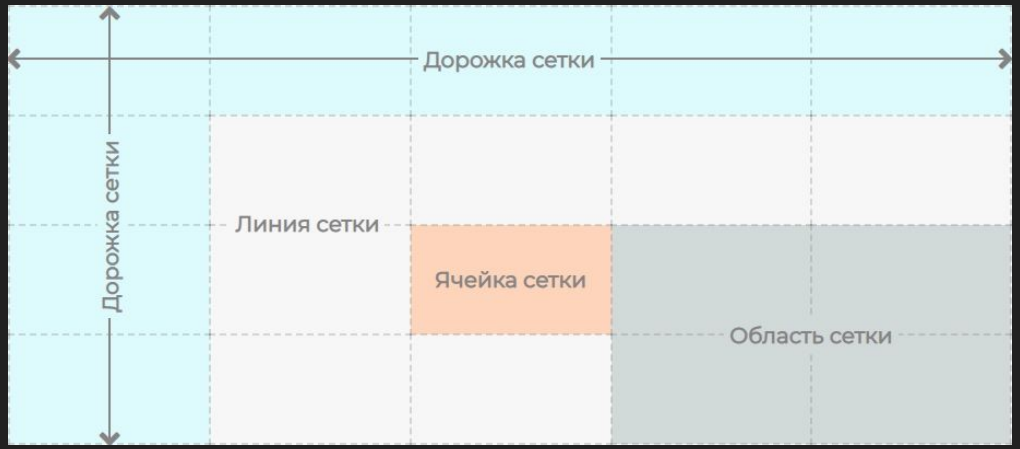
CSS - GRID

grid, flex, система колонок

Grid

Сетка (grid) представляет собой набор пересекающихся горизонтальных и вертикальных линий, делящих пространство grid-контейнера на области сетки, в которые могут быть помещены элементы сетки.



Создание контейнера-сетки: значения grid и inline-grid свойства display

Контейнер-сетка (grid container) — это блок, который устанавливает контекст форматирования по типу сетки, то есть создает область с сеткой, а дочерние элементы располагаются в соответствии с правилами компоновки сетки, а не блочной компоновки. Когда вы определяете контейнер сетки с помощью display: grid или display: inline-grid, вы создаете новый контекст форматирования для содержимого этого контейнера, который влияет только на дочерние элементы сетки.

Определение сетки

Когда вы создаете контейнер-сетку, сетка по умолчанию имеет один столбец и одну строку, которые занимают полный размер контейнера. Для разделения контейнера-сетки на столбцы и строки используются свойства grid-template-columns, grid-template-rows и grid-template-areas. С помощью этих свойств можно определить сетку явно.

Окончательная сетка может оказаться больше из-за элементов сетки, размещенных вне явной сетки; в этом случае будут созданы неявные дорожки, размер этих неявных дорожек будет определяться свойствами grid-auto-rows и grid-auto-columns.

Свойства grid и grid-template — это сокращенные обозначения, которые можно использовать для одновременной установки всех трех явных свойств сетки grid-template-columns, grid-template-rows и grid-template-areas. grid сбрасывает свойства, управляющие неявной сеткой, тогда как свойство grid-template оставляет их без изменений.

свойства grid-template-rows и grid-template-columns

Количество строк / столбцов задается с помощью свойств grid-template-rows и grid-template-columns.

Свойства не наследуются.

Гибкие размеры дорожек: единица измерения fr

fr — единица длины, которая позволяет создавать гибкие дорожки. Не является единицей измерения в обычном ее понимании, поэтому не может быть представлена или объединена с другими типами единиц в выражениях calc().

Функция minmax(min,max) определяет диапазон размеров, больше или равный min и меньше или равный max.

grid-template-rows, grid-template-columns

Значения:

`none`	Указывает, что свойство не создает явных дорожек сетки (хотя явные дорожки сетки все еще могут создаваться свойством `grid-template-areas`). При отсутствии явной сетки любые строки/столбцы будут генерироваться неявно, а их размер будет определяться свойствами `grid-auto-rows` и `grid-auto-columns`. Значение по умолчанию.
список дорожек / автоматический список дорожек	Устанавливает список дорожек в виде последовательности функций определения размера дорожек и названий линий сетки. Каждая функция определения размера дорожки может быть задана в единицах длины, как процент от размера контейнера-сетки или доля свободного пространства в сетке. Размер также может быть указан как диапазон с помощью нотации `minmax()`.

```
.grid-container {
  display: grid;
  grid-template-rows: 200px minmax(100px, 1fr);
}
```

Повтор строк и столбцов

Нотация repeat() представляет повторяющийся фрагмент списка дорожек, что позволяет записать в более компактной форме большое количество одинаковых по размерам столбцов или строк.

Первый аргумент задает количество повторений, которое может быть задано с помощью положительного целого числа или ключевых слов. Второй аргумент - размер повторяющейся дорожки.

```
repeat(число или auto-fill или auto-fit, повторяющаяся дорожка)
```

свойство grid-template-areas

Свойство grid-template-areas определяет именованные области сетки, которые не связаны с каким-либо конкретным элементом сетки, но на которые можно ссылаться из свойств размещения сетки. Синтаксис свойства обеспечивает визуализацию структуры сетки, облегчая понимание общего макета контейнера-сетки.

```
.grid-container {
  display: grid;
  grid-template-areas: "header header"
                       "sidebar content"
                       "sidebar content";
  grid-template-columns: 150px 1fr;
  grid-template-rows: 50px 1fr 50px;
}
header {
  grid-area: header;
}
aside {
  grid-area: sidebar;
}
main {
  grid-area: content;
}
```

Неявная сетка

Если элемент сетки расположен в строке или столбце, размер которых не определен явно grid-template-rows или grid-template-columns, создаются неявные дорожки сетки для его хранения. Это может произойти в случае, если строка или столбец оказались за пределами установленных размеров сетки.

grid-auto-columns, grid-auto-rows

Значения:

`auto`	Значение по умолчанию.
размер дорожки +	В качестве размера дорожки может использоваться любое значение, допустимое для задания размеров дорожек сетки.


```
.grid-container {
  max-width: 710px;
  display: grid;
  grid-template-columns: repeat(3,1fr);
  grid-template-rows: repeat(3,100px);
  grid-auto-rows: 50px;
}
.post-1 {
  grid-column: 1/3;
  grid-row: 1/3;
}
.post-2 {
  grid-column: 3;
  grid-row: 1;
}
.post-3 {
  grid-column: 3;
  grid-row: 2;
}
.post-4 {
  grid-column: 3;
  grid-row: 3;
}
.post-5 {
  grid-column: 2;
  grid-row: 3;
}
.post-6 {
  grid-column: 1;
  grid-row: 3;
}
```

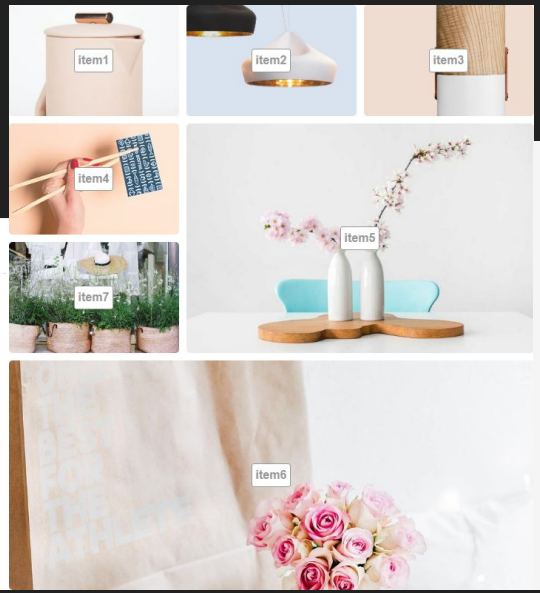
свойство grid-auto-flow

Элементы сетки, которые не размещены явно, автоматически помещаются в незанятое пространство в контейнере-сетке с помощью алгоритма автоматического размещения. Свойство grid-auto-flow управляет автоматическим размещением элементов сетки без явного положения. После заполнения явной сетки (или если явной сетки нет) автоматическое размещение также приведет к генерации неявных дорожек сетки.

grid-auto-flow

Значения:

`row`	Алгоритм автоматического размещения размещает элементы, заполняя каждую строку по очереди слева-направо (для LTR-языков), добавляя новые строки по мере необходимости. Значение по умолчанию.
`column`	Алгоритм размещает элементы, заполняя каждый столбец по очереди сверху-вниз, добавляя новые столбцы по мере необходимости.
`dense`	Алгоритм "плотной" укладки элементов. При необходимости может менять порядок следования элементов, заполняя пустые места более крупными элементами.




```css
.grid-container {
  max-width: 710px;
  margin: 10px auto;
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  grid-gap: 10px;
  grid-auto-rows: 200px;
  grid-auto-flow: dense;
}
.item5 {
  grid-column: span 2;
  grid-row: span 2;
}
.item6 {
  grid-column: span 3;
  grid-row: span 2;
}
```

свойство grid

Свойство grid задает все явные grid-template-rows, grid-template-columns и grid-template-areas и все неявные свойства сетки grid-auto-flow, grid-auto-rows и grid-auto-columns в одном объявлении. Оно не сбрасывает свойства row-gap/column-gap. Его синтаксис соответствует grid-template, а также дополнительной синтаксической форме для определения автоматического размещения элементов сетки:

Путем явного задания дорожек на одной оси (устанавливая grid-template-rows или grid-template-columns и задавая другим значение none), и задавая, как автоматически повторять дорожки на другой оси (устанавливая grid-auto-rows или grid-auto-columns и задавая другим auto).

Для grid-auto-flow также устанавливается одно из трех допустимых значений. Все остальные подсвойства grid сбрасываются к своим начальным значениям.

CSS

```css
/* значения для grid-template */
grid: none;
grid: "a" 100px "b" 1fr;
grid: [linename1] "a" 100px [linename2];
grid: "a" 200px "b" min-content;
grid: "a" minmax(100px, max-content) "b" 20%;
grid: 100px / 200px;
grid: minmax(400px, min-content) / repeat(auto-fill, 50px);

/* значения для grid-template-rows / [auto-flow && dense? ] grid-auto-columns? */
grid: 200px / auto-flow;
grid: 30% / auto-flow dense;
grid: repeat(3, [line1 line2 line3] 200px) / auto-flow 300px;
grid: [line1] minmax(20em, max-content) / auto-flow dense 40%;

/* значения для [ auto-flow && dense? ] grid-auto-rows? / grid-template-columns */
grid: auto-flow / 200px;
grid: auto-flow dense / 30%;
grid: auto-flow 300px / repeat(3, [line1 line2 line3] 200px);
grid: auto-flow dense 40% / [line1] minmax(20em, max-content);
```

Элементы сетки

Контейнер-сетка устанавливает новый контекст форматирования для элементов сетки, который обуславливает следующие особенности:

- Для элементов сетки блокируется их значение свойства display. Значение display: inline-block вычисляется в display: block, анонимные блоки текста также занимают всю ширину контейнера и образуют разрыв строки.
- Размер элемента сетки в пределах содержащего блока определяется его областью сетки.
- Расчеты элементов сетки для width: auto и height: auto зависят от их значений align-self:
- align-self: normal; - незамещаемые элементы заполняют область сетки, замещаемые элементы используют собственные размеры;
- align-self: stretch; - обе категории элементов заполняют область сетки;
- align-self: start/center и т.д. - незамещаемые элементы устанавливают размеры в соответствии со своим содержимым, замещаемые элементы используют собственные размеры.
- Поскольку соседние элементы сетки находятся в независимых областях сетки, то поля соседних элементов сетки margin не схлопываются.
- Браузеры по-разному обрабатывают процентные значения свойств margin и padding, поэтому не рекомендуется использовать их при задании значений этих свойств.
- Поля margin: auto; расширяются, поглощая свободное пространство в соответствующем измерении, поэтому могут использоваться для выравнивания элемента.

Размещение и переупорядочивание элементов сетки

Каждый элемент сетки связан с областью сетки, которая определяет содержащий блок для элемента сетки. Положение элементов сетки определяется расположением линий сетки и диапазоном сетки - количеством занимаемых дорожек сетки. По умолчанию элемент сетки занимает одну дорожку на каждой оси. Поэтому можно опустить значение grid-column-end или grid-row-end.

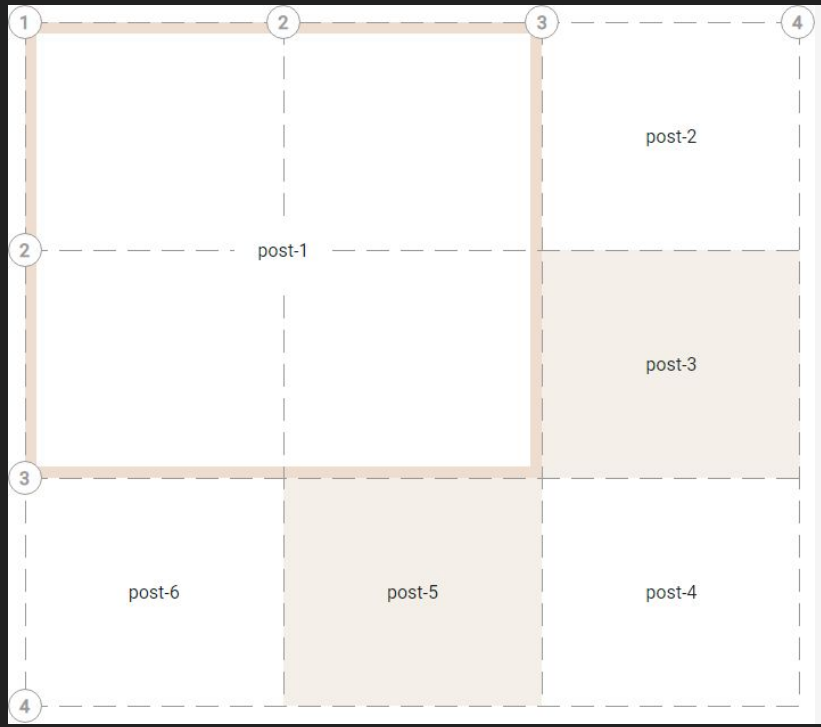
Свойства размещения на сетке - grid-row-start, grid-row-end, grid-column-start и grid-column-end и их краткая запись grid-row, grid-column и grid-area позволяют определить размещение элемента сетки, предоставив любую (или ноль) из следующих шести частей информации:

	Строка	Столбец
Начало	Начальная линия строки	Начальная линия столбца
Конец	Конечная линия строки	Конечная линия столбца
Диапазон	Диапазон строк	Диапазон столбцов

grid-row-start, grid-column-start, grid-row-end, grid-column-end

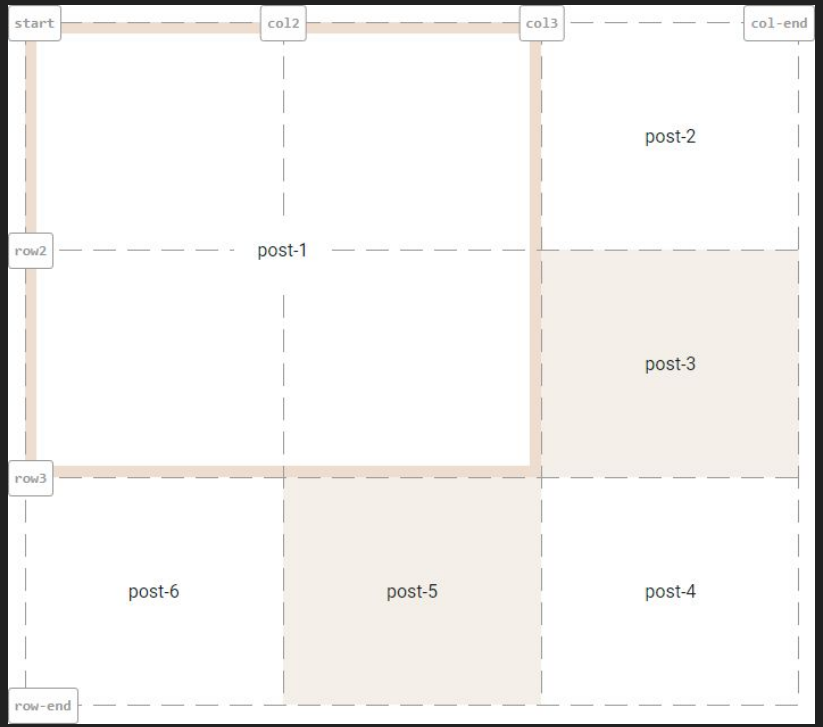
Значения:

`auto`	Свойство не влияет на размещение элемента сетки, указывая на автоматическое размещение или диапазон по умолчанию, равный единице.
имя линии	Начальная и конечная линия строки/столбца задаются в именованных линий сетки.
целое число и имя линии?	Начальная и конечная линия строки/столбца задаются с помощью целого числа (отрицательное порядковый номер линии сетки будет отсчитываться с противоположного края явной сетки) и (необязательно) имени линии.
`span` и целое число или имя линии	Ключевое слово `span` и целое положительное число/имя линии задают диапазон ячеек для размещения элемента сетки.




```css
.grid-container {
  display: grid;
  grid-template-rows: 200px 200px 200px;
  grid-template-columns: 1fr 1fr 1fr;
}
.post-1 {
  grid-row-start: 1;
  grid-row-end: 3;
  grid-column-start: 1;
  grid-column-end: 3;
}
.post-2 {
  grid-row-start: 1;
  grid-column-start: 3;
}
.post-3 {
  grid-row-start: 2;
  grid-column-start: 3;
}
.post-4 {
  grid-row-start: 3;
  grid-column-start: 3;
}
.post-5 {
  grid-row-start: 3;
  grid-column-start: 2;
}
.post-6 {
  grid-row-start: 3;
  grid-column-start: 1;
}
```

Именованные линии сетки



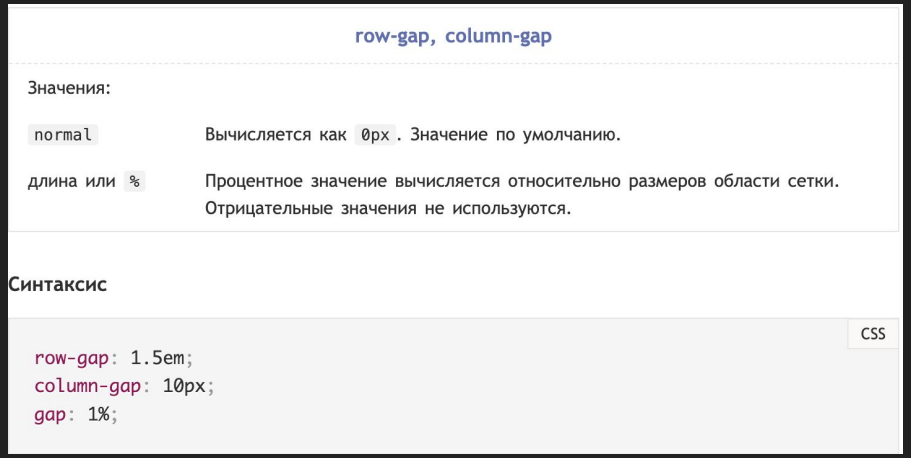

```
.grid-container {
  display: grid;
  grid-template-rows: [start] 200px [row2] 200px [row3] 200px [row-end];
  grid-template-columns: [start] 1fr [col2] 1fr [col3] 1fr [col-end];
}
.post-1 {
  grid-row-start: span 2;
  grid-column-start: span 2;
}
.post-2 {
  grid-row-start: start;
  grid-column-start: col3;
}
.post-3 {
  grid-row-start: row2;
  grid-column-start: col3;
}
.post-4 {
  grid-row-start: row3;
  grid-column-start: col3;
}
.post-5 {
  grid-row-start: row3;
  grid-column-start: col2;
}
.post-6 {
 grid-row-start: row3;
 grid-column-start: start;
}
```

Промежутки между элементами сетки: свойства row-gap, column-gap и gap

Свойства row-gap и column-gap (и их сокращенная запись gap), если они указаны в контейнере сетки, определяют промежутки между строками и столбцами сетки. При определении размера дорожки каждый промежуток рассматривается как дополнительная пустая дорожка указанного размера. Дополнительный промежуток также может быть добавлен между дорожками за счет свойств justify-content и align-content.

Промежутки добавляются только между двумя дорожками сетки, то есть они не добавляются перед первой и после последней дорожки.

Свойства не наследуются.



row-gap, column-gap

Значения:

`normal`	Вычисляется как `0px`. Значение по умолчанию.
длина или `%`	Процентное значение вычисляется относительно размеров области сетки. Отрицательные значения не используются.

Синтаксис

CSS

```
row-gap: 1.5em;
column-gap: 10px;
gap: 1%;
```

Система колонок

CSS3 columns описывает многоколоночный макет, который позволяет организовать содержимое так, чтобы оно занимало несколько вертикальных контейнеров, подобно газете или журналу.

Колонки могут содержать заголовки, текст, таблицы, картинки и любые другие inline-элементы.

THE OLD POST

NOVEMBER 24, 2012

15 Most Charming Small Towns In England

1. Berwick-upon-Tweed

Berwick was founded as an Anglo-Saxon settlement during the time of the Kingdom of Northumbria, which was annexed by England in the 10th century. The area was for more than 400 years central to historic border wars between the Kingdoms of England and Scotland, and several times possession of Berwick changed hands between the two kingdoms. The last time it changed hands was when Richard of Gloucester retook it for England in 1482. To this day many Berwickers feel a close affinity to Scotland.

Nowadays Berwick-upon-Tweed is much-visited for its highly visible history: medieval town walls, Elizabethan ramparts, 13th century castle ruins, its 17th century 'Old Bridge', town hall, Britain's earliest army barracks, England's northernmost hotel, amongst others.

2. Rye

Ancient Rye is all cobbled streets and tumbledown rows of houses by the sea. Originally part of the Cinque Ports Confederation, five strategic towns important for trade and military purposes in medieval times, today Rye is practically a living museum. Rye Castle, popularly known as Ypres Tower, was built in 1249 by Henry III to protect against frequent raids by the French; even older, the Norman-era St. Mary's Church looks over the town. Rye is also just a few minutes away from one of England's most famous beaches, Camber Sands, a two-mile-long playground for kitesurfers and beachlovers.

свойство column-width

Свойство column-width указывает минимальную ширину, которую должен занимать каждый столбец.

Свойство не наследуется.

Значения:

`auto`	Означает, что ширина столбца будет определяться другими свойствами (например, `column-count`, если оно имеет значение, отличное от `auto`). Значение по умолчанию.
длина	Ширина колонок задаётся в единицах длины, кроме `%`. Фактическая ширина столбца может быть больше (для заполнения доступного пространства) или уже (только если доступное пространство меньше указанной ширины столбца). Отрицательные значения не допускаются. Используемые значения будут ограничены минимум `1px`.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-width: auto;
column-width: 100px;
column-width: 10em;
column-width: 3.3vw;
column-width: inherit;
column-width: initial;
```

свойство column-count

Свойство column-count описывает количество колонок, а их ширина будет рассчитываться, исходя из ширины доступного пространства. Если одновременно с column-count задается column-width, то значение column-count будет считаться максимальным числом колонок.

Свойство не наследуется.

Значения:

`auto`	Означает, что количество столбцов будет определяться другим свойством, например, `column-width`, если оно также не имеет значение `auto`. Значение по умолчанию.
число	Описывает максимальное количество колонок. Значение задаётся целым числом, должно быть больше `0`.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-count: auto;
column-count: 2;
column-count: inherit;
column-count: initial;
```

Свойство columns

Свойство columns — это сокращенное свойство для установки column-width и column-count. Опущенные значения устанавливаются в их начальные значения.

Свойство не наследуется.

```
columns: 12em;      /* column-width: 12em; column-count: auto */
columns: auto 12em; /* column-width: 12em; column-count: auto */
columns: 2;         /* column-width: auto; column-count: 2 */
columns: 2 auto;    /* column-width: auto; column-count: 2 */
columns: auto;      /* column-width: auto; column-count: auto */
columns: auto auto; /* column-width: auto; column-count: auto */
columns: inherit;
columns: initial;
```

свойство column-gap

Свойство column-gap определяет разрыв между колонками. Если для колонок установлена разделительная линия с помощью свойства column-rule, то эта линия будет расположена посередине промежутка, а ее ширина не изменит общую ширину.

Свойство не наследуется.

Значения:

длина	Промежуток между колонками задается в единицах длины. Значения не могут быть отрицательными. Процентное значение может быть удалено из спецификации.
`normal`	Эквивалентно `1em`. Значение по умолчанию.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-gap: normal;
column-gap: 3px;
column-gap: 2.5em;
column-gap: 3%;
column-gap: inherit;
column-gap: initial;
```

свойство column-rule-color

Свойство column-rule-color определяет цвет разделительной линии.

Свойство не наследуется.

Значения:

цвет	Цвет линии задается с помощью допустимых значений цвета. Значение по умолчанию `currentColor`.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-rule-color: pink;
column-rule-color: #D71C3B;
column-rule-color: rgb(192, 56, 78);
column-rule-color: transparent;
column-rule-color: hsla(0, 100%, 50%, 0.6);
column-rule-color: inherit;
column-rule-color: initial;
```

свойство column-rule-style

Свойство column-rule-style устанавливает стиль разделительной линии.

Свойство не наследуется.

```
column-rule-style: none;
column-rule-style: hidden;
column-rule-style: dotted;
column-rule-style: dashed;
column-rule-style: solid;
column-rule-style: double;
column-rule-style: groove;
column-rule-style: ridge;
column-rule-style: inset;
column-rule-style: outset;
column-rule-style: inherit;
column-rule-style: initial;
```

Значения:

`none`	Значение вычисляется в `0`. Значение по умолчанию.
`hidden`	Аналогично со значением `none`, линия скрыта.
`dotted`	Отображает линию набором квадратных точек.
`dashed`	Отображает линию как последовательность из тире.
`solid`	Обычная линия.
`double`	Отображает разделительную линию в виде двух параллельных тонких линий, расположенных на некотором расстоянии между собой. Толщина разделительной линии не указывается, но сумма линий и промежутка между ними равна значению `column-rule-width`.
`groove`	Отображает линию объемной, вдавленной в полотно. Это достигается путем создания тени из двух цветов, один из которых темнее, другой — светлее.
`ridge`	Отображает разделительную линию объемной, т.е. эффект, противоположный `groove`.
`inset`	Отображает сплошную линию цветом темнее, чем заданный цвет линии.
`outset`	Отображает сплошную линию цветом, заданным свойством `column-rule-color`.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

свойство column-rule-width

Свойство column-rule-width устанавливает ширину разделительной линии. Отрицательные значения не допускаются. Не работает без свойства column-rule-style.

Свойство не наследуется.

Значения:

`thin`	Тонкая линия.
`medium`	Значение по умолчанию. Средняя толщина линии.
`thick`	Утолщенная линия.
длина	Ширина разделительной линии задается в единицах длины.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-rule-width: thin;
column-rule-width: medium;
column-rule-width: thick;
column-rule-width: 1px;
column-rule-width: 2.5em;
column-rule-width: inherit;
column-rule-width: initial;
```

свойство column-rule

Свойство column-rule является сокращенной записью свойств column-rule-width column-rule-style column-rule-color.

Свойство не наследуется.

```
column-rule: dotted;
column-rule: solid 8px;
column-rule: solid blue;
column-rule: thick inset blue;
column-rule: inherit;
column-rule: initial;
```

свойство column-span

Свойство column-span позволяет элементу охватывать несколько столбцов. Указывается не для блока-контейнера, а для для конкретного элемента внутри, например, для заголовка.

В будущем будет возможно указать количество колонок для охвата, подобно атрибуту colspan, который может быть применен к ячейке таблицы, но в спецификации CSS3 есть только два возможных значения: none и all.

Свойство не работает по умолчанию в Firefox. Пользователь должен явно включить функцию, в layout.css.column-span.enabled должно быть установлено значение true. Чтобы изменить настройки в Firefox, зайдите в about:config.

Свойство не наследуется.

Значения:

`none`	Содержимое элемента отображается в пределах одной колонки. Значение по умолчанию.
`all`	Элемент охватывает все колонки. Колонка разбивается в том месте, где отображается элемент.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-span: none;
column-span: all;
column-span: inherit;
column-span: initial;
```

свойство column-fill

Свойство column-fill контролирует заполнение колонок содержимым. Существует две стратегии заполнения колонок: колонки могут быть выровнены по высоте или нет. Если колонки выровнены, браузеры должны попытаться минимизировать изменения высоты колонки, учитывая при этом вынужденные разрывы, widows, orphans и другие свойства, которые могут влиять на высоту колонок. Если колонки не выровнены, они заполняются последовательно, некоторые из них могут быть частично заполнены или вообще не заполнены.

Свойство не наследуется.

Значения:

`auto`	Заполняет колонки последовательно.
`balance`	Отображает содержимое одинаково во всех колонках. Значение по умолчанию.
`balance-all`	Выравнивает содержимое равномерно между колонками, насколько это возможно.
`initial`	Устанавливает значение свойства в значение по умолчанию.
`inherit`	Наследует значение свойства от родительского элемента.

```
column-fill: auto;
column-fill: balance;
column-fill: balance-all;
column-fill: inherit;
column-fill: initial;
```

Переполнение

Переполнение внутри многоколоночных контейнеров

За исключением случаев, когда это может привести к разрыву колонки, содержимое, которое выходит за границы колонки, выходит за ее границы и не обрезается. Это касается, в первую очередь, изображений. Чтобы решить эту проблему, нужно установить для изображений следующие свойства:

```
img {
  display: block;     /*убираем нижний отступ под картинкой*/
  width: 100%;        /*растягиваем изображение на всю ширину блока-контейнера*/
}
```

Переполнение

Содержимое и разделительные линии, которые выходят за рамки колонок по краям многоколоночного контейнера, обрезаются в соответствии со свойством overflow.

Многоколоночный контейнер может иметь больше колонок, чем у него есть для этого места из-за ограничения высоты колонок (например, с помощью height или max-height) и явных разрывов колонок. В этом случае дополнительные колонки создаются в направлении строки, перемещаясь на следующие страницы.

FLEX - пример 12 колончатой системы

```
.container {
    display: flex;
    flex-wrap: wrap;
    gap: 8px;
  }

.item {
    flex-basis: calc(
      100% / 12 - 8px
    ); /* Устанавливаем ширину элемента в зависимости от 12 колонок */
    border: 1px solid #000; /* Добавляем рамку для наглядности */
    box-sizing: border-box; /* Учитываем рамку в расчете ширины элемента */
    text-align: center; /* Центрируем содержимое по горизонтали */
    height: 100px;
    background-color: blue;
 }
```

Ресурсы

GRID статья на русском языке - ТЫК

GRID документация - ТЫК

FLEX статья на русском языке - ТЫК

Колонки (устарели) статья на русском языке - ТЫК