

Значения

В спецификации CSS определяются самые разные типы значений, в большинстве из которых употребляются единицы измерения. Сочетая основные типы значений (например, числовых) с единицами измерения (например, пикселями), можно добиться практически любых результатов привлекательного оформления документов средствами CSS.

Ключевые слова

Ключевые слова определяются на основе свойств и имеют назначение, характерное только для данного свойства. Например, ключевое слово **normal** имеет совершенно особое назначение в свойствах `font-variant` и `letter-spacing`. Как и в именах свойств, в ключевых словах регистр букв не учитывается. Особое место принадлежит ключевому слову **inherit**, которое допускается во всех свойствах и всегда имеет одно и то же назначение: получить связанное со свойством значение из родительского элемента. Имеется еще одно ключевое слово **initial**, которое специально предназначено для представления исходного или устанавливаемого по умолчанию значения данного свойства. Таким образом, в результате объявления `font-family: initial` возвращается гарнитура шрифта (Times для большинства пользователей), устанавливаемая среди глобальных параметров браузера по умолчанию. Состояние и применение ключевого слова **initial** не были ясны на момент написания этой книги, и поэтому могут быть ненадежны.

Значения цвета

Ниже перечислены формы обозначения значений цвета.

#RRGGBB

Это обозначение в виде пар шестнадцатеричных значений хорошо известно тем, кто пользуется традиционной HTML-разметкой в авторской разработке веб-страниц. В данной форме записи первая пара цифр соответствует интенсивности красного цвета, вторая пара — интенсивности зеленого цвета, а третья — интенсивности синего цвета. В шестнадцатеричном обозначении каждая пара цифр указывается в пределах **00-FF**. Таким образом, чистый синий цвет обозначается как **#0000FF**, чистый красный цвет — как **#FF0000** и т.д.

#RGB

Это более краткая форма описанного выше обозначения цвета шестью шестнадцатеричными цифрами. В этой форме каждая цифра дублируется для получения в конечном итоге эквивалентного значения из шести цифр. Таким образом, шестнадцатеричное значение цвета **#F8C** становится значением цвета **#FF88CC**.

rgb (rrr , ggg , bbb)

В этой форме значения цвета допускается указывать в пределах **0-255**, где разрешаются только целые числа. И совсем не случайно, что эти пределы являются десятичным эквивалентом пределов **00-FF** в шестнадцатеричной форме. Так, чистый зеленый цвет обозначается в данной форме как **rgb (0 , 255 , 0)**, а белый цвет — как **rgb (255 , 255 , 255)**.

rgb (rrr . rr% , ggg . gg% , bbb . bb%)

В этой форме значения цвета RGB допускается указывать в процентах от **0%** до **100%**, где разрешаются десятичные числа (например, **75.5%**). Так, значение черного цвета обозначается как **rgb (0% , 0% , 0%)**, тогда как значение чистого синего цвета — как **rgb (0% , 0% , 100%)**.

hsl (hhh . hh , sss . ss% , lll . ll%)

В этой форме допускается обозначать цвет по его оттенку (углу цветового тона на цветовом круге), насыщенности и яркости (т.е. по цветовой модели HSL — “оттенок-насыщен-

ность–яркость”). Оттенок всегда указывается безразмерным числом в пределах от 0 до 360, а насыщенность и яркость — в процентах. Оттенки 0 и 360 равнозначны и оба относятся к красному цвету. Допускается указывать оттенки свыше 360, но они все равно приводятся к пределам 0–360. Таким образом, обозначение оттенка 454 равнозначно установке оттенка 94. Любое значение цвета HSL, независимо от его оттенка, будет воспроизведено как оттенок серого, если значение насыщенности равно 0%. Конкретный оттенок серого зависит от значения яркости. А любое значение цвета HSL, независимо от его оттенка, будет воспроизведено сплошным черным цветом, если значение яркости равно 0%, или сплошным белым цветом, если значение яркости равно 100%. Обычное для большинства цветов значение яркости составляет 50%.

rgba (rrr, ggg, bbb, a. aa)

rgba (rrr. rr%, ggg. gg%, bbb. bb%, a. aa)

hsla (hhh. hh, sss. ss%, lll. ll%, a. aa)

Эта форма служит расширением предыдущих форм, включая в себя значение альфа-канала (т.е. непрозрачности цвета). Значение непрозрачности должно быть указано вещественным числом в пределах от 0 до 1 включительно, но только не в процентах. Таким образом, обозначения **rgba (0, 0, 255, 0.5)**, **rgba (0, 0, 100%, 0.5)** и **hsla (0, 100%, 50%, 0.5)** равнозначны и определяют один и тот же полупрозрачный красный цвет. А шестнадцатеричное обозначение для цвета RGBA не предусмотрено.

<ключевое слово>

На основании исходных цветов по стандарту VGA в Windows для распознавания названий отдельных цветов определены 17 следующих ключевых слов: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, orange, purple, red, silver, teal, white и yellow. Как правило, цвета распознаются в браузерах и по другим ключевым словам, например цвета 147 X11, описанные в разделе 4.3 спецификации цвето-

вого модуля W3C CSS3 Color Module (по адресу <http://www.w3.org/TR/css3-color>). В этом цветовом модуле из списка основных распознаваемых ключевых слов исключено слово `orange`. Тем не менее оно присутствует в списке цветов X11 и поддерживается во всех известных браузерах по исторически сложившимся причинам.

currentColor

Этим специальным ключевым словом обозначается текущее вычисляемое значение свойства `color` отдельного элемента. Это означает, что с помощью объявления `background-color: currentColor` можно задать такой же цвет заднего плана, как и цвет переднего плана, что, в общем, не рекомендуется делать. Применительно к свойству `color` это равнозначно объявлению `border: 1px solid currentColor`. Этим ключевым словом удобно пользоваться для установки и отмены цвета границы, составляя документы по модели DOM с помощью сценариев.

transparent

Это специальное ключевое слово служит сокращенной формой для обозначения прозрачного цвета `rgba(0, 0, 0, 0)`, которое автоматически вычисляется всякий раз, когда употребляется ключевое слово `transparent`.

Числовые значения

Числовое значение обозначается как положительное или отрицательное, если это разрешается. Числа могут быть вещественными и представленными типом данных `<number>` или целыми и представленными типом данных `<integer>`. Они могут также иметь ограниченный диапазон допустимых значений, как, например, значения цвета в пределах **0–255**. Чаще всего диапазон чисел ограничивается только неотрицательными числами, но иногда такие числа могут быть представлены как `<non-negative number>` или `<non-negative integer>`, т.е. как неотрицательное вещественное или целое число.

Значения в процентах

Значение в процентах обозначается как число типа `<number>` со знаком процента (%). Между числом и знаком процента не должно быть никаких пробелов. Значение в процентах всегда вычисляется относительно какого-то другого значения. Например, в объявлении `font-size: 120%`; для элемента задается шрифт размером **120%** от вычисляемого значения свойства `font-size` в его родительском элементе. В некоторых свойствах может допускаться употребление только неотрицательных значений.

Значения длины

Значение длины обозначается как положительное или отрицательное число, если это разрешено, после которого сразу же следует двухбуквенное сокращение, обозначающее используемые единицы измерения длины. Между числом и обозначением единицы измерения не должно быть никаких пробелов. При обозначении нулевого (0) значения длины указывать единицы измерения не нужно. Единицы измерения длины разделяются на следующие два вида: *абсолютные единицы измерения*, которые теоретически измеряются одинаково, и *относительные единицы измерения*, которые измеряются относительно чего-то другого.

Абсолютные единицы измерения длины

Ниже перечислены абсолютные единицы измерения длины, употребляемые в CSS.

Дюймы (in)

Это типичные единицы измерения длины, указываемые на измерительных линейках в США. Преобразование дюймов в единицы, применяемые для измерения длины на экранах мониторов и прочих устройств отображения, обычно выполняется в

лучшем случае приблизительно, поскольку отображение информации на рабочих участках экранов во многих системах никак не связано с такими “настоящими” единицами измерения, как дюймы. Поэтому пользоваться дюймами следует крайне осторожно, разрабатывая документы, предназначенные для вывода на экран.

Сантиметры (cm)

Сантиметры обычно указываются на измерительных линейках в большинстве стран мира. В каждом дюйме насчитывается **2,54** сантиметра, а **1** сантиметр равен **0,394** дюйма. В отношении применения сантиметров в качестве единиц измерения длины отдельных элементов документов на экранах мониторов действуют те же самые предостережения, что и для дюймов.

Миллиметры (mm)

В одном сантиметре насчитывается **10** миллиметров, а в дюйме — **25,4** миллиметра, и **1** миллиметр равен **0,0394** дюйма. Но в отношении миллиметров следует иметь в виду упомянутые выше предупреждения по поводу измерения длины отдельных элементов документов на экранах мониторов.

Пункты (pt)

Пункты являются стандартными в полиграфии единицами измерения длины на печатных и наборных машинах и в программах обработки и верстки документов. В соответствии с современным определением в одном дюйме насчитывается **72** пункта. Следовательно, заглавные буквы текста, набранные кеглем (т.е. размером шрифта) **12** пунктов, должны быть высотой в одну шестую дюйма. Например, объявление `p {font-size: 18pt;}` равнозначно объявлению `p {font-size: 0.25in;}` при надлежащем преобразовании единиц измерения длины в среде отображения информации, как пояснялось выше.

Цицero (pc)

Это еще одна единица измерения, применяемая в полиграфии. Один цицero равнозначен **12** пунктам, т.е. в одном дюйме насчитывается **6** цицero. Заглавные буквы текста, набранные

кеглем в 1 цецеро, должны быть высотой в одну шестую дюйма. Например, в объявлении `p {font-size: 1.5pc;}` задается такой же размер шрифта для набора текста, как и в объявлениях, приведенных выше при пояснении пунктов. Применяя цецеро, следует также принимать во внимание упомянутые выше предостережения.

Относительные единицы измерения длины

Ниже перечислены относительные единицы измерения длины, используемые в CSS.

Высота шрифта (em)

Эта единица измерения обозначает высоту кегельной площадки заданного шрифта. Высота шрифта в CSS равнозначна знакоместу для заданного шрифта, а, по существу, — вычисляемому значению свойства `font-size`. В единицах измерения высоты шрифта можно указывать относительные размеры шрифтов. Например, **1.2em** означает то же самое, что и **120%** при указании шрифта.

Высота шрифта корневого элемента (rem)

Эта единица измерения обозначает высоту шрифта корневого элемента (т.е. элемента разметки `html` в HTML- и XHTML-документах). А в остальном она подобна единице измерения **em**.

X-height (ex)

Эта единица измерения обозначает x-высоту (т.е. размер строчных букв) шрифта. Но в определение подавляющего большинства шрифтов x-высота не входит, и поэтому во многих браузерах применяется следующее весьма приблизительное соотношение единиц измерения высоты шрифта: **1ex** равно **0.5em**. Исключением из этого правила является версия 5 браузера Internet Explorer для компьютеров Macintosh, в которой предпринимается попытка определить фактическую x-высо-

ту шрифта путем внутреннего растривания очень большой буквы **x** и подсчета пикселей!

Ширина нуля (ch)

Эта единица измерения обозначает ширину одного нуля (**Unicode +0300**) в текущей гарнитуре шрифта и в его размере.

Пиксели (px)

Пиксель представляет собой, как правило, прямоугольный и реже квадратный элемент изображения на экране, но в CSS пиксели определяются более абстрактно. В точки зрения CSS размер пикселя определяется таким образом, чтобы на один дюйм приходилось **96** пикселей. Многие пользовательские агенты пренебрегают этим определением в пользу простого обращения к пикселям на экране монитора. При увеличении, уменьшении изображения или печати страниц в силу вступают масштабные коэффициенты, где элемент шириной **100px** может быть воспроизведен на устройстве вывода больше 100 точек по ширине.

Единица измерения ширины области просмотра (vw)

Эта единица измерения вычисляется относительно ширины области просмотра, которая делится на **100**. Так, если ширина области просмотра составляет **937** пикселей, то **1vw** равно **9.37px**. Если же ширина области просмотра изменяется, например, в результате того, что окно браузера делается шире или уже, то вместе с ней изменяется и величина **vw**.

Единица измерения высоты области просмотра (vh)

Эта единица измерения вычисляется относительно высоты области просмотра, которая делится на **100**. Так, если ширина области просмотра составляет **650** пикселей, то **1vh** равно **6.5px**. Если же высота области просмотра изменяется, например, в результате того, что окно браузера делается длиннее или короче, то вместе с ней изменяется и величина **vh**.

Минимальная единица измерения области просмотра (vmin)

Эта единица измерения составляет **1/100** часть ширины или высоты области просмотра в зависимости от того, какая из этих

величин *меньше*. Так, если имеется область просмотра шириной **937** пикселей и высотой **650** пикселей, то **1vm** равно **6.5px**.

URI

Значение URI типа `<uri>` является ссылкой на файл, содержащий, например, графическое изображение или другую таблицу стилей. URI определяется в CSS относительно таблицы стилей, которая его содержит. Сокращение “URI” означает “Uniform Resource Identifier” (Универсальный идентификатор ресурса) и в последнее время чаще употребляется вместо URL. (Формально URL являются подмножеством URI.) Тем не менее в спецификации CSS, которая была впервые определена, когда URI еще обозначались как URL, это означает, что ссылки на URI будут фактически появляться в форме `url(<uri>)`. Забавно!

Углы

Углы выражаются в форме `<angle>`, где после числа типа `<number>` сразу же следует единица измерения угла. Существуют следующие разновидности единиц измерения углов: градусы (`deg`), грады (`grad`), радианы (`rad`) и обороты (`turn`). Например, прямой угол можно объявить одним из следующих способов: **90deg**, **100grad**, **1,571rad** или **0,25turn**, и в каждом случае указанные значения прямого угла преобразуются в градусы в пределах от **0** до **360**. То же самое относится и к отрицательным значениям углов, которые вполне допустимы. Так, величины **-90deg** и **270deg** обозначают один и тот же прямой угол.

Время

Величина времени типа `<time>` выражается в виде неотрицательного числа типа `<number>`, после которого сразу же следует

единица измерения времени. Существуют следующие разновидности единиц измерения времени: секунды (s) и миллисекунды (ms). Величины времени употребляются в акустических стилях, которые не нашли широкого признания, а также в намного лучше поддерживаемых монтажных переходах и в анимации.

Частоты

Величина частоты типа `<frequency>` выражается в виде неотрицательного числа типа `<number>`, за которым сразу же следует единица измерения частоты. Существуют следующие разновидности единиц измерения частоты: герцы (Hz) и килогерцы (kHz). Они обозначаются без учета регистра букв, и поэтому величины **6kHz** и **6khz** равнозначны. На момент написания данной книги величины частоты использовались в акустических стилях, которые не нашли широкой поддержки.

Символьные строки

Символьная строка типа `<string>` состоит из последовательности символов, заключаемых в одинарные или двойные кавычки. Если в состав символьной строки входит та же самая кавычка, в которую она заключается, то такая кавычка должна быть экранирована, как, например, в следующих строках: `'That \'s amazing!'` или `"Deploy the \"scare quotes\" at once!"`. Если же в символьной строке необходимо указать знак новой строки, его следует обозначить как `\A`, что в Уникоде означает кодовую точку для знака новой строки. Любой символ в Уникоде может быть представлен с помощью экранированной ссылки на кодовую точку. Так, левая закругленная двойная кавычка (`“`) может быть представлена в Уникоде как `\201C`. Если символьная строка все же содержит знак перевода строки из соображений удобочитаемости, этот знак должен быть экранирован, а при обработке строки — удален.