

# HTML5 Overview

## *No more long DOCTYPEs*

To ensure browsers render your page in standards-based HTML5, you'll need to specify a new DOCTYPE at the top of your markup page. This is all you need:

```
<!DOCTYPE html>
```

That's right, with HTML5 you won't have to copy/paste long DOCTYPEs from one site to another. Earlier DOCTYPEs were long because they were SGML-based and required a reference to a DTD. HTML5 is not SGML-based, so a simpler DOCTYPE is all that's needed to tell a browser to render a page in W3C standards mode.

## *New Structural Elements*

HTML5 includes new structural tags that can replace the `<div>` tags web designers commonly used to structure their markup. No, `<div>` is not going away, but these new structure tags will make your markup cleaner.

For example, rather than using `<div id="header">`, `<div id="footer">`, or `<div id="content">`, HTML5 now has actual tags for those: `<header>`, `<footer>`, `<article>`, `<nav>`, and `<section>`. For a complete list, see the W3C spec.

So not only can you use these new tags to structure your entire page, you can also use them to structure sections on the page. For example:

```
<header>Page Banner</header>
<article>
  <header>Title of the Article</header>
  <p>Article text.</p>
  <footer>About the author</footer>
</article>
<footer>About the web site</footer>
```

Simply style the main `<header>` and `<footer>` tags, and then their descendant selectors – `article header` and `article footer`.

## Forms

Some of the cooler changes HTML5 involves the markup for web forms. Many interactive features you used to implement with CSS or JavaScript are now part of the HTML5 spec.

### Autofocus and Placeholder Text

Among the new features for `<input>` tags are elements for `placeholder` text and `autofocus`. The `placeholder` text displays in a form field with instructions on what to enter into that field. Once a user clicks on the field, the `placeholder` text disappears. And adding `autofocus` to an `<input>` tag makes that field the focus of the form as soon as the page loads.

```
<form>
  <input type="text" placeholder="Enter some text." autofocus>
  <input type="submit" value="Submit">
</form>
```

### Input Types

Form field data validation is also handled in HTML5, enabling you to specify what kind of data you allow in your input fields rather than relying on JavaScript validators. In fact, HTML5 has 13 new field types, but the most popular will likely be `email`, `url`, `date`, and `time`.

The `date` and `time` types, in particular, offer a native picker control so you don't have to script the controls yourself.

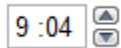
```
<input type="date">
```

2010-09-22 ▾

| Week | Mon | Tue | Wed | Thu | Fri | Sat | Sun |
|------|-----|-----|-----|-----|-----|-----|-----|
| 35   | 30  | 31  | 1   | 2   | 3   | 4   | 5   |
| 36   | 6   | 7   | 8   | 9   | 10  | 11  | 12  |
| 37   | 13  | 14  | 15  | 16  | 17  | 18  | 19  |
| 38   | 20  | 21  | 22  | 23  | 24  | 25  | 26  |
| 39   | 27  | 28  | 29  | 30  | 1   | 2   | 3   |
| 40   | 4   | 5   | 6   | 7   | 8   | 9   | 10  |

Today None

```
<input type="time">
```



In addition to the generic `date` and `time` types, you can also specify `datetime` (includes both date and time controls), `month` (for just month and year selection), and `week`.

So an example of the new `<form>` elements code would be:

```
<form>
  <input type="email" placeholder="Enter email address.">
  <input type="url" placeholder="Enter URL.">
  <input type="date">
  <input type="submit" value="Submit">
</form>
```

## ***What about browser support?***

Browser support for HTML5 gets better each year, but no browser currently supports the entire HTML5 spec. All of the major browsers with the exception of Internet Explorer 8 support between 75% and 90% of the draft spec.

The current browsers with the best HTML5 support are listed below in order of greatest to least:

1. Google Chrome 13
2. Firefox 5
3. Safari 5.1
4. Opera 11.5
5. Internet Explorer 9.0 (IE 8 only supports 21% of HTML5 features)

Check out the [browser compatibility percentages](#) for individual HTML5 features, along with projections on how future versions of each browser will support HTML5.

## ***HTML5 Resources***

The list of valuable HTML5 resources on the web is growing by the day. Below is are some of the resources that contributed to this article.

### [W3C HTML5 Draft Spec](#)

The W3C recommends you read it like you do any other spec: “First, it should be read cover-to-cover, multiple times. Then, it should be read backwards at least once. Then it should be read by picking random sections from the contents list and following all the cross-references.” From section 1.7.1 – who says the W3C doesn't have a sense of humor?

### [W3C: HTML5 differences from HTML4](#)

Does just what the title says. It also provides the rationale behind some of the changes they made.

### [Dive Into HTML5](#)

Mark Pilgrim takes a close look at some of HTML5's more exciting features. Concise and entertainingly written.

### [HTML5 Showcase](#)

Web site showcase where you can view and vote on the best HTML5 sites in the world.

### [W3Schools HTML5 Tag Reference](#)

A handy reference table of all HTML5 tags, as of the current W3C HTML5 spec. It calls out the new tags, as well as tags that are no longer supported.

### [HTML5 Doctor](#)

Let the Doctor (no, not that British fellow) show you how to implement HTML5 in today's browsers.