

The markup strikes back

The importance of good markup

Agenda

1. Introduction
2. Getting it right
3. The standards
4. HTML and doctype
5. Mini Q&A
6. CSS coding guidelines
7. Mini Q&A
8. —Intermission—
9. JavaScript
10. Mini Q&A
11. Browser compatibility
12. Mini Q&A
13. Wicket and markup
14. Coding standards and references
15. Conclusion
16. Q&A

Minimal CSS example

Say you're working on a site that aggregates celebrity interviews. The bread and butter of your site is, then, the interview page. It'll probably look something like this:

```
...
<body class="interview">
  <p class="question">blah blah...</p>
  <p class="answer">yada yada...</p>
  ...
</body>
...
```

To differentiate questions from answers, you decide to make questions boldface:

```
.interview p.question
{
  font-weight: bold;
}
```

Now say some time passes, and you want to add a FAQ to your site. It'll have the same basic question-and-answer structure, right?

```
...
<body id="faq">
  <p class="question">blah blah...</p>
  <p class="answer">yada yada...</p>
  ...
</body>
...
```

It stands to reason you want to differentiate questions from answers here, too. There's no problem, because you've already written a style rule, right? Not so fast. Since your rule only applies to p elements with class question inside elements of class interview, your FAQ questions won't pick up the style rule. To solve this problem, you could create a new style rule like this:

```
#faq p.question
{
  font-weight: bold;
}
```

But that's not very efficient, is it? If you decide later that all questions should also be dark blue, now you have two rules to update. Maintenance has just become twice as difficult.

Now, the smarter thing would have been to use a less specific selector in the first place.

```
.question
{
  font-weight: bold;
}
```

Then you wouldn't have had to do anything when you added the FAQ, and would have gotten that style for free. What's more, if you decide later on to reformat your markup to use `<dt>` and `<dd>` instead of `<p>`, you won't have to update your styles at all.

The moral of this story is that your selectors should always be as general as you can reasonably make them. It promotes repeatability and cleanliness, and if you want to differentiate the two types of questions later on, you won't have to scrounge around for ways to make your new rules more specific than the existing ones to ensure that they take precedence.