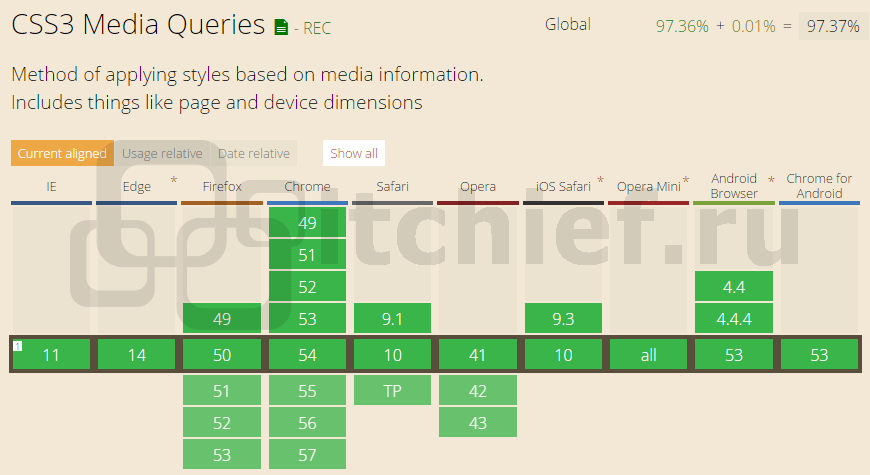
Что такое медиа запросы?

Медиа запросы (media queries) - это правила CSS, которые позволяют управлять стилями элементов в зависимости от значений технических параметров устройств. Иными словами, это конструкции, которые позволяют определять на основании некоторых условий какие стили необходимо использовать на веб-странице, а какие нет.

Медиа запросы появились в спецификации CSS3 и на сегодняшний день поддерживаются во всех современных браузерах (Chrome 4+, Firefox 3.5+, IE 9+, Opera 9+, Safari 4+).

Поддержка браузерами CSS3 медиа запросов (media queries)

Поддержка медиа запросов в браузере IE8 осуществляется посредством подключения к странице скрипта respond.js:

<!-- Respond.js для IE8 (media queries) -->

<!-- ПРЕДУПРЕЖДЕНИЕ: Respond.js не будет работать при просмотре страницы через file:// -->

<!--[if lt IE 9]>

<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>

<![endif]-->

Медиа запросы предназначены для создания адаптивных дизайнов сайтов. Адаптивный дизайн отличается от других тем, что он может "приспосабливаться" (видоизменяться) в зависимости от того, какую ширину экрана имеет устройство (браузер). Более подробно познакомиться с адаптивным дизайном можно в статье ["Что такое адаптивная разметка"](https://itchief.ru/lessons/bootstrap-3/lesson-no-6-adaptive-site-layout-on-the-engine,-and-twitter-bootstrap).

Но при создании адаптивных веб-страниц также необходимо обратить внимание на метатег viewport. Данный тег обеспечивает корректное отображение дизайнов адаптивных сайтов на экранах устройств, имеющих высокую плотность пикселей. Иными словами, он устанавливает соответствие между CSS и физическим разрешением веб-страницы. Более подробно разобраться, как работает метатег viewport можно в статье ["Знакомство с meta viewport"](https://itchief.ru/lessons/html-and-css/meta-viewport-how-it-works).

Подключение метатега viewport к странице в большинстве случаях осуществляется так:

<meta name="viewport" content="width=device-width, initial-scale=1">

Синтаксис медиа запросов

Для создания медиа запросов используется следующий синтаксис:

@media **условие** {

/\* стили (они будут выполняться, если устройство соответствует указанному условию)

}

Основные типы устройств:

* all - все устройства (по умолчанию).
* print - принтеры и режим предварительного просмотра страницы перед печатью.
* screen - устройства с дисплеями.

Логические операторы:

* and - требует обязательного выполнения всех указанных условий.  
  Например:

@media screen and (min-width: 1200px) and (orientation: landscape) { /\* Стили CSS ... \*/ }

Стили CSS в вышеприведённом примере выполняться только в том случае, если страница будет выводиться на устройство с экраном, иметь область просмотра более 1200 пикселей в ширину, а также находиться в альбомном режиме.

* , (запятая) - требует обязательного выполнения хотя бы одного из указанных условий в медиа запросе.

@media (min-width: 544px), (orientation: landscape) { /\* Стили CSS ... \*/ }

Стили CSS в этом примере будут применяться к странице в двух случаях. Т.е. тогда, когда устройство будет иметь viewport не менее 544 пикселей (включительно) или ориентацию landscape.

* not - предназначен для отрицания указанного условия. Имеет по отношению к оператору and меньший приоритет, т.е. оператор not всегда выполняется после and.

@media not screen and (orientation: portrait), (min-width: 992px) { /\* Стили CSS ... \*/ }

Стили CSS, находящиеся в этом правиле, будут применены к странице только в том случае, если устройство не является screen и не будет иметь портретную ориентацию. А также они (стили CSS) будут применены к элементам документа ещё тогда, когда устройство (браузер) будет иметь ширину рабочей области не менее 992 пикселя (включительно).  
Т.е. запрос в вышеприведённом примере будет обрабатываться так:

@media not (screen and (orientation: portrait)), (min-width: 992px) { /\* Стили CSS ... \*/ }

Медиа функции

Для составления условия в @media можно использовать следующие фукнции:

* width - указывает требования к ширине области просмотра устройства (браузера).
  1. /\* применить стили CSS только для устройств с шириной области просмотра, равной 320px \*/
  2. @media (width: 320px) { /\* Стили CSS ... \*/ }
* min-width - задаёт минимальную ширину области viewport в px, em или других единицах.
  1. /\* для устройств (браузеров), которые предоставляют для страницы минимальную ширину области просмотра, равную 544 пикселя \*/
  2. @media (min-width: 544px) { /\* Стили CSS ... \*/ }
* max-width - указывает на то, какой должна быть максимальная рабочая область устройства (браузера).
  1. /\* стили, которые будут применены к элементам страницы с рабочей областью не больше 1199 пикселей \*/
  2. @media (max-width: 1199px) { /\* Стили CSS ... \*/ }
* height, min-height и max-height - задают требования аналогично вышеприведённым функциям, но в отношении высоты viewport.
  1. /\* стили, которые будут применены к элементам страницы в том случае, если viewport браузера будет больше 720px в высоту \*/
  2. @media (min-height: 720px) { /\* Стили CSS ... \*/ }
* orientation - функция, которая проверяет то, в каком режиме (portrait или landscape) отображается страница.  
  Пример, в котором в зависимости от ориентации экрана, отображается одна или другая картинка:
  1. /\* landscape (альбомный) - это режим, в котором наоборот ширина viewport больше её высоты \*/
  2. @media (orientation: landscape) {
  3. #background-image { background: url(image1.png) no-repeat; }
  4. }
  6. /\* portrait (портретный) - это режим, в котором высота viewport больше ширины \*/
  7. @media (orientation: portrait) {
  8. #background-image { background: url(image2.png) no-repeat; }
  9. }
* aspect-ratio (min-aspect-ratio, max-aspect-ratio) - позволяют указать то, как ширина устройства должна относиться к высоте. В качестве значений допускается использовать только целые значения.
  1. /\* для дисплеев с соотношением сторон 16/9 \*/
  2. @media screen and (device-aspect-ratio: 16/9) { /\* Стили CSS ... \*/ }
  4. /\* для дисплеев с соотношением сторон 1336/768 \*/
  5. @media screen and (device-aspect-ratio: 1336/768) { /\* Стили CSS ... \*/ }
* resolution (min-resolution, max-resolution) - указывает разрешение (плотность пикселей) устройства вывода. В качестве единиц измерения разрешения используются следующие величины: dpi (количество точек на дюйм), dpcm (количество точек на сантиметр), dppx(количество точек на пиксель).
  1. /\* для экранов, имеющих высокую плотность пикселей
  2. (т.е. для таких, у которых отношение аппаратных пикселей к CSS не менее 2) \*/
  3. @media screen and (min-resolution: 2dppx) { /\* Стили CSS ... \*/ }
  5. /\* при печати с разрешением свыше 300 точек на дюйм \*/
  6. @media print and (min-resolution: 300dpi) { /\* Стили CSS ... \*/ }

Использование медиа-запросов при подключении файлов CSS

Медиа запросы также можно применять в качестве значения атрибута media элемента link. Это позволит в зависимости от параметров устройства определить, какие файлы CSS необходимо подсоединить к странице, а какие нет. Обычно данный функционал используется тогда, когда к разным классам устройств необходимо применить различные стили CSS.

<!-- Стили xs-styles.css будут подсоединены к странице только на устройствах c шириной меньше 543 пикселей (включительно) -->

<link rel="stylesheet" media="screen and (max-width: 543px)" href="styles-xs.css">

Кроме этого медиа запросы можно также использовать в правиле @import, которое предназначено для импортирования стилей из других файлов CSS в текущий.

/\* импортирование стилей из файла styles-xs.css в текущий файл стилей только для устройств, которые предоставляют веб-странице viewport, имеющий ширину 543 пикселя или меньше. \*/

@import url(styles-xs.css) (max-width: 543px);

Медиа запросы для Bootstrap 3

Организация media queries в порядке возрастания классов устройств xs, sm, md и lg (по умолчанию):

1. /\* Устройства с очень маленьким экраном (смартфоны, меньше 768px) \*/
2. /\* Стили CSS (по умолчанию) - для ширины viewport <768px \*/
4. /\* Устроства с маленьким экраном (планшеты, 768px и выше) \*/
5. @media (min-width: 768px) {
6. /\* Стили для устройств с шириной viewport, находящейся в диапазоне 768px - 991px \*/
7. }
9. /\* Устройства со средним экраном (ноутбуки и компьютеры, 992px и выше) \*/
10. @media (min-width: 992px) {
11. /\* Стили для устройств с шириной viewport, находящейся в диапазоне 992px - 1199px \*/
12. }
14. /\* Устройства с большим экраном (компьютеры, 1200px и выше) \*/
15. @media (min-width: 1200px) {
16. /\* Стили для устройств с шириной viewport >1200px \*/
17. }

Вышеприведённые запросы необходимо использовать только в указанном порядке.

Для того чтобы media запросы можно было применять в какой угодной последовательности, их необходимо расширить включив в них дополнительно выражение max-width. Это заставит их работать только в указанном диапазоне.

1. @media (max-width: 767px) {
2. /\* стили для xs-устройств \*/
3. }
4. @media (min-width: 768px) and (max-width: 991px) {
5. /\* стили для sm-устройств \*/
6. }
7. @media (min-width: 991px) and (max-width: 1199px) {
8. /\* стили для md-устройств \*/
9. }
10. @media (min-width: 1200px) {
11. /\* стили для lg-устройств \*/
12. }

Медиа запросы для Bootstrap 4

Синтаксис медиа-запросов для Bootstrap 4, которые можно использовать только в следующем порядке (последовательного увеличения минимальной ширины viewport):

1. /\* xs - устройства (до 576px) \*/
2. /\* CSS для ширины, которая меньше 575px (включительно) \*/
4. /\* sm-устройства (больше или равно 576px) \*/
5. @media (min-width: 576px) {
6. /\* CSS для: 576px <= ширины <= 767px \*/
7. }
9. /\* md-устройства (больше или равно 768px) \*/
10. @media (min-width: 768px) {
11. /\* CSS для: 768px <= ширины <= 991px \*/
12. }
14. /\* lg-устройства (больше или равно 992px) \*/
15. @media (min-width: 992px) {
16. /\* CSS для: 992px <= ширины <= 1119px \*/
17. }
19. /\* xl-устройства (больше или равно 1200px) \*/
20. @media (min-width: 1200px) {
21. /\* CSS для: ширины >= 1200px \*/
22. }

Список media запросов для фреймворка Bootstrap 4, которые можно применять только в обратном порядке (в порядке убывания ширины области просмотра окна браузера):

1. /\* xl-размер (>=1200px) \*/
2. /\* CSS для >=1200px \*/
4. /\* lg-размер (<=1199px) \*/
5. @media (max-width: 1199px) {
6. /\* CSS для ширины от 992px до 1199px \*/
7. }
9. /\* md-размер (<=991px) \*/
10. @media (max-width: 991px) {
11. /\* CSS для ширины от 768px до 991px \*/
12. }
14. /\* sm-размер (<=768px) \*/
15. @media (max-width: 767px) {
16. /\* CSS для ширины от 576px до 767px \*/
17. }
19. /\* xs-размер (<=575px) \*/
20. @media (max-width: 575px) {
21. /\* CSS для ширины до 575px (включительно) \*/
22. }

Перечень медиа-запросов для Bootstrap 4, которые можно использовать в таблице стилей в любой последовательности:

1. /\* xs (<=543px) \*/
2. @media (max-width: 575px) { ... }
4. /\* sm (>=576 и <=767) \*/
5. @media (min-width: 576px) and (max-width: 767px) { ... }
7. /\* md (>=768 и <=991) \*/
8. @media (min-width: 768px) and (max-width: 991px) { ... }
10. /\* lg (>=992 и <=1199) \*/
11. @media (min-width: 992px) and (max-width: 1199px) { ... }
13. /\* xl (>=1200) \*/
14. @media (min-width: 1200px) { ... }

Код JavaScript, учитывающий параметры устройств

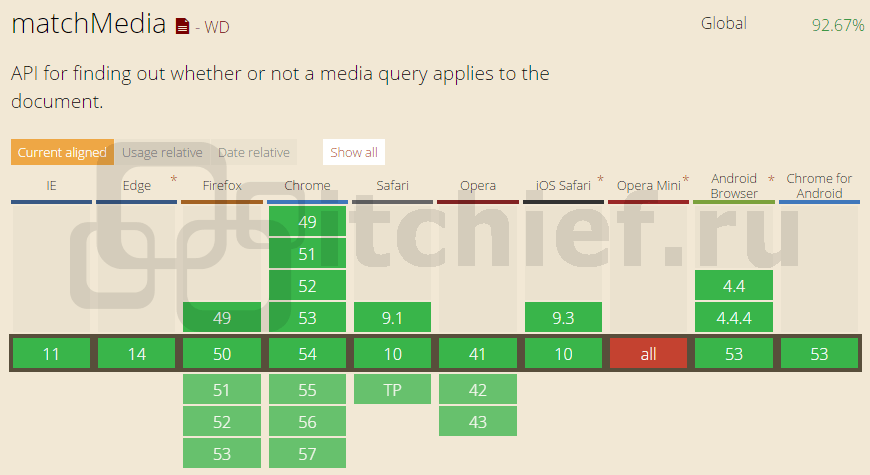
Наиболее простой способ создания кода JavaScript, учитывающий параметры устройств (аналогично CSS медиа запросам), осуществляется с помощью метода matchMedia объекта window.

Осуществляется это следующим образом:

1. // например, проверим, соответствует ли указанный медиа запрос (screen and (max-width: 543px)) устройству
2. // результат проверки можно получить с помощью свойства matches (true или false)
3. if (window.matchMedia('screen and (max-width: 543px)').matches) {
4. // ... действия, если устройство отвечает медиа запросу
5. } else {
6. // ... действия, если устройство не соответствует значениям медиа запроса
7. }

Например, эту возможность можно применить для асинхронной загрузки картинок в зависимости от того какой размер viewport имеет устройство (браузер).

Метод matchMedia не поддерживается Internet Explorer 9 и другими старыми браузерами. Для того чтобы обеспечить эту функциональность в старых браузерах можно воспользоваться методом mq библиотеки Modernizr.

Поддержка браузерами метода matchMedia (JavaScript) - ноябрь 2016

1. if (Modernizr.mq('(max-width: 767px)')) {
2. // ... действия, если устройство соответсвуют указанному медиа условию
3. } else {
4. // ... действия, если устройство не отвечает заданному медиа условию
5. }