

Курс: HTML/CSS + JS С НУЛЯ

Тренер:
Илья Литвинов

Лекция 2



Содержание

- 1.CSS - понятие
- 2.Структура языка
- 3.Селекторы
- 4.Псевдоклассы/псевдоэлементы
- 5.Единицы измерения
- 6.Блочные/строчные элементы/позиционирование
- 7.Текст
- 8.Цвет
- 9.Фоны
- 10.Рамки
- 11.Тени



CSS

CSS

CSS (Cascading Style Sheets), или каскадные таблицы стилей, описывают правила форматирования отдельного элемента веб-страницы. Создав стиль один раз, его можно применять к любым элементам страницы сколько угодно раз.

Стиль - это правило которое указывает браузеру что и каким образом форматировать, цвет, размер, положение на странице. Фактически определение стиля состоит из двух основных элементов: это сам элемент веб-страницы, который непосредственно подлежит форматированию браузером, — *селектор*, а также формирующие команды — *блок объявления*.

HTML используется для структурирование, CSS для изменения внешнего вида страницы

Селекторы

Определение стиля состоит из двух основных частей: самого элемента веб-страницы — селектора, и команды форматирования — блока объявления.

```
селектор {  
    свойство: значение;  
    свойство: значение;  
}
```

```
div {  
    width: 100px;  
    height: 100px;  
    background: red;  
}
```

Селекторы

*

.class

#id

tag

[attribute]

:pseudoclass

::pseudoelement

.class, #id

.class tag

tag.class

#id > tag

#id + .class

.class ~ tag

[attr="value"]

[attr~="value"]

[attr|="value"]

[attr^="value"]

[attr\$="value"]

[attr*="value"]

Селекторы

«Псевдоклассы определяют динамическое состояние элементов, которое изменяется с помощью действий пользователя, а также положение в дереве документа. Примером такого состояния служит текстовая ссылка, которая меняет свой цвет при наведении на неё курсора мыши. При использовании псевдоклассов браузер не перегружает текущий документ, поэтому с помощью псевдоклассов можно получить разные динамические эффекты на странице.»

Селектор:Псевдокласс { Описание правил стиля }

Псевдоклассы состояния

:active

Происходит при активации пользователем элемента. Например, ссылка становится активной, если навести на неё курсор и щёлкнуть мышкой. Несмотря на то, что активным может стать практически любой элемент веб-страницы, псевдокласс `:active` используется преимущественно для ссылок.

:link

Применяется к непосещенным ссылкам, т. е. таким ссылкам, на которые пользователь ещё не нажимал. Браузер некоторое время сохраняет историю посещений, поэтому ссылка может быть помечена как посещенная хотя бы потому, что по ней был зафиксирован переход ранее.

:focus

Применяется к элементу при получении им фокуса. Например, для текстового поля формы получение фокуса означает, что курсор установлен в поле, и с помощью клавиатуры можно вводить в него текст

Псевдоклассы состояния

:hover

Псевдокласс `:hover` активизируется, когда курсор мыши находится в пределах элемента, но щелчка по нему не происходит.

:visited

Данный псевдокласс применяется к посещённым ссылкам. Обычно такая ссылка меняет свой цвет по умолчанию на фиолетовый, но с помощью стилей цвет и другие параметры можно задать самостоятельно

:focus

Применяется к элементу при получении им фокуса. Например, для текстового поля формы получение фокуса означает, что курсор установлен в поле, и с помощью клавиатуры можно вводить в него текст

Псевдоклассы положения эл-та.

:first-child

Применяется к первому дочернему элементу селектора, который расположен в дереве элементов документа. Чтобы стало понятно, о чем речь, разберём небольшой код

:nth-child(odd | even | <число> | <выражение>)

Используется для добавления стиля к элементам на основе нумерации в дереве элементов.

Псевдоеlementы

- `:first-letter` - Стиль первой буквы текстового блока
- `:first-line` - Стиль первой строки текстового блока
- `:after` - Добавляет содержимое после элемента.
- `:before` - Добавляет содержимое до элемента.
- `::selection` - Стиль выделенного пользователем текста.

Единицы измерения

Абсолютные

- дюймы (in)
- сантиметры (cm)
- миллиметры (mm)
- пункты (pt)
- пики (pc)

В терминах спецификации css $1\text{pt} = 1/72\text{in}$, а $1\text{pc} = 12\text{pt}$.

В свойстве font-size задание отрицательного значения в единицах длины, например -25cm, **недопустимо**.

Почему абсолютные? Потому что за ними в физическом мире стоит реальная величина, то есть *эталоны*.

С такими единицами работает устройство вывода, которое имеет реальный физический размер, например: при печати будем использовать такие единицы.

А вот для мониторов эти единицы не имеют никакого значения, *есть некая условность*, но это действительно только лишь *условность*.

Относительные

- em (кегельная)
- x-height (ex)
- px (пикселы)
- rem(рем)

Проценты (%) — всегда величина, о которой стоит говорить отдельно. Она настолько важная, что W3C вынес ее в отдельную колонку. То есть % можно писать не у всех правил, а только у некоторых и когда мы пишем % мы должны понимать от чего они считаются.

Пиксель — это мельчайшая точка, которую можно установить на экране компьютера. Почему пиксель **относительная** величина? Мы берем *физический* монитор, у которого есть диагональ — это реальная физическая величина. Но ведь мы можем поставить на нем разрешение *любое*, правильно? В зависимости от поставленного нами разрешения, величина пикселя может меняться. Вот почему пиксель — **относительная** величина. То есть на разных мониторах px имеет разный размер.

Единица em ссылается на размер стандартного шрифта, установленного в глобальных параметрах браузера.

rem — размер относительно базового (определённого в .html{ }) размера шрифта

Блочные/строчные элементы

Блочные/строчные элементы

Блочные и строчные элементы определяют модель визуального форматирования, которая основывается на возможности изменения представления элемента браузером пользователя, используя значения свойства `display`.

Блочные элементы

Блочные и строчные элементы определяют модель визуального форматирования, Блочный элемент (`{display: block;}`) создает разрыв строки перед тегом и после него. Он образует прямоугольную область, по ширине занимающую всю ширину веб-страницы или блока-родителя, если для него не задано значение `width`.

Блочные элементы могут содержать внутри себя элементы любого типа. Нельзя размещать блочные элементы внутри строчных, за исключением элемента ``. Для блочных элементов можно задавать `margin` и `padding`.

Свойства `width` и `height` устанавливают ширину и высоту области содержимого элемента. Фактическая ширина элемента складывается из ширины полей (внутренних отступов), границ и внешних отступов. основывается на возможности изменения представления элемента браузером пользователя, используя значения свойства `display`.

Строчные элементы

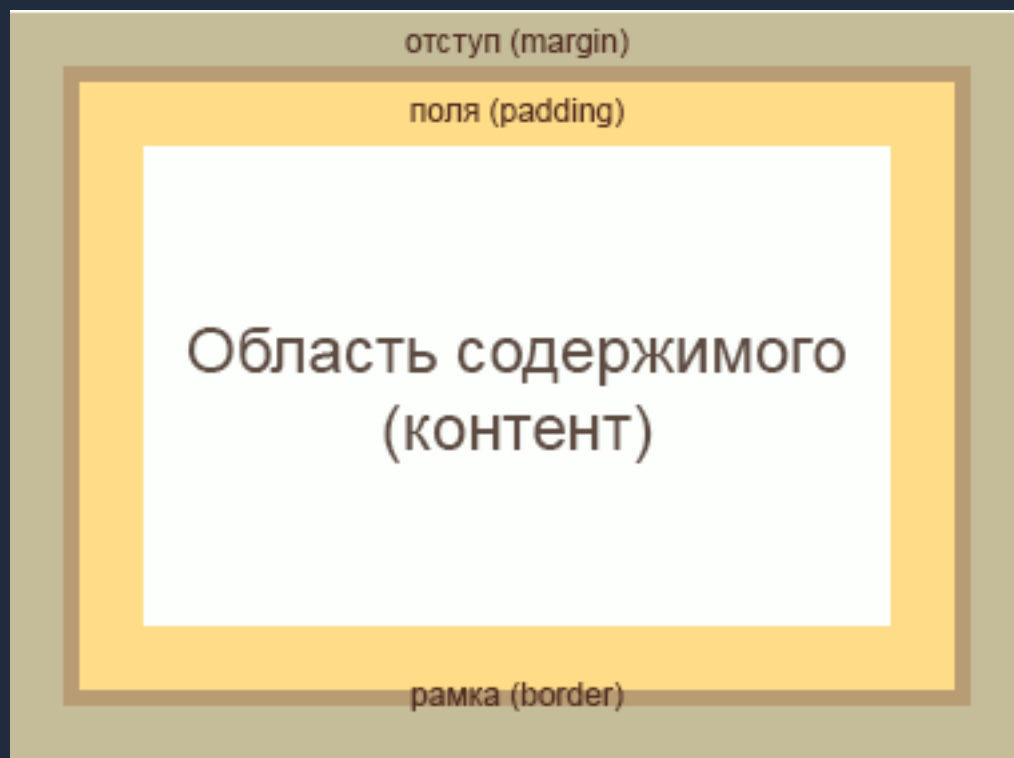
Встроенные (строчные) элементы (`{display: inline;}`) не создают блоки, они отображаются на одной строке с содержимым рядом стоящих тегов. Строчные элементы являются потомками блочных элементов. Они игнорируют верхние и нижние `margin` и `padding`, но если для элемента задан фон, он будет распространяться на верхний и нижний `padding`, заходя на соседние строки текста.

Ширина и высота строчного элемента зависит только от его содержания, задать размеры с помощью CSS нельзя. Можно увеличить расстояние между соседними элементами по горизонтали с помощью горизонтальных полей и отступов.

Для того, чтобы верхние и нижние поля и отступы работали для встроенного элемента, нужно использовать конструкцию `{display: inline-block;}`, элемент останется встроенным, но к нему можно будет полноценно применить поля, отступы, задать ширину и высоту.

Box-model

В блочной модели элемент рассматривается как прямоугольный контейнер, состоящий из содержимого, внутреннего отступа, рамки и внешнего отступа.



Box-model

Область содержимого — это содержимое элемента, например, текст или изображение. **Внутренний отступ**, или поле элемента, (**padding**) добавляет отступы внутри элемента, между его основным содержимым и его границей. Если для элемента задать фон, то он распространится также и на поля элемента. Внутренний отступ не может принимать отрицательных значений, в отличие от внешнего отступа.

Внешний отступ (margin) добавляет отступы за границами элемента, создавая тем самым промежутки между элементами. Они всегда остаются прозрачными и через них виден фон родительского элемента.

Значения **padding** и **margin** задаются в следующем порядке: верхнее, правое, нижнее и левое.

Граница или рамка элемента задается с помощью свойства **border**. Если цвет рамки не задан, она принимает цвет основного содержимого элемента, например, текста. Если рамка имеет разрывы, то сквозь них будет проступать фон элемента.

Внешние, внутренние отступы и рамка элемента не являются обязательными, по умолчанию их значение равно нулю. Тем не менее, некоторые браузеры добавляют этим свойствам положительные значения по умолчанию на основе своих таблиц стилей.

Позиционирование

Позиционирование

CSS позиционирование управляет местоположением элементов на странице. Изменять положение элементов на странице можно с помощью CSS-свойств **position** и **float**.

свойство position

Свойство position определяет порядок, в соответствии с которым выбранный элемент отображается на веб-странице. По умолчанию все элементы располагаются последовательно один за другим в том порядке, в котором они определены в html-документе ({position: static}).

Блочный элемент (p, div, h1 и др.) занимает 100% ширины родительского элемента (по умолчанию — body). Поэтому блочные элементы отображаются один под другим в соответствии с разметкой страницы.

Строчный элемент (em, strong, span и др.) занимает ширину, которая соответствует ширине содержимого внутри него. Поэтому строчные элементы отображаются рядом друг с другом.

Свойство position вместе со значениями top, right, bottom и left отображает элемент с нарушением обычного порядка, смещая его на заданное расстояние. При позиционировании элементов можно использовать как положительные, так и отрицательные значения.

свойство position

static значение по умолчанию. Элементы отображаются последовательно один за другим в том порядке, в котором они определены в html-документе.

relative позволяет сместить элемент относительно его текущего положения в структуре html-документа вверх, вправо, вниз или влево (зависит от того, какое задано значение: **top** / **right** / **bottom** / **left**). Остальные элементы размещены на странице относительно его первоначального положения, без смещения.

fixed позволяет зафиксировать элемент в нужном месте страницы. При прокрутке страницы положение элемента на странице не меняется. Элемент можно сместить вверх, вправо, вниз или влево (в зависимости от заданного значения: **top** / **right** / **bottom** / **left**).

свойство position

absolute можно задавать только для блочных (`display: block`) и строчно-блочных (`display: inline-block`) элементов.

Позиция элемента смещается относительно родительского элемента вверх, вправо, вниз или влево (зависит от того, какое задано значение: `top` / `right` / `bottom` / `left`), при этом нарушается порядок отображения элементов.

Элементы, следующие за ним, могут попасть под него.

На положение также влияет значение свойства `position` родительского элемента.

Так, если у родителя значение `position` установлено как `static` или родителя нет, то отсчет положения ведется от верхнего края окна браузера.

Если у родителя значение `position` задано как `relative` / `fixed` / `absolute`, то отсчет положения элемента ведется от края родительского элемента.

Можно вкладывать один блок с `position: absolute` в другой блок с `position: absolute`.

Текст

ТЕКСТ

CSS текст представляет набор css-стилей для форматирования текстового содержимого веб-страниц. Использование CSS-стилей для форматирования текста позволяет придать html-элементам желаемый вид, благодаря чему html-теги могут применяться только по своему прямому назначению — для определения структуры документа.

Горизонтальное выравнивания

Данное свойство определяет, как будут выравниваться строки текста элемента относительно границ блока. Применяется только к блочным элементам, например, абзацам. Наследуется.

text-align

Горизонтальное выравнивания

left Выравнивание по левому краю элемента.

Синтаксис: `p {text-align: left;}`

right Выравнивание по правому краю элемента.

Синтаксис: `p {text-align: right;}`

center Выравнивание по центру элемента, управляет выравниванием содержимого, а не самих элементов. Центрирует каждую строку текста элемента.

Синтаксис: `p {text-align: center;}`

justify Выравнивание по ширине элемента. В выровненном по ширине тексте оба конца строки прижимаются к левому и правому краям родительского элемента. Пробелы между словами и буквами распределяются таким образом, чтобы длина всех строк была равна. Различные браузеры могут увеличить как отступы между словами, так и интервалы между буквами.

Синтаксис: `p {text-align: justify;}`

Отступ

Данное свойство устанавливает отступ (выступ) в первой строке элемента, помогая создать иллюзию структурированного текста. В общем случае можно применять к любому блочному элементу, задавая как положительные, так и отрицательные значения.

text-indent

Смещает первую строку на заданную величину. Можно задавать как положительные, так и отрицательные значения.

Синтаксис: `p {text-indent: 5px;}`

Вертикальное выравнивание

высота строк

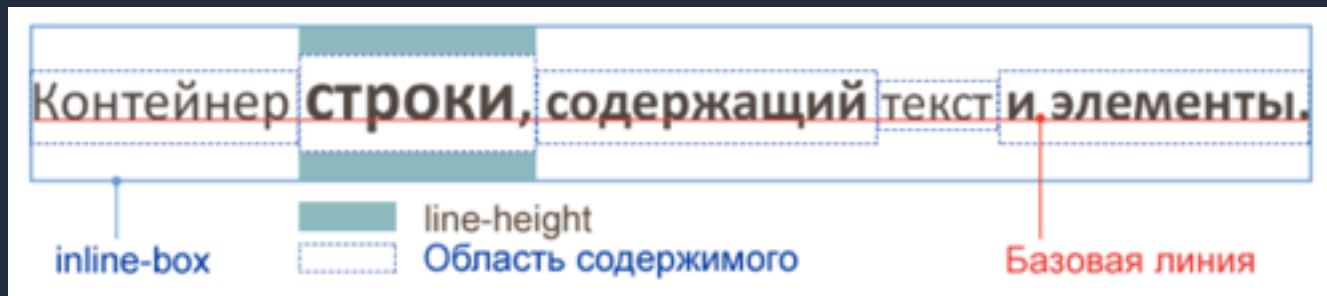
Свойство задаёт расстояние между базовыми линиями строк текста, определяя величину, на которую увеличивается или уменьшается высота блока каждого элемента. Управляет межстрочным интервалом — дополнительным расстоянием между строками текста над и под шрифтом. Наследует значения родительского элемента.

Свойство определяет минимальное, а не абсолютное значение, поэтому расстояние между базовыми линиями текста может превышать значение line-height.

Чтобы определить межстрочный интервал, нужно найти разность между line-height и **font-size**, разницу поделить на два, а каждую половину прибавить к области содержимого сверху и снизу. Принимает только положительные значения. Стандартный межстрочный интервал эквивалентен 120%.

Вертикальное выравнивание

высота строк



`h1 {line-height: 20px(%, em, число);}`

Вертикальное выравнивание

vertical-align

Применяется только к строчным элементам, к изображениям и полям форм. Не выравнивает содержимое блочного элемента. Не наследуется.

vertical-align

Вертикальное выравнивание

vertical-align

baseline Выравнивает базовую линию элемента по базовой линии его родителя, совмещая среднюю линию элемента со средней линией родительского элемента.

Синтаксис: `img {vertical-align: baseline;}`

sub Делает элемент подстрочным (аналогично с тегом `<sub>`). Величина понижения элемента может меняться в зависимости от браузера пользователя.

Синтаксис: `img {vertical-align: sub;}`

super Делает элемент надстрочным (аналогично с тегом `<sup>`). При этом значения `sup` и `super` не меняют размер шрифта, по умолчанию текст надстрочного и подстрочного элемента имеет такой же размер, как и текст родительского элемента.

Синтаксис: `img {vertical-align: super;}`

Вертикальное выравнивание

vertical-align

top Верхний край элемента совмещается с верхним краем самого высокого элемента в линии.

Синтаксис: `img {vertical-align: top;}`

text-top Верхний край элемента совмещается с верхним краем шрифта родительского элемента.

Синтаксис: `img {vertical-align: text-top;}`

middle Средняя линия элемента (обычно изображения) совмещается с линией, проходящей через середину родительского элемента.

Синтаксис: `img {vertical-align: middle;}`

Вертикальное выравнивание

vertical-align

bottom Нижний край элемента совмещается с нижним краем самого низкого элемента в линии.

Синтаксис: `img {vertical-align: bottom;}`

text-bottom Нижний край элемента совмещается с нижним краем шрифта родительского элемента.

Синтаксис: `img {vertical-align: text-bottom;}`

длина (px, em) Позволяет переместить элемент на заданное расстояние.


Синтаксис: `img {vertical-align: 5px;}`

% Не позволяет устанавливать middle, вычисляется как часть line-height элемента, а не его родителя, т.е. если установить значение vertical-align, равное 50% для элемента с line-height равным 20px, то базовая линия элемента поднимется на 10px.

Синтаксис: `img {vertical-align: 100%;}`

Вертикальное выравнивание


vertical-align

Изображение  `{vertical-align: baseline}` не имеет базовой линии, поэтому оно выравнивается по базовой линии родителя (в данном случае, текста).

Этот абзац содержит **надстрочный** `{vertical-align: super}` и **подстрочный** `{vertical-align: sub}` текст, который имеет такой же размер, как и основной текст.

Изображение  `{vertical-align: bottom}` выравнивается по низу контейнера строки, смещаясь ниже базовой линии текста.

Изображение  `{vertical-align: text-bottom}` выравнивается относительно нижней линии текста строки.

Изображение  `{vertical-align: top}` выравнивается по верху контейнера строки, при этом учитывается размер шрифта родительского элемента.

Шрифты

Шрифты

CSS шрифты управляют внешним видом шрифта текста веб-страниц. Используя различные шрифты для заголовков, абзацев и других элементов, можно задавать определенный стиль письменных сообщений, передавая желаемые эмоции и настроение, поэтому к выбору шрифтов для своих страниц нужно подходить очень взвешенно, продуманно применяя к ним стили.

font-family

Данное свойство используется для выбора начертания шрифта

family-name Название (имя) семейства шрифтов. Рекомендуется указывать вместе с базовым семейством.

```
h1 {font-family: Times, 'New Century Schoolbook', serif;}
```

generic-family Базовое семейство. CSS определяет пять базовых семейств шрифтов:

- Шрифты с засечками — Serif (Times New Roman, Times, Garamond, Georgia)
- Рубленые шрифты — Sans-serif (Helvetica, Geneva, Arial, Verdana, Trebuchet, Univers)
- Моноширинные шрифты — Monospace (Courier, Courier New, Andale Mono)
- Рукописные шрифты — Cursive (Comic Sans, Gabriola, Monotype Corsiva, Author, Zapf Chancery)
- Аллегорические шрифты (Western, Woodblock, Klingon)

```
h1 {font-family: sans-serif;}
```


font-style

Данное свойство позволяет выбрать стиль начертания для текста. При этом разница между курсивом и наклонным текстом заключается в том, что курсивное начертание вносит небольшие изменения в структуру каждой буквы, а наклонный текст представляет собой наклонную версию прямого текста. Наследуется

normal Устанавливает для текста обычный шрифт.

Синтаксис: `h1 {font-style: normal;}`

italic Выделяет текст курсивом.

Синтаксис: `h1 {font-style: italic;}`

oblique Устанавливает наклонное начертание шрифта.

Синтаксис: `h1 {font-style: oblique;}`

font-style

Normal font style

Italic font style

Oblique font style

font-style

Normal font style

Italic font style

Oblique font style

font-variant

Свойство позволяет выбирать варианты ввода шрифта — обычными или малыми прописными буквами. Наследуется.

normal Значение по умолчанию, которое описывает обычный текст.

Синтаксис: `h1 {font-variant: normal;}`

small-caps Все строчные буквы заменяются на малые прописные, которые отличаются от обычных прописных слегка измененными пропорциями и уменьшенным размером.

Очень похоже на `text-transform: uppercase`, отличие состоит в том, что здесь прописные буквы имеют разные размеры.

Синтаксис: `h1 {font-variant: small-caps;}`

font-weight

Свойство определяет насыщенность шрифта.

normal Устанавливает обычный шрифт. Эквивалентно значению насыщенности, равной 400.

Синтаксис: `h1 {font-weight: normal;}`

bold Делает шрифт текста полужирным. Эквивалентно значению насыщенности, равной 700.

Синтаксис: `h1 {font-weight: bold;}`

bolder Насыщенность шрифта будет больше, чем у предка.

Синтаксис: `span {font-weight: bolder;}`

lighter Насыщенность шрифта будет меньше, чем у предка.

Синтаксис: `span {font-weight: lighter;}`

font-size

absolute-size xx-small, x-small, small, medium, large, x-large, xx-large. Абсолютные размеры определены относительно друг друга и коэффициент масштабирования между двумя соседними абсолютными размерами составляет примерно 1,5 при переходе от меньшего к большему и 0,66 при переходе от большего к меньшему. В качестве стандартного размера принимается medium.

Синтаксис: h3 {font-size: small;}

relative-size smaller, larger. Относительные размеры обуславливают изменение размера шрифта элемента относительно родителя. При этом размер шрифта может выйти за рамки размеров, предполагаемых для xx-small и xx-large.

синтаксис: h1 {font-size: xx-large;}, em {font-size: large;}

px, pt Размер шрифта устанавливается с помощью положительного числового значения, как целого, так и дробного, например, p {font-size: 20px;}.

% Вычисляется на основании любого размера, унаследованного от родительского элемента. Обеспечивает более точную настройку вычисляемого размера шрифта. Задание размеров шрифта с помощью em эквивалентно процентному значению.

Синтаксис: h3 {font-size: 120%;}

color

HEX В данной системе используются арабские десятичные цифры от 0 до 9 и латинские буквы от A до F. Для веб-дизайна взяты 16 основных цветов, так называемый шестнадцатеричный код цвета #RRGGBB, где каждая пара отвечает за свою долю цвета: RR - красный, GG - зеленый и BB - синий. Каждая доля цвета находится в диапазоне от 00 до FF, например, `h2 {color:#ff0000;}`. Значение поддерживается всеми браузерами.

RGB RedGreenBlue, обозначает количество соответствующего тона (красный,зеленый,синий) в получаемом цвете. Запись цвета имеет следующий вид: `strong {color: rgb(0,0,255);}` или `p {color: rgb(0%,0%,100%);}`. Значение поддерживается всеми браузерами.

RGBA Система цветопередачи RGB, расширенная параметром Alpha, который используется для управления смешиванием цветов. Значение поддерживается IE9+, Firefox 3+, Chrome, Safari, and in Opera 10+, запись цвета имеет следующий вид: `span {color: rgba(255,0,0,0.3);}`, последнее число определяет степень прозрачности, задается значением от 0 до 1, где 0 соответствует полной прозрачности, а 1 — непрозрачности.

Фоны

Фон

CSS фон позволяет с помощью css-свойства background добавлять фон для любого html-элемента. Каждая веб-страница имеет фон, на котором можно размещать рисунки и текст. В качестве фона любого элемента можно задавать цвет, градиент или изображение.

Цвет фона

Свойство `background-color` позволяет задать цвет фона для любого html-элемента. Для блочных элементов цвет фона распространится на всю ширину и высоту блока элемента, для строчных - только на область их содержимого. Не наследуется.

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

transparent Значение по умолчанию. Задаёт прозрачный фон для элемента.

Фоновое изображение

Свойство `background-image` позволяет установить фоновое изображение для элемента. Фоновым изображением может быть как картинка, так и градиент, который задается с помощью функций `background-image: linear-gradient()`, `background-image: radial-gradient()`. Не наследуется.

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

`none` Удаляет изображение для элемента.

```
1  div {
2      background-image: url(http://html5book.ru/images/flower.png);
3  }
4  table {
5      background-image: radial-gradient(farthest-corner at 50% 50%, white, #DCECF8);
6  }
7  p {
8      background-image: linear-gradient(to top left, white, lightblue);
9  }
```

Повтор фоновых изображений

Используя свойство `background-repeat`, можно определить, каким образом будет повторяться фоновый рисунок. По умолчанию рисунок всегда находится в верхнем левом углу. 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.

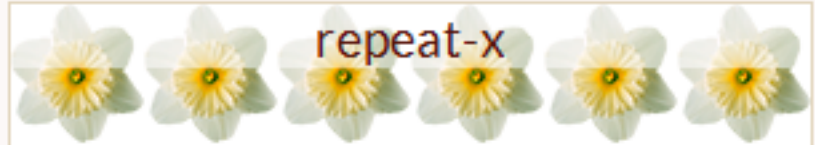
repeat-x Фоновый рисунок повторяется от левого до правого края веб-страницы по верхнему краю страницы.

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

Повтор фоновых изображений



no-repeat



repeat-x



repeat-y



repeat

Позиционирование фоновых изображений

Свойство `background-position` позволяет управлять точным расположением фонового изображения. Можно определить начальную позицию фонового изображения в виде горизонтальной и вертикальной координат посредством ключевых слов, точных абсолютных и процентных значений. Не наследуется.

left, center, right Горизонтальное позиционирование

top, center, bottom Вертикальное позиционирование

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.

Можно одновременно комбинировать значения в `px`, `em` и `%`.

Позиционирование фоновых изображений

center center



50% 100%



left



50px



Фиксация изображения на месте

Свойство `background-attachment` позволяет управлять положением фонового изображения 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.

```
1  div {  
2    background-attachment: scroll;  
3  }  
4  div {  
5    background-attachment: fixed;  
6  }  
7  div {  
8    background-attachment: local;  
9  }
```


Заполнение фоном отступов и границ

Свойство `background-clip` определяет, будет ли цвет фона ограничиваться началом границы элемента или будет продолжаться до внешнего края границы. Не наследуется.

border-box Фон простирается до внешнего края границы элемента.

padding-box Фона простирается до внешнего края отступов элемента.

content-box Фон окрашивает только содержимое элемента.

```
1  div {  
2      background-clip: border-box;  
3  }  
4  div {  
5      background-clip: padding-box;  
6  }  
7  div {  
8      background-clip: content-box;  
9  }
```

Заполнение фоном отступов и границ



Размер изображения

Свойство `background-size` 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.

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`.

% Задает 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.

Размер изображения

```
1  div {  
2    background-size: 300px 150px;  
3  }  
4  div {  
5    background-size: 50% 30%;  
6  }  
7  div {  
8    background-size: cover;  
9  }  
10 div {  
11   background-size: contain;  
12 }
```

Рамки

Рамки

CSS3 рамка предоставляет расширенный набор возможностей форматирования границ элементов с помощью свойств, позволяющих закруглить углы элемента, а также использовать изображения для оформления границ элемента.

`border: [border-width || border-style || border-color] | inherit`

```
1  .brd {  
2      border: 4px double black; /* Параметры границы */  
3      background: #fc3; /* Цвет фона */  
4      padding: 10px; /* Поля вокруг текста */  
5  }
```

Закругление границ углов

Свойство **border-radius** позволяет определить, каким образом закруглить границы углов. Кривая для каждого угла определяется с помощью одного или двух радиусов, определяющих его форму: круга или эллипса.

Закруглить углы можно как для всех углов элемента одновременно (с помощью **border-radius**), так и по отдельности, с помощью конструкций **border-top-left-radius**, **border-top-right-radius**, **border-bottom-right-radius**, **border-bottom-left-radius**.

длина Позволяет закруглить углы блока с помощью значений единиц длины — (px, em).

% Значения, позволяющие закруглить углы, задаются в процентах от длины и ширины сторон блока.

Тени

Тени

CSS3 тень бывает двух видов: текстовая тень (text-shadow) и блочная тень (box-shadow).

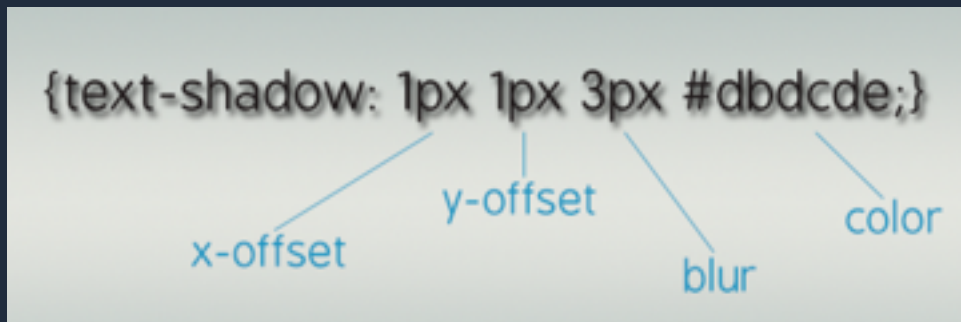
Размытые или объемные, цветные или многослойные тени создают удивительные эффекты для элементов.

Задавая тень для элемента, можно указывать только одно значение длины и цвет, таким образом создавая цветную копию элемента.

Text-shadow

Используя свойство text-shadow, вы сможете улучшить внешний вид текста или отдельных его элементов, придав необычный дизайнерский стиль.

Для создания каждой тени задаются цвет и три длины (смещение по оси X, смещение по оси Y, радиус размытия). Если значения длин отрицательные, то тень смещается влево и вверх от текста.



Text-shadow

```
2 h3 {  
3 text-shadow: 1px 1px 1px #776a64, -1px 1px 1px #776a64;  
4 color: #faf2e5;  
5 }
```

Box-shadow

Свойство box-shadow позволяет создать прямоугольную тень позади любого блока элемента. Также, тени могут следовать контурам блоков со скругленными углами. Не наследуется.

```
{box-shadow: 1px 1px 3px 2px #dbdcde;}
```

x-offset

y-offset

blur

spread

color

Box-shadow

Свойство box-shadow позволяет создать прямоугольную тень позади любого блока элемента. Также, тени могут следовать контурам блоков со скругленными углами. Не наследуется.

```
{box-shadow: 1px 1px 3px 2px #dbdcde;}
```

x-offset

y-offset

blur

spread

color

Ссылки

<http://html5book.ru/>

<http://csstemplater.com/>