate the [pseudo-algorithm](http://www.w3.org/TR/css3-multicol/#pseudo-algorithm) .

### Column shorthand

This is a shorthand property for setting `column-width` and `column-count` .  
**NOTE** Omitted values are set to their initial values.

#### Code example

```
.example {  
  columns: 3 100px;  
}
```

### Column width

This property describes the width of columns in multicol elements.  
**auto**  
means that the column width will be determined by other properties (e.g., `column-count` , if it has a non-`auto` value).  
**<length>**  
describes the optimal column width. The actual column width may be wider (to fill the available

#### Code example

```
.example {  
  column-width: 100px;  
}
```

space), or narrower (only if the available space is smaller than the specified column width). Specified values must be greater than 0.

## Column count

This property describes the number of columns of a multicol element.

### auto

means that the number of columns will be determined by other properties (e.g., `column-count`, if it has a non-auto value).

### <length>

describes the optimal number of columns into which the content of the element will be flowed.

Values must be greater than 0. If both `column-width` and `column-count` have non-auto values, the integer value describes the maximum number of columns.

### Code example

```
.example {  
    column-count: 3;  
}
```

## Column gap

The `column-gap` property sets the gap between columns. If there is a column rule between columns, it will appear in the middle of the gap.

**DAFAULT** The normal value is UA-specific. A value of 1em is suggested.

**NOTE** Column gaps cannot be negative.

### Code example

```
.example {  
    column-count: 3;  
    column-gap: 40px;  
}
```

## Column rule shorthand

This property is a shorthand for setting `column-rule-width`, `column-rule-style`, and `column-rule-color` at the same place in the style sheet. Omitted values are set to their initial values.

### Code example

```
.example {  
    column-count: 3;  
    column-rule: 2px solid #333;  
}
```

## Column rule width

This property sets the width of the rule between columns. The `<border-width>` values are defined in [\[Border width - CSS21\]](http://www.w3.org/TR/CSS21/box.html#value-def-border-width) (<http://www.w3.org/TR/2010/WD-CSS2-20101207/box.html#value-def-border-width>).

**NOTE** Negative values are not allowed.

### Code example

```
.example {  
    column-count: 3;  
    column-gap: 20px;  
    column-rule: 2px solid #333;  
    column-rule-width: 5px;  
}
```

## Column rule style

The `column-rule-style` property sets the style of the rule between columns of an element.

The `<border-style>` values are defined in [\[Border style - CSS21\]](http://www.w3.org/TR/2010/WD-CSS2-20101207/box.html#value-def-border-style)

(<http://www.w3.org/TR/2010/WD-CSS2-20101207/box.html#value-def-border-style>) and the values are interpreted as in the the collapsing border model.

**NOTE** The `none` value forces the computed value of `column-rule-width` to be '0'.

### Code example

```
.example {  
    column-count: 3;  
    column-gap: 20px;  
    column-rule: 2px solid #333;  
    column-rule-style: dashed;  
}
```

## Column rule color

This property sets the color of the column rule. The `<color>` values are defined in [\[Colors and](#)

### Code example

## Column breaks

When content is laid out in multiple columns, the user agent must determine where column breaks are placed. The problem of breaking content into columns is similar to breaking content into pages.

Three new properties are introduced to allow column breaks to be described in the same properties as page breaks: `break-before`, `break-after`, and `break-inside`. These properties take the same values as `page-break-before`, `page-break-after`, and `page-break-inside` [Page Breaks - CSS21] (<http://www.w3.org/TR/2010/WD-CSS2-20101207/page.html#page-breaks>). In addition, some new keyword values are added [Column breaks - CSS3] (<http://www.w3.org/TR/css3-multicol/#column-breaks>).

**auto** Neither force nor forbid a page/column break before the generated box.

### Code example

```
.example .page-header {
  column-break-before: auto;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari  | Opera   |
|-------------|------------|---------|---------|---------|---------|
| DEFAULT     | DEFAULT    | DEFAULT | DEFAULT | DEFAULT | DEFAULT |

**always** Always force a page break before the generated box.

### Code example

```
.example .page-header {
  column-break-before: always;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**avoid** Avoid a page/column break before the generated box.

### Code example

```
.example .page-header {
  column-break-before: avoid;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**left** Force one or two page breaks before the generated box so that the next page is formatted as a left page.

### Code example

```
.example .page-header {
  column-break-before: left;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**right** Force one or two page breaks before the generated box so that the next page is formatted as a right page.

### Code example

```
.example .page-header {
  column-break-before: right;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**page** Always force a page break before the generated box.

Code example

```
.example .page-header {
  column-break-before: page;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**column** Always force a column break before the generated box.

Code example

```
.example .page-header {
  column-break-before: column;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**avoid-page** Avoid a page break before the generated box.

Code example

```
.example .page-header {
  column-break-before: avoid-page;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**avoid-column** Avoid a column break before the generated box.

Code example

```
.example .page-header {
  column-break-before: avoid-column;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**auto** Neither force nor forbid a page/column break after the generated box.

Code example

```
.example .page-header {
  column-break-after: auto;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari  | Opera   |
|-------------|------------|---------|---------|---------|---------|
| DEFAULT     | DEFAULT    | DEFAULT | DEFAULT | DEFAULT | DEFAULT |

**always** Always force a page break after the generated box.

Code example

```
.example .page-header {
  column-break-after: always;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**avoid** Avoid a page/column break after the generated box.

Code example

```
.example .page-header {
  column-break-after: avoid;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
|             |            |        |         |        |       |

**left** Force one or two page breaks after the generated box so that the next page is formatted as a left page.

Code example

```
.example .page-header {
  column-break-after: left;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**right** Force one or two page breaks after the generated box so that the next page is formatted as a right page.

Code example

```
.example .page-header {
  column-break-after: right;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari | Opera |
|-------------|------------|---------|---------|--------|-------|
| KO          | KO         | -WEBKIT | KO      | KO     | KO    |

**page** Always force a page break after the generated box.

Code example

```
.example .page-header {
  column-break-after: page;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**column** Always force a column break after the generated box.

Code example

```
.example .page-header {
  column-break-after: column;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**avoid-page** Avoid a page break after the generated box.

Code example

```
.example .page-header {
  column-break-after: avoid-page;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**avoid-column** Avoid a column break after the generated box.

Code example

```
.example .page-header {
  column-break-after: avoid-column;
}
```

Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

**auto** Neither force nor forbid a page/column break inside the generated box.

Code example

Compatibility

|  |  |  |  |  |  |
|--|--|--|--|--|--|
|  |  |  |  |  |  |
|--|--|--|--|--|--|

```
.example .page-header {
  column-break-inside: auto;
}
```

| Explorer 10 | Explorer 9 | Chrome  | Firefox | Safari  | Opera   |
|-------------|------------|---------|---------|---------|---------|
| DEFAULT     | DEFAULT    | DEFAULT | DEFAULT | DEFAULT | DEFAULT |

## avoid

Avoid a page/column break inside the generated box.

### Code example

```
.example .page-header {
  column-break-inside: avoid;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

## avoid-page

Avoid a page break inside the generated box.

### Code example

```
.example .page-header {
  column-break-inside: avoid-page;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

## avoid-column

Avoid a column break inside the generated box.

### Code example

```
.example .page-header {
  column-break-inside: avoid-column;
}
```

### Compatibility

| Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|-------------|------------|--------|---------|--------|-------|
| KO          | KO         | KO     | KO      | KO     | KO    |

## Column span

The `column-span` property makes it possible for an element to span across several columns. This property describes how many columns an element spans across. Values are:

**none** DEFAULT  
The element does not span multiple columns.

**all**  
The element spans across all columns. Content in the normal flow that appears before the element is automatically balanced across all columns before the element appears. The element establishes a new block formatting context.

NOTE An element that spans more than one column is called a **spanning element**.

### Code example

```
.example {
  columns: 3;
}
.example .page-header {
  column-span: all;
}
```

## Column fill

There are two strategies for filling columns: columns can either be balanced, or not. If columns are balanced, UAs should minimize the variation in column length. Otherwise, columns are filled sequentially and some columns may end up partially filled, or with no content at all. In any case, the user agent should try to honor the `widows` and `orphans` properties. The values are:

**balance** DEFAULT  
Balance content equally between columns, if possible.

**auto**  
Fills columns sequentially.

NOTE

- In continuous media, this property will only be consulted if the length of columns has been constrained. Otherwise, columns will automatically be balanced.
- In continuous media, this property does not have any effect in overflow columns (see below).
- In paged media, this property will only have effect on the last page the multicolumn element appears on.

### Code example

```
.example {
  columns: 3;
  column-fill: balance;
  overflow: hidden;
  height: 600px;
}
```

# CSS Grid Layout W3C Candidate Recommendation 12 April 2011 [\[W3C\] \(http://www.w3.org/TR/css3-grid-layout/\)](http://www.w3.org/TR/css3-grid-layout/)

There are a lot of TODOs in the Working Draft. Specs not stabilizes. Incompatible in most browsers.

## Compatibility

| CSS property                         | Explorer 10 | Explorer 9 | Chrome | Firefox | Safari | Opera |
|--------------------------------------|-------------|------------|--------|---------|--------|-------|
| display: grid / inline-grid          | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-columns                         | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-rows                            | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-row-span / grid-column-span     | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-row-sizing / grid-column-sizing | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-row-align / grid-column-align   | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-cell                            | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-template                        | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-flow                            | -MS         | KO         | KO     | KO      | KO     | KO    |
| grid-layer                           | -MS         | KO         | KO     | KO      | KO     | KO    |

## Grid declaration

A Grid element is declared by setting the display property.  
The values are:

### grid

A value of grid causes an element to display as a block-level Grid element.

### inline-grid

A value of inline-grid causes an element to display as an inline-level Grid element.

### HTML Code example

```
<div class="example">
  <header></header>
  <aside></aside>
  <section></section>
  <footer></footer>
</div>
```

### CSS Code example

```
.example {
  display: grid;
  grid-columns: 300px 1fr;
  grid-rows: 100px 1fr 50px;
}
.example header {
  grid-column: 1;
  grid-row: 1;
  grid-column-span: 2;
}
.example aside {
  grid-column: 1;
  grid-row: 2;
}
.example section {
  grid-column: 2;
  grid-row: 2;
}
.example footer {
  grid-column: 1;
  grid-row: 3;
  grid-column-span: 2;
}
```

## Grid columns

### Definition

### HTML Code example

```
<div class="example">
  <section class="block1">
</section>
  <section class="block2">
</section>
  <section class="block3">
</section>
  <section class="block4">
</section>
</div>
```

### CSS Code example

```
.example {
  display: grid;
  grid-columns: 150px 1fr;
  grid-rows: 50px 1fr 50px;
}
.example section {
  grid-column: start end;
  grid-row: start end;
}
```

## Flexible Box Layout Module W3C Working Draft, 22 March 2011 [\[W3C\]](#)

(<http://www.w3.org/TR/css3-flexbox/>)

There are a lot of TODOs in the Working Draft. Specs not stabilizes. Incompatible in all browsers.

## Compatibility

| CSS property                      | Explorer 10          | Explorer 9      | Chrome          | Firefox         | Safari          | Opera           |
|-----------------------------------|----------------------|-----------------|-----------------|-----------------|-----------------|-----------------|
| display: flexbox / inline-flexbox | <span>-MS-BOX</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> |
| flex-align                        | <span>-MS-BOX</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> |
| flex-direction                    | <span>-MS-BOX</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> |
| flex-order                        | <span>-MS-BOX</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> |
| flex-pack                         | <span>-MS-BOX</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> | <span>KO</span> |

**Internet Explorer 10** implements something similar with `display: -ms-box` in place of `display: flexbox` and properties like `-ms-box-*` instead of `flex-*`. You can see the example in [Internet Explorer Test Drive Hands On: CSS3 Flexible Box Layout](http://ie.microsoft.com/testdrive/Graphics/hands-on-css3/hands-on_flex.htm) ([http://ie.microsoft.com/testdrive/Graphics/hands-on-css3/hands-on\\_flex.htm](http://ie.microsoft.com/testdrive/Graphics/hands-on-css3/hands-on_flex.htm))

## Felxbox declaration

A flexbox item creates a new BFC. The margins of a flexbox item do not collapse with any other margin. Flexboxes "shrinkwrap" their contents by default (when their width or height properties are auto), similar to tables or floats.

### HTML Code example

```
<div class="example">
  <button class="button1">Button 1</button>
  <button class="button2">Button 2</button>
  <button class="button3">Button 3</button>
  <button class="button4">Button 4</button>
</div>
```

### CSS Code example

```
.example {
  display: flexbox;
  grid-direction: tb;
}
.example button {
  margin: 0 auto;
}
```



```
}  
  

```