

CSS - единицы измерения

Душная, но важная теория

Вес селекторов в CSS

Вес селектора — это условные четыре позиции x, x, x, x

Инлайн стили - 1000

Идентификаторы - 0100

Классы, атрибуты и псевдоклассы - 0010

Теги и псевдоэлементы - 0001

Чем больше уточнение тем лучше (но лучше когда сумма всех единиц в числе не превышает 5)

!important

Перебивает вес селектора

Если несколько !important то иерархия повторяется с обычном весом но уже в рамках сравнения свойств с указанием important

!important - является плохой практикой так как тяжело контролировать, чаще всего используется в легаси коде в котором нет времени и ресурсов разбираться, в случае использования библиотеки в нестандартном кейсе (не предусмотренной библиотекой), а также для быстрого (hot-fix) решения с дальнейшим исправлением.

Работа с цветами в CSS

свойства `color`, `background-color`, `border-color`, `shadow`-свойства, `fill` у `svg` и тд;

`initial` - дефолтный цвет (у текста черный у фона белый);

`inherit` - наследует значение от родителя;

`transparent` - прозрачный;

`currentColor` - ключевое слово - наследуемость с `color` (цепляется к ближайшему явно указанному `color` значению), похож на `inherit`;

Чаще всего используется HEX.

RGB и RGBA

RGB (Red, Green, Blue): Этот формат представляет цвета как комбинацию значений красного, зеленого и синего цветов. Каждый цвет представлен числом в диапазоне от 0 до 255.

- `rgb(255, 0, 0)`: Красный цвет.
- `rgb(0, 255, 0)`: Зеленый цвет.
- `rgb(0, 0, 255)`: Синий цвет.

RGBA (Red, Green, Blue, Alpha): Аналогично RGB, но с дополнительным параметром прозрачности (Alpha), который определяет степень прозрачности цвета. Прозрачность задается числом от 0 (полностью прозрачный) до 1 (полностью непрозрачный).

HEX и системные цвета

HEX в CSS обозначает шестнадцатеричное представление цвета. В CSS шестнадцатеричный формат используется для представления цветов с использованием комбинации шести символов: трех пар чисел и/или букв от 0 до F, где каждая пара обозначает уровень красного, зеленого и синего цветов соответственно.

- #FF0000: Красный цвет.
- #00FF00: Зеленый цвет.
- #0000FF: Синий цвет.

используем # для уточнения что это HEX

HSL и HSLA

HSL (Hue, Saturation, Lightness): Этот формат представляет цвета, используя оттенок (Hue), насыщенность (Saturation) и светлоту (Lightness). Оттенок измеряется в градусах на цветовом круге (от 0 до 360), насыщенность и светлота измеряются в процентах.

- `hsl(0, 100%, 50%)`: Красный цвет.
- `hsl(120, 100%, 50%)`: Зеленый цвет.
- `hsl(240, 100%, 50%)`: Синий цвет.

HSLA (HSL с альфа-каналом): Аналогично HSL, но с дополнительным параметром прозрачности (Alpha), который определяет степень прозрачности цвета. Прозрачность задается числом от 0 (полностью прозрачный) до 1 (полностью непрозрачный).

Работа с “длиной” в CSS

Чаще всего используем PX - явное указание размера;

Для адаптивности размеров элементов - % и VH и VW;

Для адаптивного шрифта Em и Rem;

Все остальное используем не чаще чем раз в 100 лет (знать что есть но не учить как пользоваться, легко “гуглится”;

Почему важно знать?

- Для компетентности (ответ на вопрос во время собеседования);
- Если встретится в легаси кода - не потеряетесь.

Обычные единицы измерения (прямо как в жизни)

в CSS также существуют единицы измерения для определения размеров в физических единицах, таких как дюймы (inches), сантиметры (centimeters), миллиметры (millimeters), пункты (points) и пики (picas)

IN (Inches - Дюймы): 1 дюйм равен 2.54 сантиметрам;

CM (Centimeters - Сантиметры);

MM (Millimeters - Миллиметры);

PT (Points - Пункты): 1 пункт равен $\frac{1}{72}$ дюйма;

PC (Picas - Пики): 1 пика равен 12 пунктам или $\frac{1}{6}$ дюйма.

РХ - пиксели

Единица измерения `px` в CSS обозначает пиксели (pixels). Пиксель - это наименьшая отдельная точка на экране, и веб-страница часто отображается в браузере с использованием этой единицы измерения.

- Фиксированный размер: `px` представляет абсолютное значение размера, что означает, что элементы, определенные в `px`, будут иметь фиксированный размер независимо от размера экрана или устройства просмотра. Например, `width: 100px;` задает ширину элемента в 100 пикселей.
- Подходит для точной работы: Использование `px` подходит для точного контроля над размерами и расположением элементов на веб-странице, особенно если нужно точно соответствовать дизайну.
- Не рекомендуется для резиновых макетов: Использование `px` может привести к проблемам при создании резиновых (flexible) макетов, которые должны адаптироваться к разным размерам экранов. В таких случаях рекомендуется использовать относительные единицы измерения, такие как `%`, `em`, `rem`, `vw`, `vh`, и т.д.
- Поддерживается всеми браузерами: Единица измерения `px` широко поддерживается всеми современными браузерами, и она является стандартной единицей измерения в CSS.

rem и em - относительные размеры (зависит от шрифта)

Единицы измерения em и rem в CSS также используются для задания размеров, но они являются относительными единицами измерения, которые зависят от размера шрифта. Вот как они работают:

em определяет размер относительно размера шрифта родительского элемента.

rem (root em) определяет размер относительно размера шрифта корневого элемента (html).

По умолчанию размер шрифта 16px

% - процентное соотношение

Ширина и высота: Установка ширины и высоты элемента в процентах относительно размера его родителя. Например, `width: 50%`; означает, что ширина элемента будет составлять половину ширины его родителя.

Отступы (`margin`) и внутренние отступы (`padding`): Установка отступов в процентах позволяет создавать адаптивные и резиновые макеты. Например, `margin-left: 10%`; создаст отступ слева, равный 10% ширины родительского элемента.

Позиционирование: В связи с позиционированием элементов, проценты часто используются для определения положения элемента относительно его родительского блока.

Шрифты: Проценты также могут применяться для определения размера шрифта. Например, `font-size: 120%`; увеличит размер шрифта на 20% относительно размера шрифта родительского элемента.

Размер родителя, от которого идет процентный расчет, должен быть явно указан

VW и VH

Относительные единицы измерения `vh` и `vw` в CSS используются для задания размеров элементов на основе высоты и ширины видимой области окна браузера (viewport). Вот как они работают:

1 `vh` равен 1% высоты видимой области окна браузера.

1 `vw` равен 1% ширины видимой области окна браузера.

`vh` и `vw` часто используются для создания адаптивных и отзывчивых дизайнов, особенно для элементов, которые должны занимать определенную часть экрана независимо от его размера

vmin и vmax

vmin и vmax - это относительные единицы измерения в CSS, которые используются для задания размеров элементов в зависимости от размеров видимой области окна браузера (viewport). Они учитывают минимальные и максимальные размеры окна браузера для определения размеров элементов

- vmin определяет размер относительно минимального измерения видимой области окна браузера.
- vmax определяет размер относительно максимального измерения видимой области окна браузера.
- Например, если ширина окна равна 800 пикселей, а высота 600 пикселей, то 1 vmin будет равен 6 пикселям (минимальное значение), 1 vmax также будет равен 8 пикселям (максимальное значение).

ex и ch (прям духота)

В CSS единицы ex и ch используются для определения размеров, связанных с текстом, а именно с высотой строчной буквы (ex) и шириной символа "0" (ch).

- ex определяет высоту строчной буквы в текущем шрифте.
- ch определяет ширину символа "0" в текущем шрифте.

CALC - вычисления

Функция `calc()` в CSS позволяет выполнять арифметические операции при задании значений свойств. Она чрезвычайно полезна, когда требуется комбинировать различные единицы измерения или проводить вычисления на основе других значений.

СИНТАКСИС:

CSS

```
property: calc(expression);
```

- `calc()` позволяет использовать различные единицы измерения внутри одного выражения, что делает ее мощным инструментом для создания адаптивных макетов.
- Выражение внутри `calc()` должно быть заключено в скобки.
- Поддержка `calc()` хорошо поддерживается современными браузерами и может использоваться в большинстве случаев без ограничений.

var - css переменные

В CSS переменные (CSS variables), также известные как Custom Properties, позволяют хранить значения свойств CSS для повторного использования в различных местах вашего стиля. Они предоставляют гибкость и удобство при настройке дизайна веб-страницы.

- Переменные объявляются в корне CSS или в селекторах.
- Их имя начинается с двойного дефиса (--) и далее следует имя переменной.
- Значение переменной задается с помощью свойства var().

CSS

```
:root {  
  --primary-color: #3498db;  
}
```

CSS

```
.element {  
  background-color: var(--primary-color);  
}
```

Время, углы и абсолютные значения

Время: s - секунда, ms - миллисекунда.

Углы:

- deg - градусы;
- rad - радианы (180 градусов/Пи);
- turn (оборот 360 градусов).

Числа без единиц измерения:

прозрачность (opacity), z-index (вложенность по оси Z), font-weight (жирность текста)

Свойства для изучения размеров

width - ширина

height - высота

background-color - цвет заднего фона

color - цвет текста

font-size - размер шрифта (текста)

font-weight - “жирность” шрифта (текста)

Домашнее задание

Сверстать свою таблицу веса селекторов

Сверстать блоки с “весами” 0123, 1000, 0234.

Сверстать 6 прямоугольников разных цветов и размеров используя разные методы указания как высоты и ширины так и цвета.

Сверстать 3 прямоугольника с текстом внутри так, чтобы размер прямоугольника и текста менялся при изменении экрана

Сверстать пару прямоугольников с текстом так, чтобы размеры как текста так и прямоугольников зависели от переменных и были использованы вычисляемые свойства calc

Ресурсы

Вес селекторов - [ТЫК](#)

Единицы размеров - [ТЫК](#)

Цвета css - [ТЫК](#)

переменные css (var) - [ТЫК](#)

вычисления в css (calc) - [ТЫК](#)