



Единицы измерения

px % em rem vh vw vmin vmax



Абсолютная единица - **РХ**

Пиксель рх – это самая базовая, абсолютная и окончательная единица измерения.

Количество пикселей задаётся в настройках разрешения экрана, один рх – это как раз один такой пиксель на экране. Все значения браузер в итоге пересчитает в пиксели.

Пиксели могут быть дробными, например размер можно задать в 16.5рх. Браузер использует дробные пиксели для внутренних вычислений.

Пример: есть элемент шириной в 100рх, его нужно разделить на три части – появляются 3 части по 33.333...рх. При окончательном отображении дробные пиксели округляются и становятся целыми. Для мобильных устройств, у которых много пикселей на экране, но сам экран маленький, чтобы обеспечить читаемость, браузер автоматически применяет масштабирование.



Абсолютная единица - **РХ**

Достоинства:

Четкость и понятность

Недостатки:

Не позволяет устанавливать соотношение между различными размерами

Не подходит для adaptive/responsive вёрстки (за исключением max и min size)

px это “каменный век” в современной web разработке



Относительные единицы - %

Единицы **px** не лучшим образом подходят для веб-документов, что заставляет нас использовать относительные единицы. В теории, единицы **%** и **em** являются идентичными, но на практике они имеют незначительные различия, которые важно учитывать.

Рассмотрим пример, где будем использовать в качестве базовой единицы font-size проценты (в тэге **body**). Если вы измените вашу базовую единицу font-size с **%** на **em** (то есть **body {font-size: 1em;}**), вы должны бы не заметить разницы. Но, посмотрите, что происходит, когда «1em» является нашей базовой единицей, и когда клиент меняет размер шрифта в настройках своего браузера.

Относительные единицы - %

	body { font-size: 100%; }	body { font-size: 120%; }
font-size: 1em	The quick brown fox	The quick brown
font-size: 12pt	The quick brown fox	The quick brown fox
font-size: 16px	The quick brown fox	The quick brown fox
font-size: 100%	The quick brown fox	The quick brown

© KyleScheffer.com

Как вы можете видеть, **em** и **%** увеличили размер шрифта, в то время как **px** этого не сделал.

Установка абсолютного размера для вашего текста может быть простым делом, но гораздо лучше для ваших посетителей использовать масштабируемый текст, который может быть отображен на любом устройстве или любой машине. По этой причине, единицы **em** и **%** предпочтительнее в использовании для текста веб-документа.



Вечный спор - **EM** VS **REM**

EM - это единица изменения в типографии, равное текущему указанному размеру шрифта. Название **em** связано с буквой **M**. Изначально данная единица была получена из ширины заглавной буквы **M** при использовании определенного шрифта. **Зависит от размера родителя.**

REM (root EM) - произошел от **em**. Главное достоинство - дает возможность сделать так, чтобы значения единиц **em** во всем HTML-документе были основаны на одном «главном» значении, указанном в «html»-элементе. Все **em**-единицы, использованные в документе, будут основаны на главном значении **rem**. **Зависит от размера корневого элемента(html)**

rem (root em) units - CR

Global

97.47% + 0.29% = 97.76%

Type of unit similar to `em`, but relative only to the root element, not any parent element. Thus compounding does not occur as it does with `em` units.

Current aligned

Usage relative

Date relative

Show all


IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome for Android	UC Browser for Android	Samsung Internet
			49		9.3				
			61		10.2				
	15		62	10.1	10.3				4
11	16	57	63	11	11.2	all	62	11.4	6.2
	17	58	64	TP					
		59	65						
		60	66						

Notes

Known issues (5)

Resources (3)

Feedback



Что выбрать? **EM** или **REM**?

Каждый может определить для себя с чем работать удобнее. Вычислять все эти числа в голове довольно затруднительно и долго. К счастью, вам не следует беспокоиться, если вы используете один из CSS препроцессоров Sass/SCSS, LESS. Все цифры можно вычислять с помощью функций.

Мой вариант:

```
body { font-size 62.5; } + rem
```




Новые единицы в CSS3 - **vw** **vh** **vmin** **vmax**

vw и **vh** единицы вычисляются относительно размеров окна браузера. Для настольных компьютеров ширина окна браузера больше ширины области просмотра (добавляется ширина скроллбара), поэтому если для элемента установить ширину **100vw**, то он выйдет за пределы **html**.

Так как ширина элемента, указанная с помощью **100vw** больше ширины области просмотра, то для создания полноэкранных фоновых изображений лучше использовать ширину **100%**, которая будет равна ширине корневого элемента **html**.

1 vh = 1% высоты окна браузера.

1 vw = 1% ширины окна браузера.



Новые единицы в CSS3 - **vw** **vh** **vm** **vm**

vm и **vm** – отличная замена или дополнение к CSS медиа запросам по ориентации экрана (@media screen and (orientation : portrait) или @media screen and (orientation : landscape)), так как они моментально ощущают изменение соотношения сторон окна браузера.

В то время, как единицы **vh** и **vw** всегда вычисляются относительно соответствующих размеров окна браузера, **vm** и **vm** определяются минимальным или максимальным значением высоты или ширины. Например, если ширина окна браузера равна **1200px**, а высота **700px**, то **vm** будет равен **7px**, а **vm** — **12px**.

1 vm = 1% меньшего размера окна браузера по высоте или ширине.

1 vm = 1% большего размера окна браузера по высоте или ширине.

Viewport units: vw, vh, vmin, vmax - CR

Global

89.35% + 4.6% = 93.94%

Length units representing a percentage of the current viewport dimensions: width (vw), height (vh), the smaller of the two (vmin), or the larger of the two (vmax).

Current aligned										
Usage relative										
Date relative										
Show all										
IE	Edge *	Firefox	Chrome	Safari	iOS Safari *	Opera Mini *	Chrome for Android	UC Browser for Android	Samsung Internet	
			49		9.3					
			61		10.2					
	² 15		62	10.1	10.3				4	
² 11	16	57	63	11	11.2	all	62	11.4	6.2	
	17	58	64	TP						
		59	65							
		60	66							

Notes

Known issues (11)

Resources (7)

Feedback

² Partial support refers to not supporting the "vmax" unit.



Подводим итоги

Мы рассмотрели единицы измерения:

px – абсолютные пиксели. Используется для максимально конкретного и точного задания размеров.

em – задаёт размер относительно шрифта родителя, используется там, где нужно упростить масштабирование компонентов.

rem – задаёт размер относительно шрифта **<html>**, используется для удобства глобального масштабирования: элементы которые планируется масштабировать, задаются в rem, а **JS** меняет шрифт у **<html>**.

% – относительно такого же свойства родителя (как правило, но не всегда), используется для ширин, высот и так далее, без него никуда, но надо знать, относительно чего он считает проценты.

vw, vh, vmin, vmax – относительно размера экрана.



Спасибо! :)

Материалы и статьи используемые в докладе:

<https://www.w3.org/> <https://css-tricks.com/> <https://codepen.io/> <http://thenewcode.com/> <https://caniuse.com/>

Единицы измерения

<https://learn.javascript.ru/css-units>

Почему не стоит использовать px

<https://webformyself.com/em-rem-ili-px-pochemu-nelzya-ispolzovat-tolko-pikseli/>

<https://engageinteractive.co.uk/blog/em-vs-rem-vs-px>

Почитать о em и rem

<http://www.coolwebmasters.com/cssstyle-sheets/2491-understanding-em-units-css.html>

<http://www.coolwebmasters.com/cssstyle-sheets/2491-understanding-em-units-css.html>

em vs. px vs. pt vs. percent

<https://habrahabr.ru/post/42151/>

em vs rem

<https://habrahabr.ru/post/280125/>

vh vw vmin vmax

<https://html5book.ru/edinicy-izmereniya-vh-vw-vmin-vmax/>

<https://habrahabr.ru/post/126863/>

Курс от **Loftschool**: “Веб разработка для начинающих”

Наставник группы: **Болотова Ольга**

Доклад подготовил: **Брюханов Александр**