

How do you organise CSS in your project? [closed]

Asked 15 years, 11 months ago Modified 13 years ago Viewed 2k times



6



Closed. This question is [opinion-based](#). It is not currently accepting answers.



Want to improve this question? Update the question so it can be answered with facts and citations by [editing this post](#).

Closed 8 years ago.

[Improve this question](#)

One of the most challenging thing I have felt while working on (complex) web application is the organizing the CSS. Here are the different approaches we have tried on multiple projects.

1: Have a different stylesheet for every web page/module.

Obviously we were very new to web apps then, and this approach resulted in too many style sheets and too much repetition of styles. We had a tough time to achieve consistency across the application.

2: Have a common style sheets which is shared across the similar web pages.

This worked well for sometime until it became too complex. Also we found that we had too many exceptions which still resulted in tweaking common styles for particular cases, which if done incorrectly can affect different parts of the application and at some point it becomes difficult. Also having a large development team (across different time zones) and tough project timeline didn't helped our cause.

Although #2 works, but still we have seen our products still doesn't have the similar UI quality and consistency as we would like to.

Are there any CSS style guidelines that one should refer for very complex web 2.0 application. How do other people maintain their stylesheets?

css

stylesheet

Share

Improve this question

Follow

asked Jan 7, 2009 at 17:41



MOZILLA

5,970 ● 14 ● 56 ● 59

6 Answers

Sorted by:

Highest score (default)



I've found myself in similar situations.



3



First off, make sure that you're using CSS effectively. If you don't feel like you're an absolute pro at using CSS, take some time to study up and you'll significantly reduce redundancy and end up with a stylesheet that's easier to work with.

In most cases, there isn't much of a performance hit if you consolidate all of your styles into one file, and in fact, splitting your styles into dozens of files just so that you can be sure to exclude any that won't be used is likely to result in longer loading times because of all of the extra requests. But as I'm sure you know, a massive CSS file can quickly grow into a headache to maintain.

Consider this hack to achieve a compromise. Use your language of choice (PHP for me) to serve up your CSS. By that I mean include your style file like this:

```
<link rel="stylesheet" type="text/css" href="styles.ph
```

, have the header of that file return it with the text/CSS content type, and have that file

a) Pull multiple stylesheets into one file and/or b) Change how the styles are written depending on various parameters (file being loaded, user data, time of day, etc.)

A is a good solution overall for reducing developmental headache and page-loading overhead. B will likely necessitate you also setting appropriate file expiration dates to avoid having the browser just ignore whatever

new styles you want to generate at the moment in favor of what was downloaded on the last visit. Since you can usually accomplish the same thing in B as you can by simply having a static stylesheet and dynamically-written element class/ID names, it's generally not an ideal solution unless if you're doing something really strange.

And get a designer to advise you on this kind of stuff. Often it's hard for us developers to wrap our heads around some of the finer points of efficient CSS as well as people trained in that specific area.

Share Improve this answer

answered Jan 7, 2009 at 18:10

Follow



Phantom Watson

2,707 ● 4 ● 25 ● 40



2



I've been in this a lot of times. First, in the early times, I used to do just a stylesheet with everything inside, not much anyway, was the old times; then I decided for your second approach, the first one I luckily thought it was a mistake, too much code and pieces floating around...



The second approach is good up to the time when you start to make questions... I mean:



Should the background style for this div go in the graphic.css or in the layout.css?

Should the font style go in fonts.css or in layout when comes to the width of the P?

Should the margin for a title with the icon position div go to the graphic.css or to the layout.css or to the fonts.css (it would be simpler to use the same declaration for the icon, the text and the position...)?

Then you realized there's something wrong about this approach.

What I do now is commenting. A LOT.

template.css -

```
/* ////////// Headers ////////// */
#header {
    width: 1004px;
    height: 243px; /* 263H-20MARG=243H */
    padding: 20px 0px 0px 0px;
    background-color: #000000;
    background-image: url('../images/redset/top1-bk.png');
    background-repeat: no-repeat;
    background-position: right top;
    clear: both;
}

/* logo */
#logo {
    background-image: url('../images/redset/wh-logo.png');
    background-repeat: no-repeat;
    width: 327px;
    height: 263px;
    float: left;
    margin: -20px 0px 0px 0px;
}
#logo a {
    width: 327px;
    height: 263px;
}

/* Top menu & Banner */
#menuybanner {
```

```

        text-align: center;
/* margin-right: 65px; optional*/
}
#bannerz {
    height: 152px;
    width: 653px;
    text-align: left;
    margin-right: 24px;
/* optional: width: 100%;
margin: 0px */
}
#bigmenu {
    text-align: left;
    margin: 18px 0px 14px 74px;
}

#bigmenu img {
    margin: 0px 22px 0px 0px;
}

```

Originally this would have been in three different css: layout, graphics and texts. Now I know what everyone does.

By the way I know it rises the weight of the archive but I prefer not to do some mixed effects, cause everyone that comes after me and reads the css should be able to understand what I did and css like these:

```

a, .b, .c, .d, #f, #2 { background-color: black; }

```

Are really hard to unveil. Of course if you need to do it, go ahead, but I mean, sometimes they are just grouped like for nothing just to be more cryptic... like moodle... hahaha.

Hope being of help.

See ya.

Share Improve this answer

answered Jul 21, 2009 at 17:20

Follow



Billeeb

216 ● 3 ● 14



1



You want to take advantage of cascading nature of CSS and the ways rules are inherited.

Code first the most general cases and then change specifics.



For a normal size project this should not get out of hand at all.



To see things more clearly you can use an index sheet and call other stylesheets from it. When you want to make changes you will know which stylesheet to go to and you will save time. Here is an example from one of my projects.

```
/*
This is the CSS index page. It contains no CSS code but
*/
@import url("main/reset.css");
@import url("main/colors.css");
@import url("main/structure.css");
@import url("main/html-tags.css");
@import url("main/sign-up-sign-in.css");
@import url("main/pagination.css");
@import url("main/menu-items.css");
@import url("main/teachers-list.css");
@import url("main/footer.css");
```

```
@import url("main/misc-custom-sections.css");  
@import url("main/error-messages.css");
```

Good luck finding your own style.

Share Improve this answer

answered Jan 7, 2009 at 18:04

Follow



[allesklar](#)

9,570 ● 6 ● 38 ● 53

using this way, means we link the page to this index style sheet only? or what way?, thanks – [user287745](#) Dec 15, 2010 at 5:12

Yes, you link to index file in your html file. The code above calls the other files. – [allesklar](#) Dec 18, 2010 at 20:29



1



I use one mastersheet template.css which styles my main template. For any site which requires a seperate bit of styling that can't be covered by the main template I either put it in the site head, if it's short, or create a new sheet for that case.



Ideally I want to design the template.css file to be flexible to cover most cases.



Share Improve this answer

answered Jan 7, 2009 at 18:05

Follow



[Rich Bradshaw](#)

72.9k ● 46 ● 186 ● 241



I typically try to group my CSS by visual elements, and only include relevant stylesheets for a given page to keep

0



my load times low. Using PHP or whatever environment you use to dynamically merge the required stylesheets into a single stylesheet for a given page is a good solution.

One thing that helps me is that I actually created pseudo namespaces for my CSS. I know that CSS 3 has support for namespaces, and that makes it easier, but since some browsers don't support it, this is what I do:

1. Create folders and files relevant to your project (I use Java namespace style)
 - For example
/css/com/mydomain/myprojectname/globalheader.css
2. Next, I use class names that map to the file system location
 - For example <div id='header' class='com-mydomain-myprojectname-globalheader-topClass'>
3. Use separators and good comments in your css file
 - For example /*----- begin link section -----*/
4. Use PHP or whatever to load these files and combine them into one stylesheet on load (you could cache the resulting sheets if you are really clever. The namespace convention will prevent collisions between class names.

While the designers think this is verbose, it makes it really easy to find specific css classes in the file system, without a load time hit. Also, you won't have the problem of one designer / developer overwriting another's classes.

Share Improve this answer

answered Jan 7, 2009 at 18:22

Follow



Heat Miser

19.7k ● 7 ● 37 ● 38



0

maintaining css files is a LOT easier if you can get everyone on board with utilizing cascading properly and keep your targeting strengths to the minimum.



Make sure that elements inherit styles and that overrides aren't too heavy will keep your css from getting crazy. By doing this, you then allow yourself to have just 2 or 3 style sheets for layout/base styles and overrides. If you put heavy control levels on what gets into the layout/base style sheets, and make regular trips in to reassess whats in the overrides sheet to see what can be moved up to the base and what can be simplified you'll free yourselves up to allow people to override at will, but also to keep control of creep.

There's my theory...

Share Improve this answer

edited Dec 10, 2011 at 0:21

Follow



user142162

answered Jan 7, 2009 at 18:22



Steve Perks

5,540 ● 4 ● 30 ● 41
