columns Универсальное свойство, которое позволяет одновременно задать ширину и количество колонок многоколоночного текста.

*columns: column-width || column-count*

Комбинация свойств column-width и column-count. Порядок значения не имеет.

Ширина колонок 200 пикселей, количество задаётся браузером.

columns: 200px auto;

Две колонки, их ширина определяется браузером.

columns: 2;

column-count Определяет количество колонок в многоколоночном тексте.

Краткая информация

Значение по умолчанию auto

Наследуется Нет

Применяется К блочным элементам (кроме таблиц), ячейкам и элементам, у которых display установлен как inline-block

Анимируется Да

*column-count: <число> | auto*

1. <число> Целое число больше нуля, задающее оптимальное число колонок.
2. auto Число колонок вычисляется автоматически на основе других свойств (column-width, column-gap).

***.book {  
-moz-column-count: 2;  
-webkit-column-count: 2;  
column-count: 2; }***

column-width Задаёт оптимальную ширину колонки в многоколоночном тексте. Реальная ширина колонки может быть больше введённого значения, сокращение указанной ширины приводит к отмене нескольких колонок, текст при этом выстраивается в одну колонку.

*column-width: <размер> | auto*

1. <размер> Значение ширины колонки в абсолютных единицах CSS (например, пикселях).
2. auto Ширина колонок вычисляется автоматически на основе других свойств (column-count, column-gap).

**.book {  
 column-count: 3;  
 column-width: 200px;  
 -moz-column-count: 3;  
 -moz-column-width: 200px;  
 -webkit-column-count: 3;  
 -webkit-column-width: 200px; }**

column-rule-width В многоколоночном тексте задаёт толщину линий между колонок.

Значение по умолчанию medium

Наследуется Нет

Применяется К многоколоночным элементам

Анимируется Да

*column-rule-width: <размер> | thin | medium | thick*

Три переменные — thin (2 пикселя), medium (4 пикселя) и thick (6 пикселей) задают толщину линии. Для более точного значения толщину можно указывать в пикселях или других единицах.

***.book {  
 columns: 3 200px;  
 -webkit-columns: 3 200px;  
 -moz-columns: 3 200px;  
 column-rule: solid red;  
 -webkit-column-rule: solid red;  
 -moz-column-rule: solid red;  
 column-rule-width: thin;  
 -webkit-column-rule-width: thin;  
 -moz-column-rule-width: thin; }***

column-fill Определяет, как контент должен распределяться внутри колонок.

Краткая информация

Значение по умолчанию balance

Наследуется Нет

Применяется К многоколоночным элементам

Анимируется Нет

*column-fill: auto | balance | balance-all*

1. auto Колонки заполняются последовательно.
2. balance Контент равномерно распределяется между колонок. При печати это правило относится только к последней странице.
3. balance-all Контент равномерно распределяется между колонок. При печати это правило относится ко всем страницам.

***.book {  
 columns: 3 200px;  
-webkit-columns: 3 200px;  
 -moz-columns: 3 200px;  
 column-fill: balance-all;  
 -webkit-column-fill: balance-all;  
 -moz-column-fill: balance-all; }***

column-gap Задаёт расстояние между колонками в многоколоночном тексте.

*column-gap: <размер> | normal*

1. <размер> Значение ширины межколоночного интервала в единицах CSS (например, пикселах).
2. Normal Межколоночный интервал устанавливается браузером.

**.book** **{  
 column-count: 3;  
 column-width: 200px;  
 column-gap: 2em;  
 -moz-column-count: 3;  
 -moz-column-width: 200px;  
 -moz-column-gap: 2em;  
 -webkit-column-count: 3;  
 -webkit-column-width: 200px;  
 -webkit-column-gap: 2em; }**

column-rule В многоколоночном тексте отрисовывает линию между колонок и определяет её параметры.

Значение по умолчанию medium none

Наследуется Нет

Применяется К блочным элементам (кроме таблиц), ячейкам и элементам, у которых display установлен как inline-block

Анимируется Да

*column-rule: column-rule-width || column-rule-style || column-rule-color*

Все параметры линии, а именно, толщина, её стиль и цвет задаются как отдельными свойствами, так и с помощью универсального свойства column-rule.

Свойство column-rule-width определяет толщину линии. Для управления стилем линии предоставляется несколько значений column-rule-style.

***.book {  
 column-count: 3;  
 column-width: 200px;  
 column-rule: 1px solid #ccc;  
 -moz-column-count: 3;  
 -moz-column-width: 200px;  
 -moz-column-rule: 1px solid #ccc;  
 -webkit-column-count: 3;  
 -webkit-column-width: 200px;  
 -webkit-column-rule: 1px solid #ccc; }***

column-rule-color В многоколоночном тексте задаёт цвет линий между колонок. Если такой цвет явно не указан, то цвет текста в колонках и цвет линий между ними совпадает.

Значение по умолчанию Совпадает с цветом текста

Наследуется Нет

Применяется К многоколоночным элементам

Анимируется Да

*column-rule-color: <цвет>*

<цвет> устанавливает цвет линии, значение может быть в любом допустимом для CSS формате.

***.book {  
 columns: 3 200px;  
 -webkit-columns: 3 200px;  
 -moz-columns: 3 200px;  
 column-rule: 2px solid;  
 -webkit-column-rule: 2px solid;  
 -moz-column-rule: 2px solid;  
 column-rule-color: #ccc;  
 -webkit-column-rule-color: #ccc;  
 -moz-column-rule-color: #ccc; }***

column-rule-style Определяет стиль линий между колонок в многоколоночном тексте.

Краткая информация

Значение по умолчанию none

Наследуется Нет

Применяется К многоколоночным элементам

Анимируется Нет

*column-rule-style: none|hidden|dotted|dashed|solid|double|groove|ridge|inset|outset*

1. none Линия не отображается и значение её толщины обнуляется.
2. hidden Аналогично none.
3. dotted Линия состоящая из набора точек.
4. dashed Пунктирная линия, состоящая из серии коротких отрезков.
5. solid Сплошная линия.
6. double Двойная линия.
7. groove Создаёт эффект вдавленной линии
8. ridge Создаёт эффект рельефной линии
9. inset Псевдотрёхмерная линия.
10. outset Псевдотрёхмерная линия.

Вид указанных стилей представлен на рис. 1.

Стили линии

*Рис.1. Стили линии*

column-span Определяет, как должен отображаться элемент в многоколоночном тексте — занимать ширину всех колонок или только одну из них. Это свойство обычно применяют для заголовков текста или изображений, которые не должны разбиваться на колонки.

Значение по умолчанию none

Наследуется Нет

Применяется К блочным элементам (кроме таблиц), ячейкам и элементам, у которых display установлен как inline-block

Анимируется Нет

*column-span: none | all*

1. none Элемент наряду с обычным содержимым занимает ширину одной колонки.
2. all Элемент занимает все колонки.

***.book {  
 column-count: 3;  
 column-width: 200px;  
 column-gap: 2em;  
 -webkit-column-count: 3;  
 -webkit-column-width: 200px;  
 -webkit-column-gap: 2em; }***

***.book h1 {  
 column-span: all;  
 -webkit-column-span: all; }***

counter-increment Стилевое свойство counter-increment предназначено для увеличения значения счётчика приращений, который задается свойством counter-reset. Такой счётчик подсчитывает количество отображений элементов на странице и может выводиться с помощью свойства content и псевдоэлементов ::after и ::before. Это позволяет создавать списки (в том числе многоуровневые), в которых нумерация и вид задаются через стили.

Краткая информация

Значение по умолчанию none

Наследуется Нет

Применяется Ко всем элементам

Анимируется Нет

*counter-increment: [<переменная> | <число>?]+ none*

1. none Запрещает увеличение счётчика для текущего селектора.
2. <переменная> Задаёт одну или несколько переменных, для которых требуется изменить значение счётчика. Переменные разделяются между собой пробелом.
3. <число> Определяет значение приращения счётчика. По умолчанию оно равно 1. Допускается использовать только положительные целые числа, отрицательные целые числа и ноль.

Возможные сочетания значений свойств counter-reset и counter-increment показаны в таблице

Изменение нумерации списка

Код Результат

LI { list-style-type: none; } Список начинается с нуля.  
OL { counter-reset: list -1; } 0, 1, 2  
LI:before { counter-increment: list;  
content: counter(list) ". "; }

LI { list-style-type: none; } Выводятся все чётные числа.   
OL { counter-reset: list; } 2, 4, 6  
LI:before {  
counter-increment: list 2;  
content: counter(list) ". ";  
}

LI { list-style-type: none; } Выводятся все нечётные числа.  
OL { counter-reset: list -1; } 1, 3, 5  
LI:before {counter-increment: list list;  
content: counter(list) ". ";}

LI { list-style-type: none; } Список начинается с 10.  
OL { counter-reset: list 9; } 10, 11, 12  
LI:before {counter-increment: list;  
content: counter(list) ". ";}

***body { counter-reset: heading; /\* Инициируем счетчик \*/ }  
h2:before {counter-increment: heading;/\*Указываем идентификатор счетчика\*/  
content: "Глава " counter(heading) ". "; /\* Выводим текст перед содержимым тега <h2> \*/}***

***<h2>Теория ловли льва в пустыне</h2>  
<h2>Методы инверсной кинематики</h2>  
<h2>Ловля льва численными методами</h2>***

counter-reset Устанавливает переменную, в которой будет храниться счётчик отображений определенного элемента, а также начальное значение счётчика. Такой счётчик может выводиться с помощью свойства content и псевдоэлементов ::after и ::before.

Краткая информация

Значение по умолчанию none

Наследуется Нет

Применяется Ко всем элементам

Анимируется Нет

*counter-reset: [<переменная> | <число>?]+ | none*

* none
* Запрещает инициацию счётчика для текущего селектора.
* inherit
* Наследует значение родителя.
* <переменная> Задаёт одну или несколько переменных, в которых будет храниться значение счётчика. Значения разделяются между собой пробелом.
* <число> Начальное значение каждого идентификатора. По умолчанию равно 0.

***li { list-style-type: none; } /\* Убираем исходную нумерацию у списка \*/  
ol { counter-reset: list1; } /\* Инициируем счетчик \*/  
ol li:before {counter-increment: list1; /\* Увеличиваем значение счетчика \*/  
content: counter(list1) ". "; /\* Выводим число \*/ }  
ol ol { counter-reset: list2; } /\* Инициируем счетчик вложенного списка \*/  
ol ol li:before {counter-increment: list2; /\* Увеличиваем значение счетчика вложенного списка \*/  
content: counter(list1) "." counter(list2) ". "; /\* Выводим число \*/ }***