

Positioning

Vira Huskova

Для вёрстки страниц часто используются два основных инструмента

- *позиционирование (positioning)*;
- *свободное перемещение (floating)*.

CSS-позиционирование позволяет указать, где появится блок элемента, а свободное перемещение перемещает элементы к левому или правому краю блока-контейнера, позволяя остальному содержимому «обтекать» его.

В тот момент, когда браузер загружает веб-страницу, происходит рендеринг элементов (заголовки, строки, параграфы, блоки div и т. д.) в той последовательности, в которой они встречаются в разметке HTML-документа. Позиционирование упорядочивает процесс отображения всех элементов на странице сайта.

Типы позиционирования

Свойство **position** позволяет точно задать новое местоположение блока относительно того места, где он находился бы в нормальном потоке документа.

По умолчанию все элементы располагаются последовательно один за другим в том порядке, в котором они определены в структуре HTML-документа.

Значение по умолчанию: position: static;

`position: static;`

Элементы отображаются как обычно.

Использование свойств **left, top, right** и **bottom** не приводит к каким-либо **результатам**.

Используется для очистки любого другого значения позиционирования.

`position: relative;`

Относительно позиционированный элемент сдвигается со своего обычного места в разных направлениях относительно границ родительского контейнера, а пространство, которое он занимал, не исчезает.

При этом такой элемент может перекрывать другое содержимое на странице.

Если для относительно позиционированного элемента одновременно задать свойства **top** и **bottom** или **left** и **right**, то в первом случае сработает только **top**, во втором — **left**.

Добавление свойств left, top, right и bottom изменяет позицию элемента и сдвигает его в ту или иную сторону от первоначального расположения.

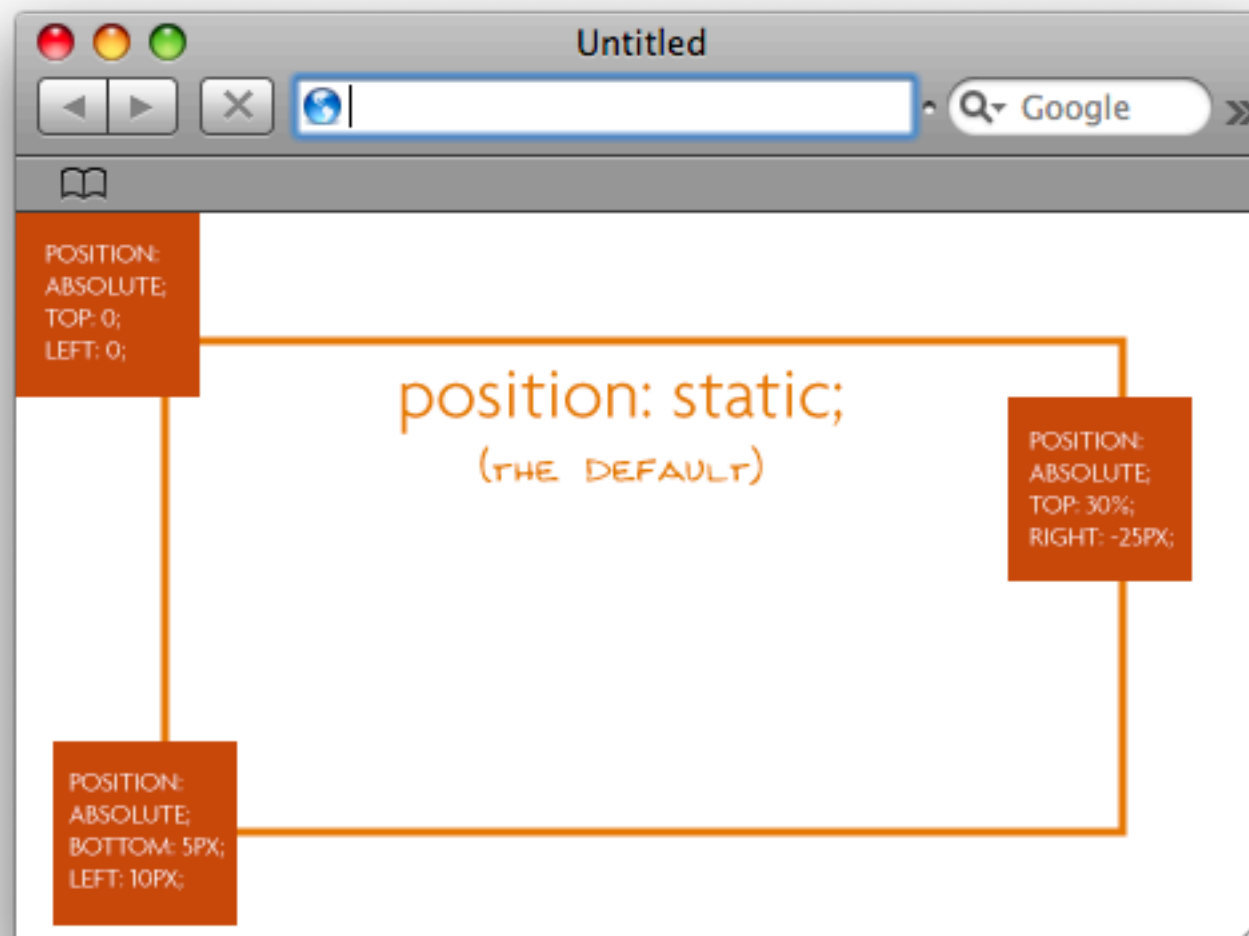
Относительное позиционирование позволяет задавать **z-index** для элемента, а также абсолютно позиционировать дочерние элементы внутри блока.

<https://codepen.io/lostsou41216364/pen/LgJBXN>

position: absolute;

Абсолютно позиционированный элемент полностью удаляется из потока документа и позиционируется относительно границ его блока-контейнера (другого элемента или окна браузера).

Блок-контейнер для абсолютно позиционированного элемента — ближайший элемент-предок, значение свойства **position** которого не равно **static**.



Местоположение краёв элемента определяется с помощью свойств смещения.

Пространство, которое занимал такой элемент, схлопывается, как будто элемента не существовало на странице.

Абсолютно позиционированный элемент может перекрывать другие элементы или быть перекрытым ими (за счёт свойства **z-index**).

Любой абсолютно позиционированный элемент генерирует блок, то есть принимает значение **display: block;**

<https://codepen.io/lostsou41216364/pen/oaPPzW>

`position: initial;`

Устанавливает значение свойства в значение по умолчанию.

`position: inherit;`

Наследует значение свойства от родительского элемента.

`position:fixed;`

По своему действию это значение близко к `absolute`, но в отличие от него привязывается к указанной свойствами `left`, `top`, `right` и `bottom` точке на экране и не меняет своего положения при прокрутке веб-страницы.

Браузер Firefox вообще не отображает полосы прокрутки, если положение элемента задано фиксированным, и оно не помещается целиком в окно браузера.

В браузере Opera хотя и показываются полосы прокрутки, но они никак не влияют на позицию элемента.

- Фиксировано позиционированный элемент всегда игнорирует родительские элементы и располагается относительно окна браузера;
- При пролистывании страницы элемент не смещается.

Зачастую фиксированное позиционирование применяется для создания вкладок, меню, заголовков и любых прочих элементов, которые должны быть постоянно видны пользователю сайта.

<https://codepen.io/lostsou41216364/pen/XxPoqj>

<https://codepen.io/lostsou41216364/pen/oaPJPp>

Site ex

<https://www.awwwards.com/20-inspirational-examples-of-interactive-maps-and-street-view-experiences-in-web-design.html>

<http://clown-charly.de/>

`position: sticky;`

Элемент с липкой позицией – это элемент, значение вычисленного `position` которого является `sticky`.

Он рассматривается как относительно позиционированный до тех пор, пока содержащий его блок не пересечет указанный порог (например, установка `top` для значения, отличного от `auto`) внутри его корня потока (или в контейнере, в котором он прокручивается), после чего он обрабатывается как «застывший» до тех пор, пока не встретит противоположный край содержащего его блока.

На то, в каком месте элемент прикрепится и где отцепиться, также влияют свойства `padding`, `margin` и `transform`. Размещение элемента над другими элементами правится с помощью свойства `z-index`.

После того, как свойство `position` примет значение `sticky`, размер элемента не изменится, а соседние элементы не сдвинутся на освободившееся пространство.

<https://codepen.io/SitePoint/pen/hewAv>

<https://codepen.io/rachelandrew/pen/NbPamG>

Смещение

Свойства описывают смещение относительно ближайшей стороны блока-контейнера.

Задаются для элементов, для которых значение свойства **position** не равно **static**. Могут принимать как положительные, так и отрицательные значения.

top, right, bottom, left

auto Значение по умолчанию. Вычисляемое значение свойства равно нулю.

длина Смещение задаётся в единицах длины.

% Процентные значения вычисляются относительно **высоты блока-контейнера** для **top** и **bottom** и **ширины блока-контейнера** для **right** и **left**.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

Для свойства **top** положительные значения перемещают верхний край позиционируемого элемента ниже, а отрицательные — выше верхнего края его блока-контейнера.

Для свойства **left** положительные значения сдвигают край позиционируемого элемента вправо, а отрицательные значения — влево. То есть, **положительные значения смещают элемент внутрь блока-контейнера**, а отрицательные — за его пределы.

Внутреннее позиционирование

Для блока-контейнера абсолютно позиционированного элемента задаётся свойство **position: relative** без смещений.

Это позволяет позиционировать элементы внутри элемента-контейнера.

<https://codepen.io/lostsou41216364/pen/ReYqZG>

Проблемы при позиционировании

1. Если ширина или высота элемента с **position: absolute;** получила значение **auto**, то её значение будет определяться шириной или высотой содержимого элемента.

Если ширина или высота объявлена явно, то именно это значение и будет присвоено.

<https://codepen.io/lostsou41216364/pen/VEGqZp>

<https://codepen.io/lostsou41216364/pen/QZVzKo>

2. Если внутри блока с **position: absolute** расположены элементы, для которых задано обтекание **float**, то высота этого элемента будет равна высоте самого высокого из этих элементов.

3. Для элемента с `position: absolute` или `position: fixed` нельзя одновременно устанавливать свойство `float`, а для элемента с `position: relative` — можно.

4. Если предок позиционированного элемента является блочным элементом, то блок-контейнер формируется областью содержимого, ограниченной рамкой (`border`). Если предок — строковый элемент, блок-контейнер формируется внешней границей его содержимого. Если предка нет, блоком-контейнером является элемент `body`.

Z-index

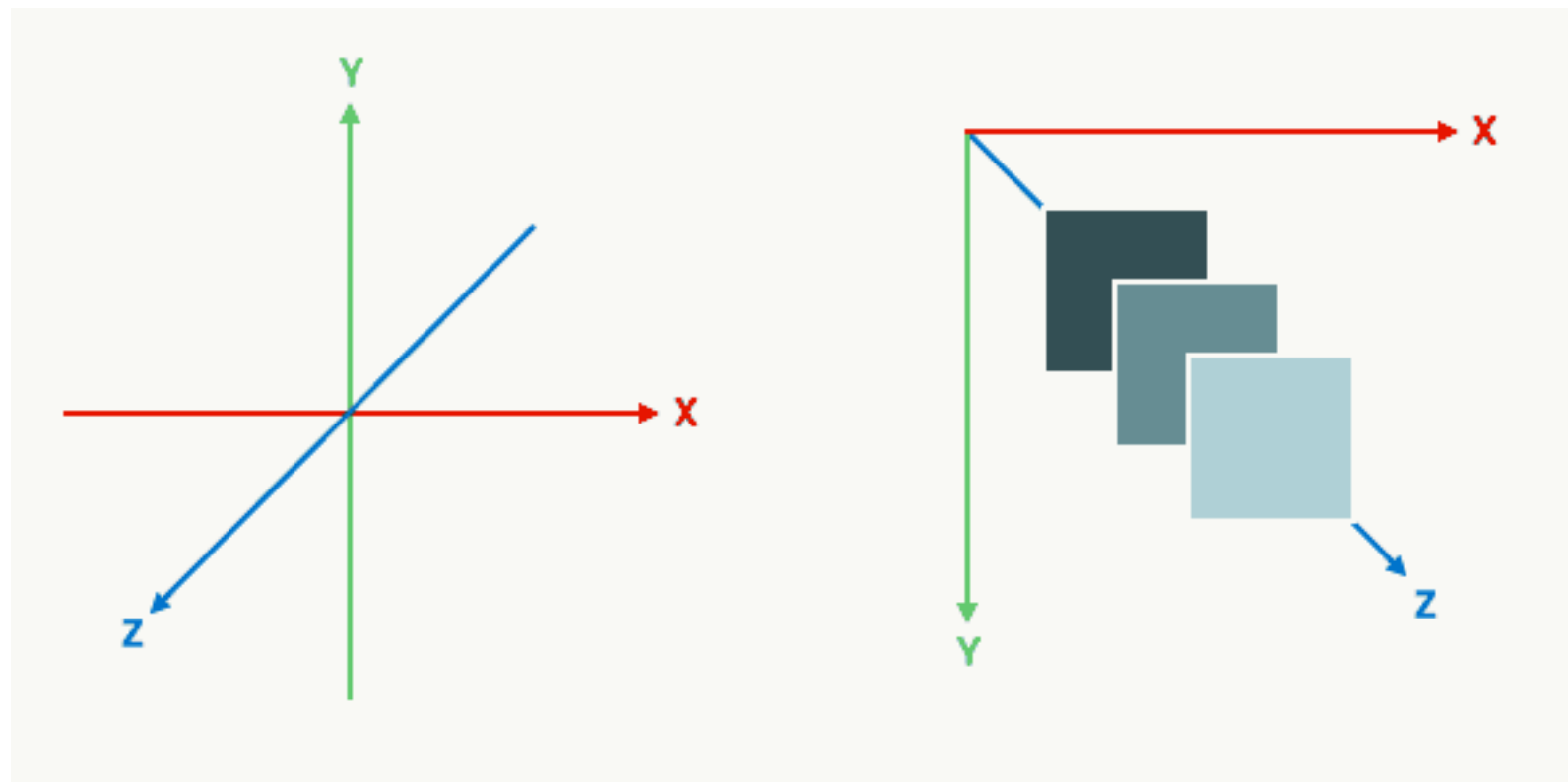
Свойство **z-index** управляет порядком наложения позиционированных элементов в случае, когда они накладываются друг на друга (пересекаются).

Свойство принимает целые положительные и отрицательные значения.

В нормальном потоке **position: static** элементы располагаются последовательно один за другим в том порядке, в котором они определены в html-документе.

По умолчанию установлен **z-index: auto;**.

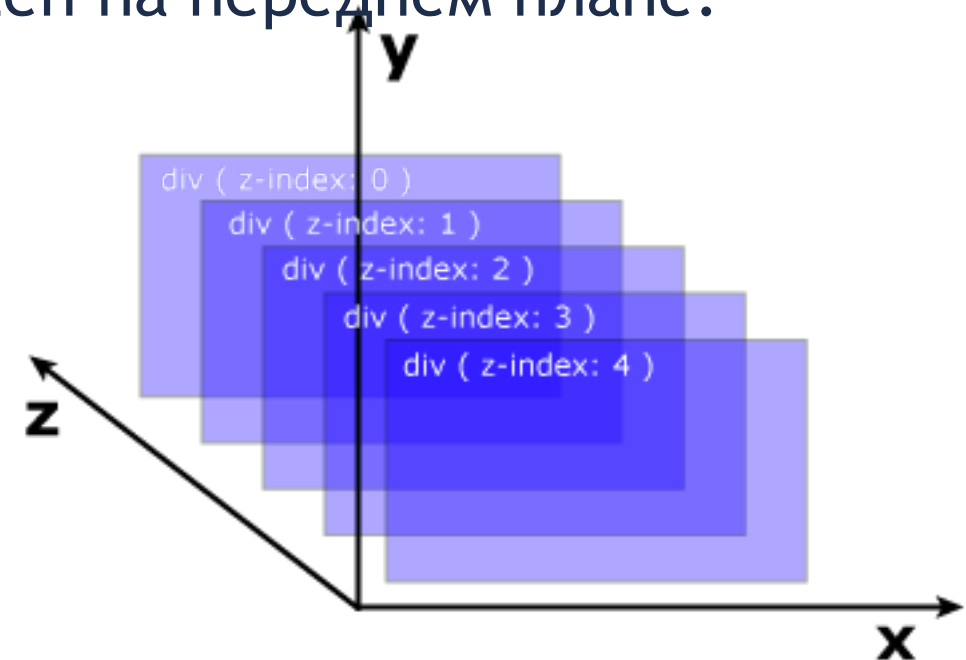
Свойство **z-index** задает порядок расположения элементов вдоль оси Z. В обычной ситуации элементы с высоким значением индекса будут перекрывать элементы с меньшим значением и значением **auto**, располагаясь на переднем плане.



Контекст наложения

Если для элементов свойства **z-index** и **position** не заданы явно, контекст наложения равен порядку их расположения в исходном коде. Если для элементов не задан **z-index**, браузер отображает элементы на странице в следующем порядке:

1. Корневой элемент **<html>**, который содержит все элементы веб-страницы.
2. Блочные элементы, не плавающие и не позиционированные.
3. Плавающие **float** не позиционированные элементы в порядке их расположения в исходном коде.
4. Строковые не позиционированные элементы (текст, изображения).
5. Позиционированные **position** элементы в порядке их следования в исходном коде. Последний из них будет расположен на переднем плане.



Свойство **z-index** создает новый контекст наложения. Оно позволяет изменить порядок наложения **позиционированных элементов**.

Элементы будут отображаться на странице в следующем порядке (если для них не заданы свойства, влияющие на контекст наложения — **opacity, filter, transform**):

Корневой элемент `<html>`, который содержит все элементы веб-страницы.

Позиционированные элементы с отрицательным значением `z-index`.

Блочные элементы, не плавающие и не позиционированные.

Плавающие `float` не позиционированные элементы в порядке их расположения в исходном коде.

Строковые не позиционированные элементы (текст, изображения).

Позиционированные элементы со значениями `z-index: 0;` и `z-index: auto;`.

Позиционированные элементы с положительными значениями `z-index`. Высокое значение индекса отобразит элемент на переднем плане. Элементы с равными значениями `z-index` будут отображаться согласно их расположения в исходном коде.

корневой элемент

`z-index < 0`

блочные элементы

`float:left; float:right;`

строковые элементы

`position ≠ static`
`z-index:0; z-index:auto;`

`position ≠ static`
`z-index > 0`

CSS background

CSS-фон — свойства, добавляющие фон для любого HTML-элемента.

Каждый элемент имеет фоновый слой, который может быть прозрачным (по умолчанию), цветной заливкой и изображением. В качестве изображения может выступать градиент или картинка. Для одного элемента можно задать несколько фоновых изображений.

Фон не отображается у пустых элементов с нулевой высотой. Свойства фона не наследуются, но фон родительского блока всегда будет виден.

background-color

цвет Цвет фона можно задать разными способами: HEX, RGB, RGBA, HSL, HSLA, а также с помощью 140 кроссбраузерных названий цветов.

transparent Значение по умолчанию. Задаёт прозрачный фон для элемента.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

```
div {background-color: papayawhip;}
```

```
h1 {background-color: palevioletred;}
```

```
p {background-color: peachpuff;}
```

```
span {background-color: peru;}
```

Заголовок

Какой-то текст, а это **СЛОВО** выделено другим цветом.

background-image

Свойство устанавливает изображение в качестве фона для элемента.

Фоновым изображением может быть картинка или градиент, который задаётся

с помощью функций **background-image: linear-**

gradient(), **background-image: radial-gradient()**

или с помощью функций повтора градиента **background-image:**

repeating-linear-gradient(), **background-image: repeating-**

radial-gradient(). Не наследуется.

url('URL') Абсолютный или относительный адрес изображения.

none Значение по умолчанию, также удаляет изображение у элемента из группы элементов, для которых установлено фоновое изображение.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

```
div {  
    background-image: url(www.google.com/images/flower.png) ;  
}  
table {  
    background-image: radial-gradient(farthest-corner at 50%  
50%, white, #DCECF8) ;  
}  
p {  
    background-image: linear-gradient(to top left, white,  
lightblue) ;  
}
```

background-repeat

Свойство определяет, каким образом будет повторяться фоновый рисунок. Не наследуется.

repeat Весь фон страницы будет заполнен фоновым рисунком. Если при этом задать background-position, то повтор будет осуществляться с указанной позиции. Значение по умолчанию.

no-repeat Фоновое изображение не будет повторяться.

repeat-x Фоновый рисунок повторяется от левого до правого края веб-страницы по верхнему краю страницы.

repeat-y Фоновый рисунок повторяется от верхнего до нижнего края веб-страницы по левому краю страницы.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

Позиционирование фоновых изображений background-position

Свойство управляет точным расположением фонового изображения.

Можно определить начальную позицию фонового изображения в виде горизонтальной и вертикальной координат посредством ключевых слов, точных абсолютных и процентных значений.

Значение по умолчанию **background-position: 0% 0%**. Не наследуется.

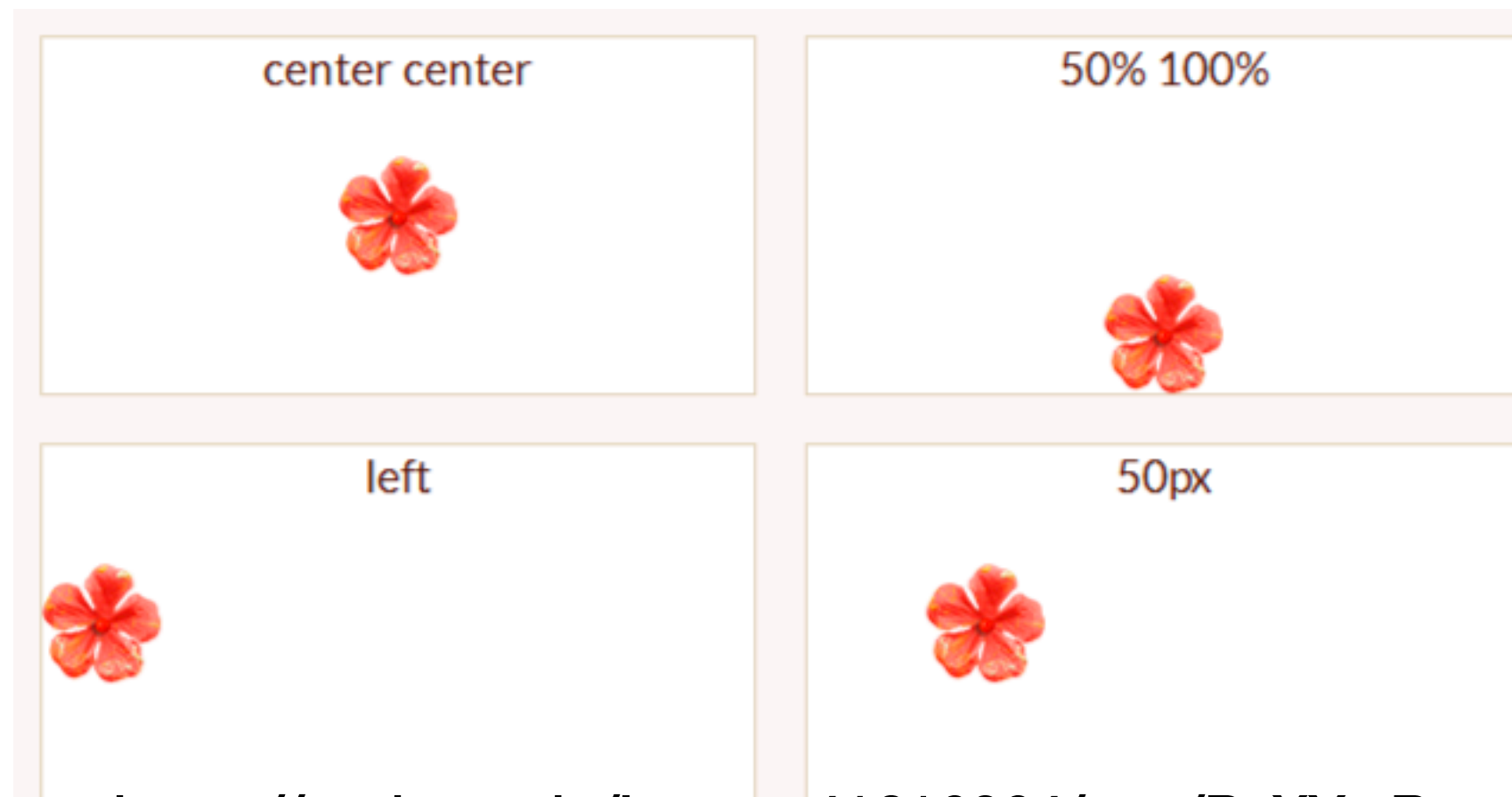
left top, left center, left bottom, right top, right center, right bottom, center top, center center, center bottom Позиционирование по горизонтали и вертикали задаётся с помощью пары ключевых слов. Если задано одно ключевое слово, второе примет значение center.

<https://codepen.io/jasongraphix/pen/lxhlm>

px / em / % Указывается два значения, первое определяет расстояние между левой стороной изображения и левым краем элемента-контейнера (по горизонтали), второе значение указывает расстояние между верхней стороной изображения и верхним краем элемента-контейнера (по вертикали). Также можно использовать отрицательные значения для обрезки части изображения.

Процентное значение рассчитывается относительно самого изображения и относительно элемента-контейнера, в котором оно расположено. Например, при задании `background-position:50% 50%`; центр изображения совпадет с центром элемента-контейнера.

Одновременно можно комбинировать значения в px, em и %.



<https://codepen.io/lostsou41216364/pen/PyXYwB>

background-attachment

Свойство позволяет фиксировать фоновое изображение при прокрутке страницы. Не наследуется.

scroll Фоновое изображение прокручивается вместе с текстом и другим содержимым. Значение по умолчанию.

fixed Предотвращает перемещение, фиксирует фоновое изображение на заднем плане.

local Фоновое изображение прокручивается вместе с содержимым элемента.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

background-clip

Свойство определяет, будет ли цвет фона ограничиваться содержимым элемента или будет простираться до внешнего края границы **border**. Не наследуется.

border-box Фон простирается до внешнего края границы элемента. Значение по умолчанию.

padding-box Фон простирается до внешнего края отступов элемента. **content-box** Фон окрашивает только содержимое элемента.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

<https://codepen.io/machal/pen/yamFI>

Фоновое изображение относительно родительского блока

background-origin

Свойство определяет, где будет позиционироваться фоновое изображение. Если одновременно задано свойство **background-attachment: fixed**, эффекта не будет.

padding-box Положение элемента вычисляется относительно верхнего левого угла с внутренней стороны границы элемента. Значение по умолчанию.

border-box Положение элемента вычисляется относительно верхнего левого угла с внешней стороны границы элемента.

content-box Положение элемента вычисляется относительно верхнего левого угла содержимого.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

background-size

Свойство позволяет масштабировать фоновое изображение по вертикали и горизонтали (**background-image**).

Оно описывает, как изображение будет растягиваться и обрезаться, чтобы полностью закрыть собой фоновую область.

С помощью этого свойства изображение также можно уменьшать по ширине и по высоте. Не наследуется.

auto Значение по умолчанию. Высота и ширина изображения равны его оригинальным размерам.

px / em / cm Размер задается парой значений, первое значение устанавливает ширину изображения, второе — высоту. Для того, чтобы фон масштабировался вместе с текстом, размеры изображения нужно задавать в em.

% Задаёт размер фонового изображения в процентах от ширины или высоты элемента, которое заполняется фоном.

cover Масштабирует изображение с сохранением пропорций так, чтобы его ширина или высота равнялась ширине или высоте блока.

contain Масштабирует изображение с сохранением пропорций таким образом, чтобы оно целиком поместилось внутри блока.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

<https://codepen.io/herihehe/full/aLwGt>

<https://codepen.io/Zonecss/pen/RyWZjG?page=1&>

Задание фона элемента одним свойством background

Свойство позволяет описать в одном объявлении следующие свойства фона: `background-color`, `background-image`, `background-position`, `background-size`, `background-repeat`, `background-origin`, `background-clip` и `background-attachment`.

Необязательно указывать все перечисленные свойства, если какое-либо свойство будет пропущено, оно примет значение по-умолчанию.

Если вы указываете в краткой записи фона свойство `background-size`, то его значения нужно будет записать через слеш /, чтобы отделить его от свойства `background-position`.

```
body {  
    background: #00ff00 url("smiley.gif") no-repeat fixed  
center;  
}
```

CSS3-текст

CSS3-текст включает свойства для работы с текстом, позволяющие решить проблемы с переносом текста и обрезкой текста в пределах содержимого.

<https://codepen.io/lostsou41216364/pen/JmwPWJ>

Обрезка строки text-overflow

Свойство позволяет ограничивать длину текста в случае, когда он не уместится в контейнер, визуально обрезаая его или отображая многоточием.

Свойство работает только при задании следующих условий: должна быть определена ширина контейнера, элемент должен иметь значения `overflow: hidden` и `white-space: nowrap`. Не наследуется.

<https://codepen.io/lostsou41216364/pen/mzabQv>

clip Значение по умолчанию. Текст обрезается в пределах области содержимого, при этом может отобразиться лишь часть символа.

<https://codepen.io/lostsou41216364/pen/LgMPvr>

ellipsis Замещает текст, не уместившийся в блок, с помощью многоточия.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

Перенос внутри слов word-break

В обычной ситуации слова переносятся на другую строку в местах пробелов или дефисов или же в случае принудительного переноса с использованием элемента `
`.

Данное свойство позволяет установить правила переноса внутри слов для того, чтобы длинные строки полностью заполняли пространство внутри контейнера.

normal Значение по умолчанию. Текст в контейнере отображается в привычном для нас виде в соответствии с правилами, установленными в языке.

break-all Пробелы в словах могут быть оформлены между любыми двумя буквами. <https://codepen.io/team/css-tricks/pen/EVeVyo>

keep-all Запрещает разрывы между парами букв слова.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

Перенос слов в строке word-wrap

Применяется в случаях, когда текст не уместается в свой контейнер и выходит за его границу.

Позволяет разрывать длинные слова и переносить их на следующую строку. Наследуется. Срабатывает лишь в случае, если значение **white-space** разрешает разрыв строк.

normal Значение по умолчанию. Слова разрываются только по правилам, принятым в языке.

break-word Разрыв слов и перенос на следующую строку происходит в месте границы контейнера.

initial Устанавливает значение свойства в значение по умолчанию.

inherit Наследует значение свойства от родительского элемента.

<https://codepen.io/oxxo/pen/nislq>