

CSS MEDIA QUERIES

Andrew Sheehan

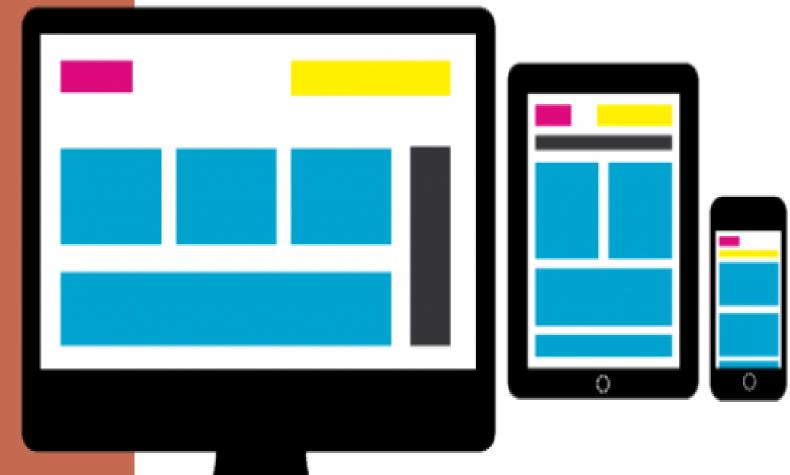
Boston University
Computer Science Dept/MET

S \triangleleft



caniuse.com (December, 2023)

WHAT ARE MEDIA QUERIES



Control the way your webapp looks on different devices & orientations

MOBILE-FIRST DESIGN





A media query is composed of a media

type + expressions

case-insensitive



@media screen (min-width: 320px) and (max-width: 768px)

AT-RULE

MEDIA TYPE

MEDIA FEATURE

OPERATOR

MEDIA FEATURE



CONTENT CAN ADAPT TO THE DEVICE





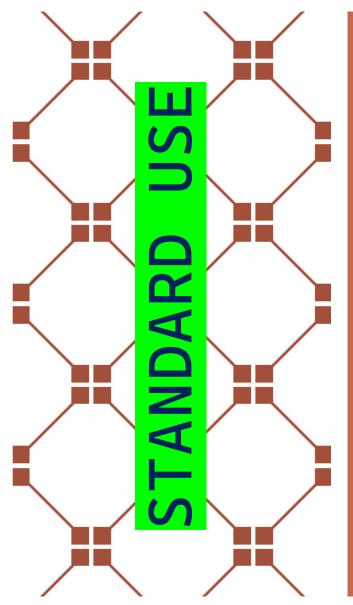
SPELLING/FORMAT MATT3RS

Misspelling your media type or having invalid logic/syntax will result in the media query not working



Value	Description
all	Used for all media type devices
print	Used for printers
screen	Used for computer screens, tablets, smart-phones etc.

ALLOWABLE MEDIA TYPES



```
@media screen and (max-width: 520px) {
   /* CSS rules go inside brackets */
@media screen and (max-width: 980px) {
   /* CSS rules go inside brackets */
@media screen and (max-width: 1440px) {
   /* CSS rules go inside brackets */
```

Shift or change content to be the width of the device



EXPRESSIONS MUST EVALUATE TO TRUE



- i. The media type of the media query matches the media type of the device.
- ii. All expressions in the media query eval to true.
- iii. If the media query list is empty (i.e. the declaration is the empty string or consists solely of whitespace) it evaluates to true.

```
@media (max-width: 2082px) {
    #main_content_area {
        display: block;
    }
}
```

Display that target a block-level element when the viewport width is equal to or less than 2082 pixels

PRINTING

When a user wants to print, they ~probably~ do not want to waste their color ink on graphics or unnecessary content.

```
@media print {
    #header,#left-side-ads,#menu {
        display: none;
    }
}
```



```
<link rel="stylesheet" type="text/css"
media="screen" href="sans-serif.css">
```

```
<link rel="stylesheet" type="text/css"
media="print" href="serif.css">
```

USING THE LINK ELEMENT

LOGICAL NEGATION (NOT)



The **NOt** keyword inverts the meaning of an entire media query. It will only negate the specific media query it is applied to. The not keyword can't be used to negate an individual feature query, only an entire media query.

That means they can be prefixed with "min-" or "max-" to express "minimum condition" or "maximum condition" constraints.

For example, this CSS will apply styles only if your browser's viewport width is equal to or narrower than 1250px:

```
@media (max-width: 1250px)
{
   /* CSS rules go here*/
}
```

Range Features



EXAMPLES

```
@media (min-width: 30em) and (orientation: landscape) { ... }
```

Viewport set to landscape mode with a width of at least 30em's

```
@media (min-height: 680px), screen and (orientation: portrait) { ... }
```

Viewport with at least a height of 680 pixels or using a screen display in portrait mode

```
@media (30em <= width <= 50em ) { ... }</pre>
```

Viewport min-width and max-width between 30em & 50em (Example using new, Level 4 media query syntax)

```
/* When the browser is at least 600px and above */
@media screen and (min-width: 600px) {
    .element {
        /* Apply some styles */
}
```

```
@media print and ( not(color) ) {
   body {
     background-color: none;
```