How to apply specific CSS rules to Chrome only?

Asked 7 years, 8 months ago Active 1 year, 10 months ago Viewed 227k times

96

20

Is there a way to apply the following CSS to a specific `div` only in Google Chrome?

```
position:relative;
top:-2px;
```

css google-chrome css-hack

edited Dec 15 '17 at 21:08
reformed
2,496 7 47 67

asked Feb 17 '12 at 13:10
user1213707
611 2 7 6

1 possible duplicate of [How to do a Chrome/Opera specific stylesheet?](#) – Jakub Feb 17 '12 at 13:12

1 What issue are you facing that forces you to do this? Targeting CSS rules for specific browsers is not great design and in most cases, shouldn't be necessary any more nowadays. – Pekka supports GoFundMonica Feb 17 '12 at 13:12 ✎

@Pekka this is my issue [stackoverflow.com/questions/9311965/…](#) , and i found that the only solution is to add those but only for chrome, please help :(– user1213707 Feb 17 '12 at 13:14

Did the answer in your original question not help? – Pekka supports GoFundMonica Feb 17 '12 at 13:15

2 Personally I develop for Chrome (which tends to copy over to Firefox) and worry about IE at the end. – SpaceBeers Feb 17 '12 at 13:27

11 Answers

190

CSS Solution

from https://jeffclayton.wordpress.com/2015/08/10/1279/

```
/* Chrome, Safari, AND NOW ALSO the Edge Browser and Firefox */
@media and (-webkit-min-device-pixel-ratio:0) {
  div{top:10;}
}

/* Chrome 29+ */
@media screen and (-webkit-min-device-pixel-ratio:0)
  and (min-resolution:.001dpcm) {
    div{top:0;}
}

/* Chrome 22-28 */
@media screen and(-webkit-min-device-pixel-ratio:0) {
  .selector {-chrome-:only(;
     property:value;
  );}
}
```

JavaScript Solution

```
if (navigator.appVersion.indexOf("Chrome/") != -1) {
// modify button
}
```

edited Mar 25 '16 at 10:11
Henry Henrinson
2,289 6 33 64

answered Nov 27 '12 at 15:23
Martin Kristiansson
1,949 1 10 5

16 This rule will apply both safari and chrome – Miuranga Jun 28 '13 at 15:38

1 @AlirezaNoori probably because it applies to both Chrome *and* Safari, not *only* Chrome. – thom_nic Jun 10 '14 at 15:19

1 IE 8 seems to accept this CSS also :(– Sam Watkins Dec 3 '15 at 2:21

1 It's also works in IE Edge, check this jeffclayton.wordpress.com/2015/08/10/1279 – llamerr Mar 25 '16 at 9:57

1 I used `data-ng-class` for angular and added a `.chrome` class with the JS expression. Worked like a charm. Thanks – Kraken Aug 24 '16 at 13:20

As we know,Chrome is a Webkit browser,Safari is a Webkit browser too,and Also Opera,so it's very hard to target the Google Chrome,using media queries or CSS hacks,but Javascript is really more effective.

24

Here is the piece of Javascript code that will target Google Chrome 14 and later,

```
var isChrome = !!window.chrome && !!window.chrome.webstore;
```

and below is a list of Available Browser hacks,for the Google chrome including the influenced browser,by that hack

WebKit hack:

```
.selector:not(*:root) {}
```

- **Google Chrome**:***All the versions***
- **Safari**:***All the versions***
- **Opera *:14 and Later***
- **Android**:***All the versions***

Supports Hacks:

```
@supports (-webkit-appearance:none) {}
```

Google Chrome 28,and Google Chrome > 28, Opera 14 and Opera > 14

- **Google Chrome**:***28 and Later***
- **Opera *:14 and Later***

Property/Value Hacks:

```
.selector { (;property: value;); }
.selector { [;property: value;]; }
```

Google Chrome 28,and Google Chrome < 28, Opera 14 and Opera > 14,and Safari 7 and Less than 7. - **Google Chrome**:***28 and Before*** - **Safari**:***7 and Before*** - **Opera *:14 and Later***

JavaScript Hacks:1

```
var isChromium = !!window.chrome;
```

- **Google Chrome**:***All the versions***
- **Opera** ***:14 and Later***
- **Android**:***4.0.4***

JavaScript Hacks:2 {Webkit}

```
var isWebkit = 'WebkitAppearance' in document.documentElement.style;
```

- **Google Chrome**:***All the versions***
- **Safari**:***3 and Later***
- **Opera** ***:14 and Later***

JavaScript Hacks:3

```
var isChrome = !!window.chrome && !!window.chrome.webstore;
```

- **Google Chrome**:***14 and Later***

Media Query Hacks:1

```
@media \\0 screen {}
```

- **Google Chrome**:***22 to 28***
- **Safari**:***7 and Later***

Media Query Hacks:2

```
@media all and (-webkit-min-device-pixel-ratio:0) and (min-resolution: .001dpcm) {
.selector {} }
```

- **Google Chrome**:***29 and Later***

- **Opera**:***16 and Later***

For more information please visit this website

answered Aug 26 '14 at 1:30
Hbirjand
1,669 15 25

1 Beware: Safari also supports the @supports ... – sebilasse Feb 12 '16 at 10:18

as Sebastian said @supports (-webkit-appearance:none) {} affects Safari, tested on v.10 – Dmitrii Malyshev Oct 21 '16 at 14:31

1 `@supports (-webkit-appearance:none) { }` is now working for MS Edge. – Vadim Ovchinnikov Aug 4 '17 at 7:32

13

An update for chrome > 29 and Safari > 8 :

Safari now supports the `@supports` feature too. That means those hacks would also be valid for Safari.

I would recommend

@ http://codepen.io/sebilasse/pen/BjMoye

```
/* Chrome only: */
@media all and (-webkit-min-device-pixel-ratio:0) and (min-resolution: .001dpcm) {
  p {
    color: red;
  }
}
```

answered Feb 12 '16 at 11:02
sebilasse
2,372 1 22 28

1 The text is also red in FireFox for me. – Kerry7777 Jun 19 at 1:13

This css browser selector may help you. Take a look.

6

CSS Browser Selector is a very small javascript with just one line which empower CSS selectors. It gives you the ability to write specific CSS code for each operating system and each browser.

answered Feb 17 '12 at 13:36
tarashish
1,360 16 30

The declaration "css browser selector" is a bit confusing for a simple javascript. CSS itself doesn't support any selections by browsers! – Armin Feb 17 '12 at 13:39

Sorry but that is what its creator has called it. I have just quoted him :(– tarashish Feb 17 '12 at 13:46

Very lurid phrase of the author ;-) – Armin Feb 17 '12 at 13:49

1 ye, don't really want to add a whole load of js just for this :(but otherwise could be helpful. – Jamie Hutber Jun 17 '12 at 13:49

4

http://www.templatemonster.com/help/how-to-create-browser-specific-css-rules-styles.html

Apply specific CSS rules to Chrome only by using `.selector:not(*:root)` with your selectors:

```
div {
  color: forestgreen;
}
.selector:not(*:root), .div1 {
  color: #dd14d5;
}


<div class='div1'>DIV1</div>
<div class='div2'>DIV2</div>
```

Run code snippet Copy snippet to answer Expand snippet

answered Nov 4 '16 at 8:12
CodeGust
538 2 11 28

2

I am using a sass mixin for chrome styles, this is for `Chrome 29+` borrowing the solution from Martin Kristiansson above.

```
@mixin chrome-styles {
  @media screen and (-webkit-min-device-pixel-ratio:0)
    and (min-resolution:.001dpcm) {
      @content;
  }
}
```

Use it like this:

```
@include chrome-styles {
  .header { display: none; }
}
```

answered Jun 1 '17 at 23:26

svnm
9,392 11 68 93

Firefox now reads this rule as well, so it's no longer good if you want to target solely Chrome. – Mahn Aug 3 at 15:38

2

Have never run across an instance where I had to do a Chrome-only css hack until now. However, I found this to move content below a slideshow where clear:both; affected nothing in Chrome (but worked fine everywhere else - even IE!).

```
@media screen and (-webkit-min-device-pixel-ratio:0) {
    /* Safari and Chrome, if Chrome rule needed */
    .container {
     margin-top:100px;
    }

    /* Safari 5+ ONLY */
    ::i-block-chrome, .container {
     margin-top:0px;
    }
```

answered May 5 '15 at 19:55

Elaine
162 1 3 15

1

So simple. Just add a second class or id to you element at load time that specifies which browser it is.

So basically at the front end, detect browser then set id/class and your css will be befined using those browser specific nametags

answered Oct 25 '16 at 15:00

yan bellavance
2,415 18 54 78

1

if you want we can add class to specific brwoser see [fiddle link][1] [1]:

```
var BrowserDetect = {
        init: function () {
            this.browser = this.searchString(this.dataBrowser) || "Other";
            this.version = this.searchVersion(navigator.userAgent) ||
this.searchVersion(navigator.appVersion) || "Unknown";
        },
        searchString: function (data) {
            for (var i = 0; i < data.length; i++) {
                var dataString = data[i].string;
                this.versionSearchString = data[i].subString;

                if (dataString.indexOf(data[i].subString) !== -1) {
                    return data[i].identity;
                }
            }
        },
        searchVersion: function (dataString) {
            var index = dataString.indexOf(this.versionSearchString);
            if (index === -1) {
                return;
            }

            var rv = dataString.indexOf("rv:");
            if (this.versionSearchString === "Trident" && rv !== -1) {
                return parseFloat(dataString.substring(rv + 3));
            } else {
                return parseFloat(dataString.substring(index +
this.versionSearchString.length + 1));
            }
        },

        dataBrowser: [
            {string: navigator.userAgent, subString: "Edge", identity: "MS Edge"},
            {string: navigator.userAgent, subString: "MSIE", identity: "Explorer"},
            {string: navigator.userAgent, subString: "Trident", identity: "Explorer"},
            {string: navigator.userAgent, subString: "Firefox", identity: "Firefox"},
            {string: navigator.userAgent, subString: "Opera", identity: "Opera"},
            {string: navigator.userAgent, subString: "OPR", identity: "Opera"},

            {string: navigator.userAgent, subString: "Chrome", identity: "Chrome"},
            {string: navigator.userAgent, subString: "Safari", identity: "Safari"}
        ]
    };

    BrowserDetect.init();


    var bv= BrowserDetect.browser;
    if( bv == "Chrome"){
        $("body").addClass("chrome");
    }
    else if(bv == "MS Edge"){
     $("body").addClass("edge");
    }
    else if(bv == "Explorer"){
     $("body").addClass("ie");
    }
    else if(bv == "Firefox"){
     $("body").addClass("Firefox");
    }


$(".relative").click(function(){
$(".oc").toggle('slide', { direction: 'left', mode: 'show' }, 500);
$(".oc1").css({
   'width' : '100%',
   'margin-left' : '0px',
   });
});
```

```
.relative {
  background-color: red;
  height: 30px;
  position: relative;
  width: 30px;
}
.relative .child {
  left: 10px;
  position: absolute;
  top: 4px;
}
.oc {
  background: #ddd none repeat scroll 0 0;
  height: 300px;
  position: relative;
  width: 500px;
  float:left;
}
.oc1 {
  background: #ddd none repeat scroll 0 0;
  height: 300px;
  position: relative;
  width: 300px;
  float:left;
  margin-left: 10px;
}
```

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.7.1/jquery.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/modernizr/2.8.3/modernizr.min.js">
</script>
<div class="relative">
<span class="child">
  o
</span>
</div>
<div class="oc">
<div class="data"> </div>
</div>
<div class="oc1" style="display: block;">
<div class="data"> </div>
</div>
```

Run code snippet Copy snippet to answer Expand snippet

answered Mar 27 '17 at 11:38

neel upadhyay

173 8

1

Chrome provides no own conditionals to set CSS definitions just for it! There shouldn't be a need to do this, cause Chrome interprets websites like defined in w3c standards.

So, you have two meaningful possibilities:

1. Get current browser by javascript (look here)
2. Get current browser by php/serverside (look here)

edited May 23 '17 at 11:47
Community ♦
1 1

answered Feb 17 '12 at 13:38
Armin
10.1k 7 38 60

not always : stackoverflow.com/a/16860022/57883 – Maslow Apr 25 '14 at 19:50

There are definitely reasons why you would want to apply to Chrome but not e.g. Safari or Firefox, e.g. differences in how browsers render <select> elements. A CSS-only solution would be a lot cleaner than having to involve server- or client-side code. – thom_nic Jun 10 '14 at 15:22

1

```
/* saf3+, chrome1+ */
@media screen and (-webkit-min-device-pixel-ratio:0) {
     /*your rules for chrome*/
     #divid{
         position:relative;
         top:-2px;
     }
}
```

check this.it work for me.

edited Sep 10 '16 at 17:15

answered Sep 10 '16 at 17:09
ashanrupasinghe
713 7 17