

CSS



Свойства ширины

Определяет ширину содержимого блока.

Ширина содержимого встроенных блоков определяется шириной отображаемого содержимого внутри них. Встроенные блоки сливаются в линейные блоки. Ширина линейных блоков определяется шириной содержащего блока, но может быть уменьшена из-за наличия свойства `float`.

Отрицательные значения не допускаются.

- `width: 100px;`
- `width: 50%;`
- `width: auto;`

Минимальная и максимальная ширина: свойства `min-width` и `max-width`

Свойства `min-width` и `max-width` позволяют ограничивать ширину содержимого до определенного диапазона. Значения не могут быть отрицательными. Для `min-width` значение по умолчанию 0, для `max-width` — none.

`min-width` - устанавливает минимальную ширину элемента. Если значение ширины (`width`) меньше значения `min-width` , то ширина элемента принимается равной `min-width` .

`max-width` - устанавливает максимальную ширину элемента. Если значение ширины (`width`) больше значения `max-width` , то ширина элемента принимается равной `max-width` .

Высота содержимого: свойство `height`

Свойство `height` определяет высоту содержимого блока. Значения длины не могут быть отрицательными.

`min-height` - задает минимальную высоту элемента.

`max-height` - задает максимальную высоту элемента.

В качестве значений принимаются пикселы (`px`), проценты (`%`) и другие единицы измерения, принятые в CSS. Отрицательные значения не допускаются. `inherit` наследует значение родителя.

Минимальная и максимальная ширина: свойства `min-width` и `max-width`

`float` - определяет, по какой стороне будет выравниваться элемент, при этом остальные элементы будут обтекать его с других сторон.

`float` : `left` | `right` | `none`

`left` - выравнивает элемент по левому краю, а все остальные элементы, вроде текста, обтекают его по правой стороне.

`right` - выравнивает элемент по правому краю, а все остальные элементы обтекают его по левой стороне.

`none` - обтекание элемента не задается.

Позиционирование

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**. Свойство не наследуется.

`position` : `static` | `relative` | `absolute` | `fixed`

`static` - значение по умолчанию, означает отсутствие позиционирования. Элементы отображаются последовательно один за другим в том порядке, в котором они определены в **HTML-документе**. Используется для очистки любого другого значения позиционирования.

Позиционирование

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**.

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

Позиционирование

`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 краев 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 счет свойства `z-index`). Любой абсолютно позиционированный элемент генерирует блок, то есть принимает значение `display: block`;

Позиционирование

`fixed` - фиксирует элемент в указанном месте страницы. Блоком-контейнером фиксированного элемента является окно просмотра, то есть элемент всегда фиксируется относительно окна браузера и не меняет своего положения во время прокрутки страницы. Сам элемент при этом полностью удаляется из основного потока документа и создаётся в новом потоке документа.

Смещение блока: свойства `top`, `right`, `bottom`, `left`

Свойства (`top`, `right`, `bottom`, `left`) 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. Не наследуются.

`top`: `auto` | `px` | `%`

`auto` - значение по умолчанию. Вычисляемое значение свойства равно нулю.

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

Смещение блока

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.



`background` - 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.

Свойство background-color

`background-color` - свойство устанавливает цветную заливку для любого **HTML-элемента**. Для блочных элементов цвет фона распространяется на всю ширину и высоту блока элемента, для строчных — только на область их содержимого.

`background-color` : `color` | `transparent`

`transparent` - значение по умолчанию. Задаёт прозрачный фон для элемента.

`color` - Значение принимает все форматы цвета свойства.

```
div { background-color : #353d4a; }
```

```
button { background-color : transparent ; }
```

`background-image` - свойство устанавливает изображение в качестве фона для элемента. Фоновым изображением может быть картинка или градиент, который задается с помощью функций: `linear-gradient()`, `radial-gradient()`.

`background-image : url(URL) | none`

`url(URL)` - абсолютный или относительный адрес изображения.

`none` - значение по умолчанию, также удаляет изображение у элемента.

```
div { background-image : url(https://html5book.ru/images/flower.png); }  
button { background-image : linear-gradient(to top, white, lightblue); }
```

Укладка изображений: свойство background-repeat

`background-repeat` - свойство определяет, каким образом будет повторяться фоновый рисунок.

`background-repeat` : `repeat` | `no-repeat` | `repeat-x` | `repeat-y`

`repeat` - весь фон страницы будет заполнен фоновым рисунком.

`no-repeat` - фоновое изображение не будет повторяться.

`repeat-x` - фоновый рисунок повторяется от левого до правого края по верхнему краю страницы.

`repeat-y` - фоновый рисунок повторяется от верхнего до нижнего края по левому краю страницы.

```
div { background-repeat : no-repeat ; }
```

Фиксация изображения: background-attachment

Свойство `background-attachment` позволяет указать, фиксировать изображение к указанному блоку.

`background-attachment` : `scroll` | `fixed`

`scroll` - фоновое изображение прокручивается вместе с текстом и другим содержимым. Значение по умолчанию.

`fixed` - предотвращает перемещение, фиксирует фоновое изображение на заднем плане.

```
div { background-attachment : fixed; }
```


Размер изображений

`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.

`background-size` : `auto` | `px` | `%` | `cover` | `contain`

`auto` - значение по умолчанию. Высота и ширина изображения равны его оригинальным размерам.

`px` - размер задается парой значений, первое значение устанавливает ширину изображения, второе — высоту.

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

Размер изображений

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.

```
div { background-size : 50px 150px; }
```

```
div { background-size : 50% 30%; }
```

```
div { background-size : cover ; }
```

Комбинированный вид

`background` - свойство позволяет описать в одном объявлении следующие свойства фона: `background-color`, `background-image`, `background-position`, `background-size`, `background-repeat` и `background-attachment`. Необязательно указывать все перечисленные свойства, если какое-либо свойство будет пропущено, оно примет значение по-умолчанию.

Если указать 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 / cover;  
}
```

Множественные фоны

С помощью **CSS3** вы можете применить несколько фонов к элементам. Они будут располагаться поверх друг друга: фон, заданный первым - в самом верху, последний фон - в самом низу.

```
body {  
    background: #00ff00 url(smile.gif) no-repeat, fixed center / cover,  
               url(smile2.gif) no-repeat, center ,  
               url(smile3.gif);  
}
```

```
body {  
    background-image: url(smile.gif), url(smile2.gif), url(smile3.gif);  
    background-repeat: no-repeat, repeat-x, no-repeat;  
    background-size: cover, 50% 30%, 150px 150px;  
}
```