

# Dreamweaver: Working in Code View



**Instructional Services at KU Libraries**  
**A Division of Information Services**

[www.lib.ku.edu/instruction](http://www.lib.ku.edu/instruction)

---

**Abstract:** Whether you enjoy the control of hand-coding or just want to understand your Web documents' underlying code, this workshop will teach you how to display and edit the code Dreamweaver generates, and introduce you to the XHTML and CSS languages it uses.

---

## Contents

Introduction .....	3
Objectives .....	3
Prerequisites .....	3
XHTML.....	3
XHTML Syntax .....	3
Working with Code .....	7
Code View.....	8
Tag Chooser and Tag Editor .....	9
Tag Inspector .....	10
Quick Tag Editor .....	10
CSS .....	11

**Instructional Services**

Dreamweaver: Working in Code View

---

CSS Syntax..... 11

For More Information..... 12

Related Training Available from Instructional Services ..... 13

Getting Additional Help ..... 13

## Introduction

While Dreamweaver's *raison d'être* is to enable users to develop web pages and create style sheets without having to learn the XHTML and CSS code languages, it also allows one to work directly with this code. This capability extends Dreamweaver's power and scope for experienced web authors, and serves as a valuable learning tool for anyone interested in understanding web documents' underlying code.

## Objectives

This workshop presents a basic overview of XHTML and CSS, and provides an introduction to the tools and interfaces provided by Dreamweaver for inspecting and editing this code.

## Prerequisites

- *Dreamweaver: Getting Started*
- *Dreamweaver: Creating Web Pages*

## XHTML

As explained in *Dreamweaver: Getting Started*, web page files created by Dreamweaver are HTML (HyperText Markup Language) documents, which are plain text files. The text of these files contains not only the content text that appears on the page, but also codes embedded in the content. This markup code reflects the selections made in the design interface of Dreamweaver and provides structure to the content, governing the way the page is rendered in a web browser.

In a web browser, and in Dreamweaver's Design view, this markup is not visible. Using *Code view*, however, you can see (and edit) the unfiltered markup text. What Code view displays are the actual contents of the text file that comprises a web page.

What follows is an overview of the markup language from which these embedded codes are drawn. With an understanding of this language, it is possible to modify—or even create from scratch—web pages through direct editing of this code.

The specific version of HTML generated by Dreamweaver 8 by default is XHTML (Extensible HyperText Markup Language) 1.0.

## XHTML Syntax

### Tags and Elements

XHTML markup is composed of *tags* that define *elements* of the document. Each individual component of a web page—paragraphs, headings, lists, list items, images, links, etc.—is an element.

Tags generally delimit the start and end of an element. Thus an element consists of a *start tag*, *content*, and a corresponding *end tag*.

## Instructional Services

### Dreamweaver: Working in Code View

---

Each tag consists of the element name (as defined by the language specification) enclosed in `<` and `>`. End tags are distinguished by a `/` before the element name. Element names must be in lower case.

- Example:

```
<h3>Interests</h3>
```

This is an **h3** (heading 3) element. `<h3>` is the start tag, `Interests` is the content, and `</h3>` is the end tag.

Some elements are inherently empty, enclosing no content. **br** (line break) is an example of an empty element. These elements are written as a single tag with a `/` *after* the element name, e.g., `<br />`.

---

**Note:** There is a space between the element name and the `/`. While not required by the language specification, this space is necessary for maximum browser compatibility.

---

Elements may be included in the content of other elements. When they are, they must be properly nested.

- Example:

```
✓ <p>This paragraph contains an <em>emphasized  
phrase</em>.</p>
```

The above is correct, because the **em** (emphasis) element starts and ends completely within the containing **p** (paragraph) element. By contrast

```
* <p>This paragraph contains an <em>emphasized  
phrase</p>.</em>
```

is *incorrect*, because the **p** and **em** elements overlap in such a way that neither fully contains the other. This is invalid.

## Attributes

Many tags have one or more additional components, known as *attributes*. Attributes specify properties of an element. Not every element uses attributes, but some elements are meaningless without specific attributes. Some attributes can be applied to many different types of elements, while some are only for use with specific elements.

An attribute appears in the start tag, but not in the end tag. The attribute name follows the element name, is separated from it by a space, and must be in lower case. Each attribute has a value, the attribute and its value are separated by an `=`, and the value is enclosed in quotation marks (`"`). An element may have more than one attribute. Multiple attributes are separated from one another by spaces, and can appear in any order.

- Example:

```
<h3 id="jsmith_interests">Interests</h3>
```

Here, the attribute **id** has been added to this **h3** element, assigning to it a unique identifier, which is specified by the attribute's value.

## White Space

*White space* characters (spaces, tabs, and returns) are insignificant in XHTML except as separators (e.g., the white space separating an element name from an attribute name). All sequences of white space characters are equivalent.

- Example:

This:

```
<ul>
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ul>
```

is equivalent to this:

```
<ul> <li>First item</li> <li>Second item</li>
<li>Third item</li> </ul>
```

This means that you are free to use white space as you see fit to make your code readable, but it also means that it has no effect on the rendered page; all desired line and paragraph breaks, etc., must be specified with appropriate elements.

## Character References

Symbols and other characters that cannot be input directly with the keyboard can be inserted into XHTML code using *character references*. A character reference consists of `&`, followed by the character name, followed by `;`. For example, `&copy;` is the character reference for ©. A list of character references is at <http://www.w3.org/TR/html4/sgml/entities.html>.

Three character references for characters that can be typed on the keyboard are nonetheless useful, because they represent parts of XHTML syntax. They are `&lt;`, which should be used for `<` to avoid confusion with the beginning of a tag; `&gt;`, which should be used for `>` to avoid confusion with the end of a tag; and `&amp;`, which should be used for `&` to avoid confusion with the beginning of a character reference.

## Comments

As with other computer languages, it is possible to include *comments*. A comment is a descriptive annotation that you insert in XHTML code to explain the code or provide other information. The text of the comment appears only in Code view and is not displayed in a browser.

Comments in XHTML are demarcated by `<!--` and `-->`.

- Example:

```
<!-- This is a comment. It will not be displayed. -->
```

## Document Structure

XHTML requires certain elements to be present, in a particular arrangement, in every XHTML document. These elements comprise a framework for the entire document contents that looks like this:

## Instructional Services

### Dreamweaver: Working in Code View

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title> </title>
</head>

<body>
</body>
</html>
```

The first tag is the *document type declaration*, which declares, in a specific way, that the file is an XHTML document, and what version of XHTML is being used. In the example above, the document type declaration indicates an XHTML 1.0 transitional document. “Transitional” indicates that the document may contain *deprecated* elements and attributes, those that have been outdated by newer constructs. A “strict” document does not allow deprecated elements or attributes. The document type declaration for an XHTML 1.0 strict document looks like this:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

By default, Dreamweaver 8 produces XHTML 1.0 transitional documents.

**Note:** The document type declaration tag has no accompanying end tag, and has a syntax unlike that of other tags in an XHTML document. This is because, strictly speaking, the document type declaration is not an XHTML tag, but an SGML (Standard Generalized Markup Language) tag. (XHTML is itself defined in SGML, which is a language for describing markup languages.)

After the document type declaration, an **html** element contains the remainder of the document. The **xmlns** attribute defines the *namespace*. The namespace is the formal collection of element and attribute names that comprise the markup vocabulary of XHTML, identified by a URL reference.

Within the **html** element are two elements: **head**, then **body**. **head** contains **title**, which specifies the page title. **body** is the container for all of the page content that is presented to the user in a browser.

Dreamweaver automatically places these required elements in each new document. (You can see this by creating a new file and immediately switching to Code view, where you will see markup much like that in the box above.)

Dreamweaver includes an additional tag automatically, in **head**, prior to **title**:

```
<meta http-equiv="Content-Type" content="text/html;
charset=ISO-8859-1" />
```

This identifies the data type, character set, and character encoding of the page, which a web browser will use to properly decode it.

## Basic Elements

Below is a selected list of basic XHTML elements. These are the elements that correspond to the Design view techniques outlined in the *Dreamweaver: Creating Web Pages* workshop.

---

- h1** Heading 1
  - h2** Heading 2
  - h3** Heading 3
  - h4** Heading 4
  - h5** Heading 5
  - h6** Heading 6
  - p** Paragraph
  - br** Line break. *An empty element.*
  - hr** Horizontal rule. *An empty element.*
  - ol** Ordered list
  - ul** Unordered list
  - li** List item
  - img** Image. Required attributes are **src**, the image source URL, and **alt**, the alternate text. Dreamweaver also includes **width** and **height** by default.
  - a** Hyperlink. The **href** attribute, required, specifies the link URL.
- 

## Reference

For a more complete HTML reference, click the **Help** menu, then click **Reference** to open the Reference panel. Choose **O'REILLY HTML Reference** from the **Book** pop-up menu, then select a tag from the **Tag** pop-up menu.

## Working with Code

While Code view is the most obvious way to view and edit a document's markup, it is not the only interface Dreamweaver provides for working directly with the underlying code of a page. This section describes the various possibilities.

## Code View

Click the **Code** button in the Document toolbar to show Code view in the document window, or click the **Split** button to show both Code and Design views simultaneously. You can also click the **Window** menu, then click **Code Inspector**. This opens a separate coding window that works just like Code view.

## Code Coloring

Code view displays different kinds of tags, attribute values, etc., in different colors, to make the code more readable.

## Code Hints

You can, of course, type directly into the document window in code view to manually edit code. The code hints feature can speed up this process.

Whenever you type a < to begin a tag, a list of tags appears (after a user-definable delay) at the insertion point. You can select the tag name from the list, in lieu of typing it, to insert it. If you begin typing the tag name, the list will scroll to it. You can close the code hints menu at any time by pressing Escape.

If you press the spacebar after entering a tag name, a list of attributes for the tag appears.

When a URL value is required, the word **Browse** appears, which can be used to automatically enter the path for a local link.

Code hinting also auto-completes end tags when you type </.

## Coding Toolbar

By default, Code view displays the Coding toolbar along the left edge of the document window. The Coding toolbar contains buttons that let you perform many standard coding operations, such as collapsing and expanding code selections, highlighting invalid code, applying and removing comments, indenting code, and inserting recently used code snippets.

Among the buttons that you may find useful:



**Select Parent Tag** selects the content and opening and closing tags that surround the insertion point or the current selection. This is useful for quickly and reliably selecting whole elements. (Note also that the tag selector in the status bar at the bottom of the document window continues to function in Code view.)



**Highlight Invalid Code** highlights (some) invalid code in yellow, to make it easier to spot errors.



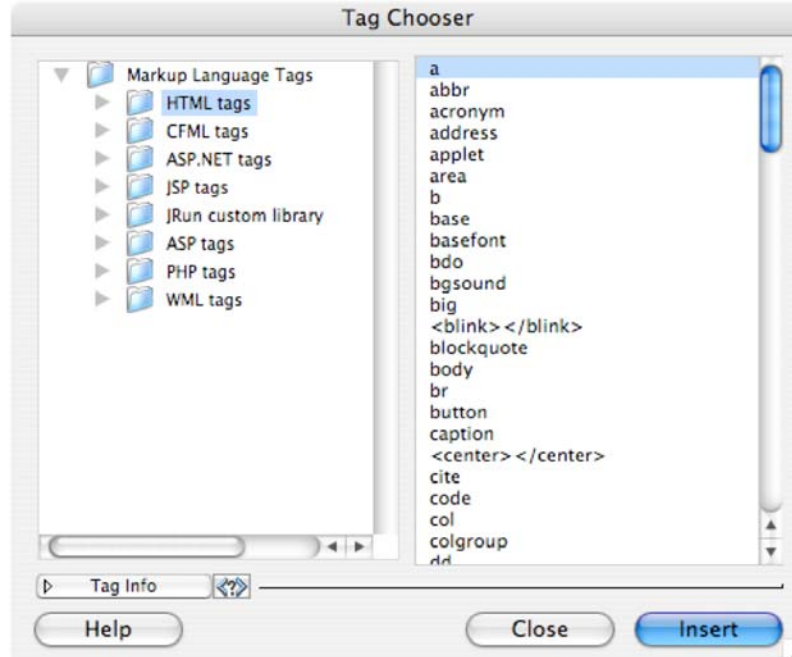
**Wrap Tag** wraps the selected code with the tag pair you specify (see *Quick Tag Editor*, p. 10).

To find out what each button does, position the pointer over it until a tooltip appears. For a description of each button, see **Inserting code quickly with the Coding toolbar** in Dreamweaver Help.

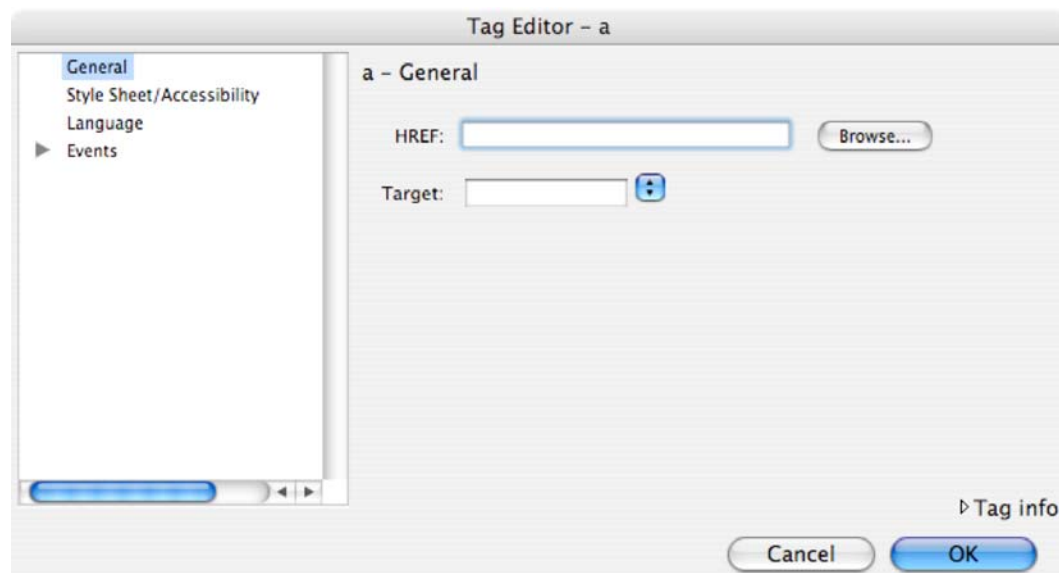
## Tag Chooser and Tag Editor

You can add tags and specify their attribute values through dialog box selections using the *Tag Chooser*. To insert a tag using the Tag Chooser:

1. Click the **Insert** menu, then click **Tag**.



2. In the **Tag Chooser**, click the **HTML tags** category (or a subcategory thereof) on the left.
3. Click the desired tag in the list on the right.
4. Click **Insert**. A *tag editor* appears, providing attribute options specific to the tag you chose.



5. Make whatever modifications you choose in the tag editor, then click **OK**.

## Instructional Services

### Dreamweaver: Working in Code View

---

6. Click the **Close** button in the **Tag Chooser** window.

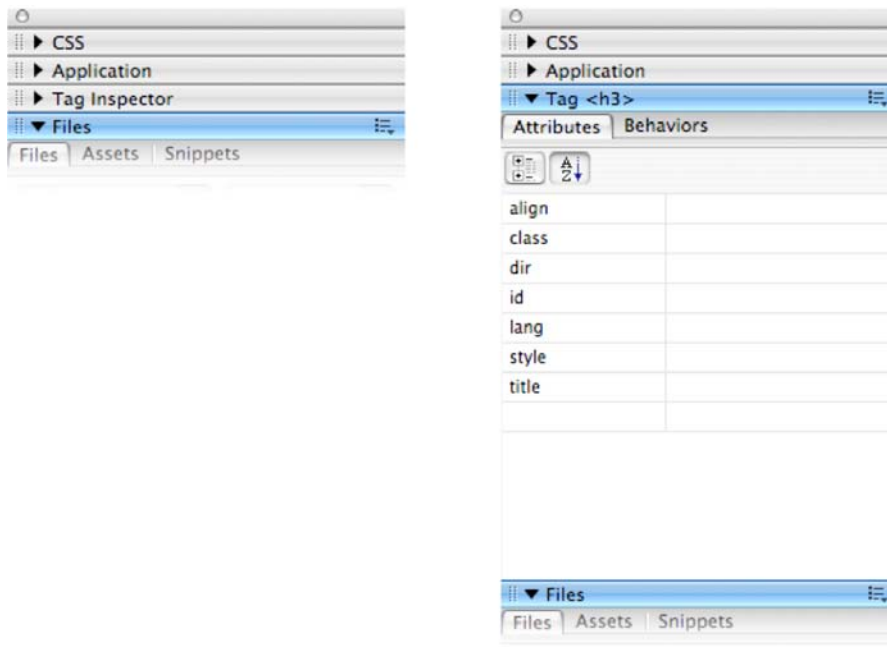
You can also use the tag editor to modify the attributes of an existing tag:

1. Select the tag in the code that you want to edit.
2. Click the **Modify** menu, then click **Edit Tag**.
3. Edit the attribute values as you see fit, then click **OK**.

The Tag Chooser is an excellent way to see what tags are available, and the tag editor can show you what attributes pertain to a given tag.

## Tag Inspector

Another way to edit the attributes of a tag is with the *Tag inspector*. By default, the Tag Inspector is open but collapsed. To expand it, click the **Tag Inspector** panel group's title bar. If it is closed, open it by clicking the **Window** menu, then clicking **Tag Inspector**.



*The Tag Inspector collapsed (left) and expanded (right)*

Next, click the **Attributes** tab. You can use the two buttons at the top of the panel to switch between categorized and alphabetized list views.

The Tag Inspector shows all applicable attributes for the currently selected tag. You can add or modify an attribute value by editing in the right column.

## Quick Tag Editor

Finally, you can directly add and edit individual XHTML tags even from Design view, using the *Quick Tag Editor*. To use the Quick Tag Editor, click the **Modify** menu, then click **Quick Tag Editor**. Add or edit the tag and attributes in the box provided. Hints menus appear to assist you (see *Code Hints*, p. 8).

## CSS

While XHTML specifies the *structure* of page content, CSS (Cascading Style Sheets) governs how web browsers *present* that content.

A style sheet file consists of a set of rules that pertain to page content, but contains no content itself. (The content is in an XHTML file.) There is, therefore, no Design view of a style sheet file. Dreamweaver uses the **CSS** panel group and dialog boxes to provide a visual interface for defining these rules. (This is covered in the *Dreamweaver: Visual Design using CSS* workshop.) The only way to open a style sheet file directly is in Code view. Another use for Code view, then, is for direct editing of style sheet files.

Below is a brief, basic overview of the CSS language.

## CSS Syntax

### Selectors

A CSS rule begins with a *selector*, which indicates to which items on a page it will apply.

The simplest, and most common, selector is a *type selector*, which is the name of an XHTML element. A type selector matches every instance of the element type. This makes it trivial to apply a presentation rule to all level-one headings, or all paragraphs, or all list items, etc. Using **body** as a selector is an effective way to apply a rule to a whole document.

There are also more sophisticated *descendant* and *child selectors*, which match any specific element that is within another (all paragraphs in a particular **div** element, for example); *attribute selectors*, which match any specific element with a given attribute set, or with a given attribute with a particular value; and *pseudo-class selectors* that match elements in a particular state, such as visited hyperlinks or an item over which the pointer is hovering.

*Class selectors* match any element whose **class** attribute is set to a particular value. This enables a rule to be applied selectively, rather than to all instances of a given element. In the CSS selector, the class name is preceded by a “.”.

*ID selectors* match a particular element in a document based on its **id** attribute. In the CSS selector, the id value is preceded by a #.

### Properties

For each rule, the selector is followed by a *declaration block*, which is a matched pair of curly braces ( { and } ) containing a list of *declarations* separated by semicolons (;). Each declaration consists of a *property*, followed by a colon (:), followed by a *value*. It is the properties and their values that specify the details of how the page items matched by the selector should be presented.

- Examples:

```
h1 {  
    font-weight: bold;  
    color: blue;  
}
```

This rule uses a type selector to set all **h1** elements to be bold and blue.

```
.pastoral {  
    color: green;  
}
```

This rule uses a class selector to set all elements whose **class** attribute has the value “pastoral” to be green.

```
#chapter1 {  
    text-align: center;  
}
```

This rule uses an ID selector to set the element whose **id** attribute has the value “chapter1” to be centered.

## White Space

As with XHTML, white space characters (spaces, tabs, and returns) are insignificant in CSS.

- Example:

This:

```
body {  
    color: navy;  
    background-color: silver;  
}
```

is equivalent to this:

```
body {color: navy; background-color: silver;}
```

## Comments

As with XHTML, it is possible to include comments (see *Comments*, p. 5).

Comments in CSS are demarcated by /\* and \*/.

- Example:

```
/* This is a comment. */
```

## Reference

For a complete CSS reference, click the **Help** menu, then click **Reference** to open the Reference panel. Choose **O'REILLY CSS Reference** from the **Book** pop-up menu, then select a property from the **Style** pop-up menu.

## For More Information

Here are some additional sources of information:

- Adobe Dreamweaver Hand Coding Support  
<http://www.adobe.com/support/dreamweaver/htmljava.html>

## Instructional Services

### Dreamweaver: Working in Code View

---

- XHTML 1.0 specification  
<http://www.w3.org/TR/xhtml1/>
- World Wide Web Consortium (W3C) HTML Home Page  
<http://www.w3.org/MarkUp/>
- CSS2 specification  
<http://www.w3.org/TR/REC-CSS2/>
- W3C CSS Home Page  
<http://www.w3.org/Style/CSS/>

## Related Training Available from Instructional Services

All workshops offered by Instructional Services are free to KU students, staff, faculty, and approved affiliates.

To learn more about or register for workshops, receive automatic announcements of upcoming workshops, and track workshops you've registered for and have attended, visit [www.infotraining.ku.edu](http://www.infotraining.ku.edu). For further workshop related questions, please email [training@ku.edu](mailto:training@ku.edu).

## Getting Additional Help

IT Customer Service Center staff are available for computing assistance seven days a week. Call, email, or stop by in person at the Computer Center to get help.

(785) 864-8080

[itcsc@ku.edu](mailto:itcsc@ku.edu)

[www.technology.ku.edu/~helpdesk](http://www.technology.ku.edu/~helpdesk)

*Last Update: 09/20/2007*